ANNUAL REPORT APPENDIX

FEMTOSECOND CARRIER PROCESSES IN COMPOUND SEMICONDUCTORS AND REAL TIME SIGNAL PROCESSING

MAY 1, 1992 - APRIL 30, 1993

CONTRACT #F49620-90-C-0039

UNCLASSIFIED
SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

Form Approved
OMB No 0704-0188

1a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED	1b RESTRICTIVE MARKINGS
2a. SECURITY CLASSIFICATION AUTHORITY	3 DISTRIBUTION/AVAILABILITY OF REPORT Approval for public release. Distribution unlimited.
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE	
4. PERFORMING ORGANIZATION REPORT NUMBER(S)	5. MONITORING ORGANIZATION REPORT NUMBER(S)
6a. NAME OF PERFORMING ORGANIZATION Cornell University 6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION Air Force Office of Scientific Research
6c. ADDRESS (City, State, and ZIP Code) 119 Phillips Hall Ithaca, NY 14853-5401	7b ADDRESS (City, State, and ZIP Code) Building 410, Bolling Air Force Base Washington, DC 20332-6448
8a. NAME OF FUNDING/SPONSORING ORGANIZATION 8b OFFICE SYMBOL (If applicable)	9 PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER F49620-90-C-0039
8c. ADDRESS (City, State, and ZIP Code)	10 SOURCE OF FUNDING NUMBERS PROGRAM ELEMENT NO. / PROJECT NO. / TASK NO / WORK UNIT ACCESSION NO

11. TITLE (Include Security Classification)
Femtosecond Carrier Processes in Compound Semiconductors and Real Time Signal Processing

12. PERSONAL AUTHOR(S)
J. Peter Krusius

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15 PAGE COUNT
Annual Report Appendix	FROM 5/1/92 TO 4/30/93	93/03/10	274

16. SUPPLEMENTARY NOTATION

17. COSATI CODES		
FIELD	GROUP	SUB-GROUP

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
Compound semiconductor, organometallic vapor phase epitaxy, femtosecond laser probing, Monte Carlo simulation, VLSI, fault tolerance, computer architecture

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

This report is the annual report on research conducted under the auspices of the Joint Services Electronics Program at Cornell University. The research is grouped under two themes: (a) femtosecond carrier processes in compound semiconductors, and (b) real time signal processing. Results on OMVPE materials growth, femtosecond laser probing of hot carriers, and ensemble Monte Carlo simulations are reported on under the first theme. Accomplishments on VLSI algorithms, fault tolerant architectures, and architectures with multiple functional units for signal processing are given under the second theme.

93-08085

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT ☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT. ☐ DTIC USERS	21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED
22a. NAME OF RESPONSIBLE INDIVIDUAL J. Peter Krusius	22b. TELEPHONE (Include Area Code) 607-255-3401 22c. OFFICE SYMBOL

DD Form 1473, JUN 86 *Previous editions are obsolete.*

SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

93 4 16 148

TABLE OF CONTENTS

TASK 1 OMVPE GROWTH OF III-V ALLOYS FOR NEW HIGH SPEED ELECTRON DEVICES

J. R. Shealy

Arsine flow requirement for the flow modulation growth of high purity GaAs using adduct-grade triethylgallium

B. L. Pitts, D. T. Emerson, and J. R. Shealy
OMVPE Facility, School of Electrical Engineering, Cornell University, Ithaca, New York 14853

(Received 1 May 1992; accepted for publication 14 August 1992)

Using arsine and triethylgallium with flow modulation, organometallic vapor phase epitaxy can produce high purity GaAs layers with V/III molar ratios near unity. We have estimated that under appropriate growth conditions the arsine incorporation efficiency into epitaxial GaAs can exceed 30%. The arsine flow requirement for obtaining good morphology has been identified over a range of substrate temperatures using adduct-grade triethylgallium. The process described reduces the environmental impact and life safety risk of the hydride based organometallic vapor phase epitaxial method.

Organometallic vapor phase epitaxy (OMVPE) has demonstrated the ability to produce a variety of device quality III-V compounds and structures. With a carefully designed gas flow switching apparatus, interface abruptness approaching a perfect compositional change across a single atomic layer has been realized. Optimized results are often achieved using reduced growth pressures. It has been suggested that growing at reduced pressures often results in sharper interfaces, reduced autodoping, and lower growth rates which increase the accuracy of layer control.[1] Furthermore, in many reactor cell designs (e.g., vertical barrel) reduced pressure is required to eliminate gas recirculation due to convection forces. One of the disadvantages in growing high purity III-V compound semiconductors by low pressure OMVPE is the increased flow requirement of highly toxic hydrides (e.g., arsine, phosphine). In conventional reduced pressure OMVPE using trimethylgallium (TMG) and arsine (AsH_3), high molar V/III ratios are necessary to obtain high purity GaAs.[2] Efforts have been made to reduce AsH_3 consumption, including precracking[3] of the arsine and substituting triethylgallium (TEG) for TMG.[4–7] None of these methods have resulted in device quality material with V/III ratios near unity. Less toxic group V liquid sources are presently available which, at V/III ratios of 10 or greater, yield 77 K mobilities greater than 100 000 cm^2/V s.[8] Low pressure OMVPE growth is still done at relatively high V/III ratios. This leads to potential safety hazards due to the expulsion of the excess arsine that does not participate in the growth process, and to the increased handling of the source containers. Efforts to minimize high pressure cylinder storage include an on-demand arsine gas generator, but a low 77 K mobility was observed (76 000 cm^2/V s).[9] In this study using flow modulation epitaxy (FME),[10] we demonstrate a process which does not require excess AsH_3 and which produces high quality GaAs epitaxial layers (77 K mobilities of 90 000 cm^2/V s). The process described allows for small quantities of arsine storage in the facility and could be used in conjunction with hydride generator technologies to minimize the safety issues involved in OMVPE growth of many III-V compounds.

The use of TEG and AsH_3 has been proven to give lower background carbon concentrations in GaAs than the widely used TMG.[4–7] The first high purity GaAs result by OMVPE, by Seki *et al.*,[4] used TEG and AsH_3 at a V/III ratio of 2 and reported a 77 K mobility of 120 000 cm^2/V s. At reduced pressures, the highest purity GaAs was grown at a V/III ratio of 17.5, resulting in 77 K mobility of 190 000 cm^2/V s on approximately 10 μm films.[5] High purity GaAs has been produced at a V/III ratio of 8 (Ref. 5) whereas in this study, significantly lower V/III ratios result in similar quality films. The reduction of the V/III ratio is attributed to the use of flow modulation. Low V/III ratios (V/III = 5–20) are also used in metalorganic molecular beam epitaxy,[11] but best results are p-type and have carbon concentrations exceeding mid 10^{14} cm^{-3}.

An investigation of high purity GaAs grown by low pressure OMVPE with flow modulation and with V/III ratios approaching unity is reported. A V/III ratio of 1.8 resulted in a film with a 77 K mobility exceeding 90 000 cm^2/V s and a room-temperature mobility exceeding 8000 cm^2/V s. Comparable results are observed with a V/III ratio of unity provided substrate temperatures greater than 610 °C are used. Finally, the AsH_3 flow requirement for this process has been identified and determined to be a strong function of substrate temperature if high quality surfaces are to be obtained. All films which were observed to have mirrorlike surfaces are of high purity as inferred from low-temperature Hall and photoluminescence data. Growths carried out with subunity V/III ratios were characterized by poor surfaces and reduced growth rates, indicative of arsenic diffusion limited growth.

GaAs layers were grown using FME at low pressure (76 Torr) in a vertical barrel multichamber OMVPE system.[12] In this system substrates are rotated through groups III and V rich spatially separated zones without valve switching. During the group III exposure cycles the local V/III ratio is estimated to be 25% of the average value. The substrate then enters a group V exposure cycle. The V/III ratio quoted throughout represents the average values determined by the total injected reactant fluxes. The susceptor was rotated at 0.1 rev/s and the growth rate was 8 monolayers/cycle (1 μm/h).

Undoped epitaxial layers were grown using adduct-purified TEG[20] and AsH_3 (100%). Layer thicknesses ranged from 3–6 μm. The substrates, (100) Si-doped

 0003-6951/92/422054-03$03.00

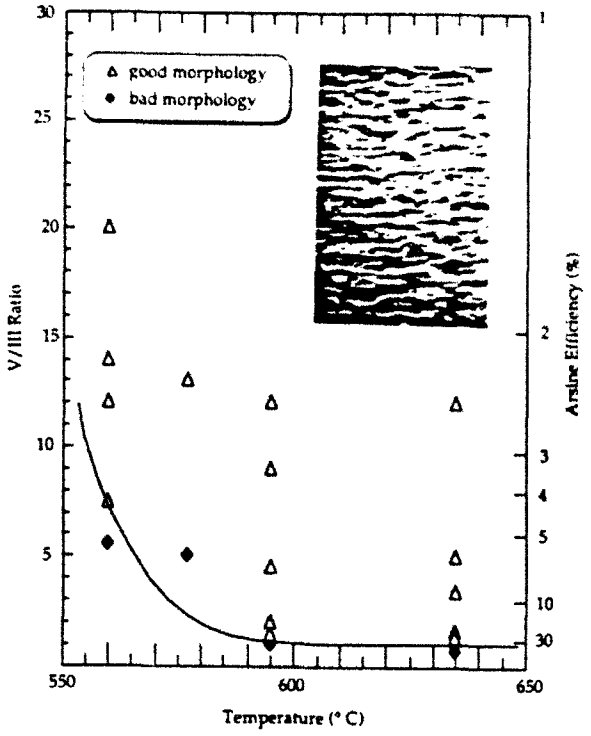


FIG. 1. The effects of V/III ratio and substrate temperature on surface morphology of GaAs using adduct purified TEG. Also the right-hand side vertical axis provides estimates of the arsine growth efficiency which is dependent on the V/III ratio. The arsine growth efficiency is defined as the ratio of the arsine in the gas stream to arsenic incorporated in the GaAs deposited on the active portion of the susceptor. The solid line was drawn empirically to suggest the transition between the good and bad morphology regions. Inset is a SEM micrograph (magnified 200×) illustrating what is meant by poor morphology.

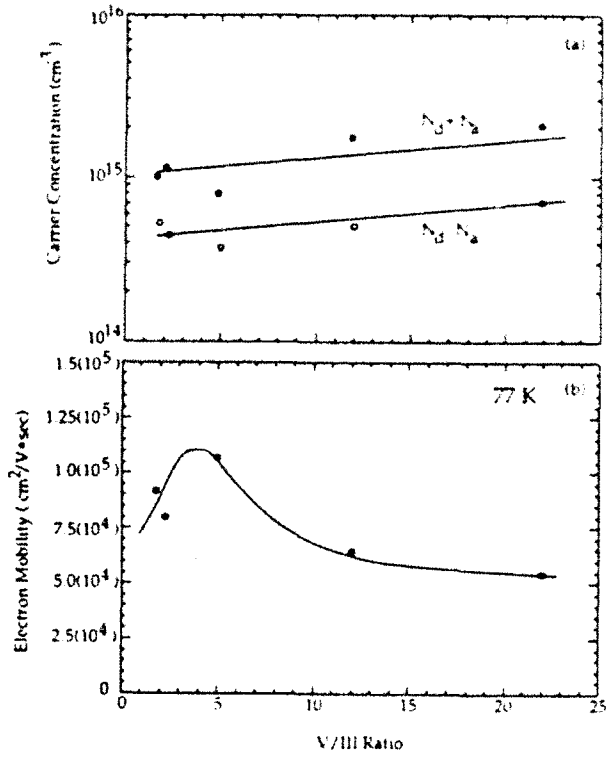


FIG. 2. Hall measurement results showing dependence of (a) net ($N_d - N_a$) and total ($N_d + N_a$) impurity concentration and (b) 77 K mobility on V/III ratio in undoped GaAs grown at 635 °C. The lines are drawn empirically to suggest a trend in the data.

n^+ GaAs and (100) semi-insulating GaAs, were first rinsed in organic solvents and then etched in $5H_2SO_4{:}1H_2O_2{:}1H_2O$ prior to growth. The TEG was held at 23 °C while a H_2 flow of 50 sccm was passed through the bubbler, maintained at 100 Torr. The growth temperature ranged from 560 to 635 °C, while the V/III ratio varied from 0.7 to 22. Growth rate measurements were performed using angle lapping and staining. Thickness uniformity was ±1.2% across a 1.5 in. diam wafer. Carrier concentrations and mobilities were measured using the van der Pauw method in a magnetic field of 3.5 kG at both 300 and 77 K. Low-temperature (1–20 K) photoluminescence (PL) was used to investigate the excitonic features as well as to identify the acceptor impurities.

Arsine efficiency was calculated for the reactor and is defined as the ratio of the AsH_3 in the gas stream to the amount of As incorporated in the GaAs on the entire active portion of the susceptor. The maximum possible efficiency (V/III = 1) is 31%, where high quality films are observed using low-temperature PL. Conducting films are obtained at higher V/III ratios ($\mu_{77\,K} = 93\,000$ cm^2/V s and $\mu_{300\,K} = 8000$ cm^2/V s) where the arsine AsH_3 efficiency is calculated to be 17.2% (V/III = 1.8). Previous studies in this reactor using TMG found that the *p-n* transition occurred around V/III = 30 at the same growth pressure, growth rate and flow modulation where a V/III = 70 resulted in a 77 K mobility of 96 000 cm^2/V s.[12] For V/III ratios less than 30, the films suffered from increasing levels of carbon contamination. Thus, a 20-fold improvement in the efficiency of AsH_3 resulted using adduct-purified TEG in place of TMG. There is also no *p-n* transition with a decrease in the V/III ratio using adduct-purified TEG.

The V/III ratio and growth temperature criteria for good surface morphology were investigated over the range from 560 to 635 °C. In Fig. 1, a good morphology/bad morphology transition curve is shown to suggest the minimum V/III ratio required at a given growth temperature. In addition, the V/III ratio relationship to AsH_3 efficiency is also provided in the figure. When the substrate temperature is below 610 °C, more arsine must be supplied as shown, indicating that less AsH_3 is being pyrolyzed. As the substrate is increased beyond 610 °C, the transition from good morphology to poor morphology approaches a constant V/III value of unity. This suggests that the AsH_3 arriving at the growth surface is completely pyrolyzed, and maximum AsH_3 efficiency can be achieved when the growth temperature exceeds 610 °C.

The impurity concentration and low temperature (77 K) mobility for samples grown at 635 °C with V/III ratios from 1.8 to 22 are given in Fig. 2. Total impurity concentration ($N_d + N_a$) was estimated using the empirical relation given by Stillman and Wolfe.[15] Net impurity concentration varied from 3.7(10^{14}) to 6.9(10^{14}) cm^{-3} while $N_d + N_a$ varied from 7.7(10^{14}) to 2.0(10^{15}) cm^{-3}. The minimum value in each case was obtained for a V/III ratio

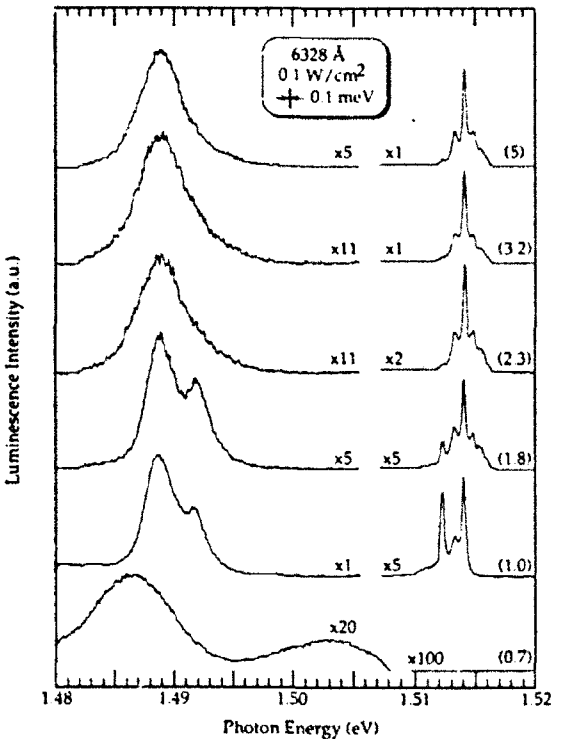


FIG. 3. Low-temperature (1 K) PL spectra of undoped GaAs layers grown at 635 °C. The luminescence intensity is magnified by the factors shown, and the average V/III ratios are given in parenthesis at the far right of the figure. Excitation conditions and experimental resolution are as indicated.

of 5, where residual carbon levels are in the low 10^{14} cm^{-3} range. The 77 K mobility varied from 55 000 to 110 000 cm^2/V s, with the maximum value obtained for a V/III ratio of 5. As indicated in Fig. 2, the impurity concentration is, in general, an increasing function of V/III ratio. The highest room-temperature mobility was greater than 8000 cm^2/V s. High purity results are readily observed with PL at a V/III ratio of unity for growth temperatures greater than 630 °C, but these films did not exhibit electrical conduction due to increased acceptor compensation. Subunity growth had Ga rich surfaces which prevented contact formation.

Low-temperature PL spectra of samples grown at 635 °C are shown in Fig. 3. The dominant feature in the excitonic region of the spectra is that of the neutral donor exciton (D^0,X). The neutral acceptor exciton peak (A^0,X) is negligible in samples grown with V/III ratios greater than 1.8, strongly indicating *n*-type material.[2] This is consistent with Hall measurement results. Two acceptor behavior in the PL spectra was observed using variable temperature measurements (1–20 K) for V/III ratios less than 1.8. Although not pronounced in the 1 K spectra, a signature of band-to-acceptor transitions of carbon ($\sim$1.493 eV) is evident in spectra observed at 12 K for V/III ratios less than 1.8. The peaks which appear in the spectra for V/III ratios of 1.8 and unity at 1.492 eV behave as band-to-acceptor transitions with the acceptors tentatively identified as magnesium. The corresponding donor-acceptor pair luminescence observed at 1.489 eV supports this assignment of acceptor species which is believed to originate from the arsine source.[9] The absence of carbon acceptor related luminescence for V/III ratios greater than 1.8 suggests that sufficient AsH_3 is present to remove the carbon from the growth surface. As the V/III ratio approaches unity, the magnitude of the neutral acceptor exciton peak is comparable to that of the neutral donor exciton peak. Finally, when the V/III ratio is reduced to subunity (0.7), the normal excitonic features are completely absent from the spectra. New spectral features appear much weaker in intensity, possibly due to defect related exciton emission at photon energies near 1.503 eV,[15] commonly observed in molecular beam epitaxy materials.

In summary, by applying flow modulation techniques, highly efficient use of AsH_3 has been demonstrated with an optimized low pressure OMVPE process for the first time. It has been observed that below 610 °C, more arsine must be supplied to sustain good morphology. Above 610 °C, maximum AsH_3 efficiency (V/III = 1) can be obtained while maintaining specular surfaces. In addition, there was no *p-n* transition region in the range studied. Using adduct-purified TEG and AsH_3 in OMVPE at reduced pressure, we have demonstrated a near-unity V/III ratio resulting in a 77 K mobility exceeding 90 000 cm^2/V s.

The authors wish to thank B. P. Butterfield, M. J. Matragrano, and K. L. Whittingham for technical assistance. N. Scott is gratefully acknowledged for his support of the development of the OMVPE facility at Cornell. This work was supported by the Joint Services Electronics Program under Grant No. F49620-90-C-0039, the Strategic Defence Initiative Objective under Contract No. N00014-89-J-1311, and the Defense Advanced Research Projects Agency under Contract No. MDA97290C0058 Optoelectronics Technology Center.

[1] J. P. Duchemin, M. Bonnet, F. Koelsch, and D. Huyghe, J. Cryst. Growth **45**, 181 (1978).
[2] J. R. Shealy, V. G. Kreismanis, D. K. Wagner, G. W. Wicks, W. J. Schaff, Z. Y. Xu, J. M. Ballantyne, L. F. Eastman, and R. Griffiths, Inst. Phys. Conf. Ser. **65**, 109 (1983).
[3] M. Ogura, Y. Ran, M. Morisaki, and N. Hase, Jpn. J. Appl. Phys. **22**, L630 (1983).
[4] Y. Seki, K. Tanno, K. Iida, and E. Ichiki, J. Electrochem. Soc. **122**, 1108 (1975).
[5] S. K. Shastry, S. Zemon, and P. Norris, Inst. Phys. Conf. Ser. **63**, 81 (1986).
[6] T. F. Kuech and R. Potemski, Appl. Phys. Lett. **47**, 821 (1985).
[7] R. Bhat, P. O. O'Connor, H. Temkin, and R. Dingle, Inst. Phys. Conf. Ser. **63**, 101 (1981).
[8] G. Haacke, S. P. Watkins, and H. Burkhard, Appl. Phys. Lett. **56**, 478 (1990).
[9] S. G. Hummel, Y. Zou, C. A. Beyler, P. Grodzinski, P. D. Dapkus, J. V. McManus, Y. Zhang, B. J. Skromme, and W. I. Lee, Appl. Phys. Lett. **60**, 1483 (1992).
[10] N. K. Kobayashi, T. Makimoto, and Y. Horikoshi, Jpn. J. Appl. Phys. **24**, L962 (1985).
[11] W. T. Tsang, J. Cryst. Growth **105** 1 (1990).
[12] J. R. Shealy, J. Cryst. Growth **87**, 350 (1988).
[13] A. C. Jones, A. K. Holliday, D. J. Cole-Hamilton, M. M. Ahmad, and N. D. Gerrard, J. Cryst. Growth **68**, 1 (1984).
[14] G. E. Stillman and C. M. Wolfe, Thin Solid Films **31**, 89 (1976).
[15] H. Kunzel and K. Ploog, Appl. Phys. Lett. **37**, 416 (1980).

J. Electrochem. Soc., Vol. 139, No. 10, October 1992 © The Electrochemical Society, Inc.

The Use of Ultraviolet Radiation at the Congruent Sublimation Temperature of Indium Phosphide to Produce Enhanced InP "Schottky" Barriers

James Singletery, Jr. and James R. Shealy

School of Electrical Engineering, Cornell University, Ithaca, New York 14853

ABSTRACT

This paper describes an ultraviolet radiation-assisted process, optimized around the congruent sublimation temperature of InP, which fabricates a very thin insulating layer on InP. In developing this process, we demonstrate, among other effects, that the increase in the barrier height is not caused by the oxidation of the surface enhanced by the presence of ozone, but enhanced by a photoinduced electron transfer (PET) process. In the past, some researchers have considered similar devices to be enhanced metal-semiconductor Schottky diodes. Although we achieved a barrier height of 0.7 V, we present measurements of series resistance and ideality factors which question the Schottky character of these devices. Furthermore, the dramatic increase in series resistance, as the barrier increases, suggests that the gate speed for microwave devices fabricated with this technology may be less than expected because of a larger than expected resistance capacitance time constant. The instability of these devices, when exposed to air, suggest that among the oxides which make up the enhanced layer, P_2O_5 is the primary material responsible for enhancement.

A comparison of the basic transport properties of GaAs and InP yields an advantage to InP in peak and saturation velocities,[1] breakdown field, and thermal conductivity.[2] These benefits have led to encouraging device results in higher power,[3] faster speed,[4] lower noise,[5] and increased radiation hardness.[6] However, the low Schottky barrier, formed for metal-semiconductor (MES) interfaces, generates large leakage currents that eventually degrade the speed, power, and gain of MES devices. To eliminate this problem, researchers typically use a metal-insulator-semiconductor (MIS) structure using SiO_2 as the insulator. But others have demonstrated the instabilities of the SiO_2/InP interface under dc operating conditions.[7] This paper describes an ultraviolet (UV) radiation-assisted process, optimized around the congruent sublimation temperature of InP, which produces Schottky barriers up to 0.7 V. Based on series resistance and ideality factor measurements, this paper also concludes that these devices exhibit behavior more like MIS structures with a very thin insulating layer rather than Schottky diodes. In addition, the increase in series resistance, as the barrier height increases, suggests that the gate speed at microwave devices fabricated with

this technology will be lower than expected due to a larger than expected resistance-capacitance (RC) time constant

Background

Of the many researchers that have used UV-assisted growth to enhance InP's Schottky barrier,[8] Iliadis at the University of Maryland[9] has achieved the most success. He has successfully developed a room-temperature process which increased the barrier height to 0.83 V. However, his process left several questions unanswered.

In preparing his samples for exposure, Iliadis used HCl as an etch. Since HCl corrosively etches InP, the question remained whether the use of HCl represented a critical step in the enhancement process or whether a more benign etch such as $HCl:H_3PO_4$ could be used.

The design of Iliadis's apparatus allowed him to vary only the UV radiation exposure time. Therefore, questions remained concerning the influence of other parameters such as growth temperature and radiation intensity, and whether the ozone producing wavelengths are critical to the process.

Finally, the question remained as to whether this process influences the series resistance of the device. We felt that the series resistance variations would provide a clue to the true Schottky nature of these devices. If the variations indicate that the devices are MES Schottky diodes, then this would lend support to the notion that unpinning of the Fermi level is occurring at the semiconductor interface. If so, this feature would provide device designers some flexibility in choosing metals that might produce even higher Schottky barrier heights. However, if the series resistance increases at a much greater rate than expected, this would suggest not only that the devices are not MES Schottky diodes, but also that the devices would exhibit lower microwave cutoff frequencies because of a larger RC time constant.

To answer these questions, we constructed a special apparatus to provide some flexibility in growth parameters and made additional current and voltage measurements to assess the true nature of this barrier enhancement.

Preparation of Schottky Diodes

Apparatus.—A custom built work station provided flexibility in growth temperature, gas composition and flow, light intensity, and wavelength (Fig. 1). Temperature control equipment consisted of a collection of Research Inc. equipment: process controller, setpoint programmer, and phase angle controller. Additional signal conditioning equipment converted the signal from the Research Inc. equipment to the low-voltage high-current signal needed to drive the heating element located inside the process chamber.

A Corso-Gray Model D-104-B-B/SS gas handling system provided control of the gas composition and flow. The key system components included Brooks rotometers which provided a maximum flow of 4.61 pm for O_2 and N_2 and a maximum flow of 1.71 pm for H_2.

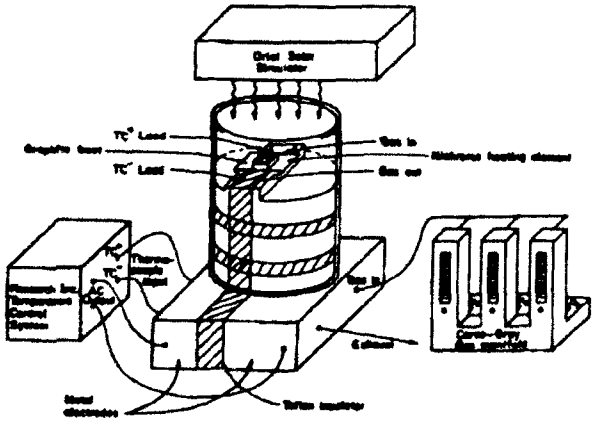


Fig. 1. Schematic of UV work station.

An Oriel Solar Simulator supplied the UV radiation. The simulator could accept lamps of three different characteristics. Although a variety of lamps can be installed in the Solar Simulator,[10] for all experiments in this paper we used the 1000 W Hg-Xe lamp. In addition, by replacing the normal mirror in the Solar Simulator with a dichoric mirror, we were able to isolate the ozone producing wavelengths from 200 to 260 nm.[11]

We designed the process chamber to handle samples up to 1 in. by 1 in. in size. A graphite boat inside the process chamber determined this upper limit in size. The graphite boat possessed a 1/6 in. tall, 0.1 in. wide lip around the edge to prevent the sample from sliding from the graphite boat during exposure. Instead of heating by RF induction, we used a resistive Nichrome metal platform to heat the graphite boat by conduction. The Nichrome heating platform was 1.6 in. by 2.0 in. and 0.005 in. thick and possessed supports which held the platform and graphite boat 3/8 in. above the entrance and exit ports for the gases. This unusual positioning of the heating platform and gas ports might be primarily responsible for the unusual flow rate effects that are discussed later in this paper.

Sample preparation.—Sample preparation began with a degrease procedure which involved the use of the soap solution Fl-70 and a DI water rinse, then an acetone rinse and ultrasonic bath, followed by a methanol rinse and ultrasonic bath. After another DI water rinse, the next step in sample preparation involved a pre-etch, to remove surface oxides, using $H_2SO_4:H_2O_2:H_2O$ (5:1:1). After a third DI water rinse, an InP etch using HCl or HCl and H_3PO_4 mixture was then performed. The Results section discusses the advantage of one solution over another. After a final DI water rinse, the samples were blown dry with nitrogen.

Fabrication of diodes.—We fabricated Schottky diodes on 1 cm by 1 cm samples cleaved from undoped InP substrates with a carrier concentration in the mid 10^{15} cm^{-3} range. After performing the wet chemical procedure described above, the samples were placed into the process chamber for irradiation under different growth conditions. The samples were held in place with a perforated metal mask, each perforation allowed for the deposition of dots which were 127 μm in diameter. This size proved practical for two reasons. The dots were small enough to make current measured low enough to prevent the saturation of the measurement equipment. Nevertheless, the dots were still large enough to make the alignment of the measurement probes easy.

Mounting of samples.—After deposition, the samples were mounted on a 3 in. by 3 in. by 1/8 in. copper plate, the top surface of the plate was coated with indium to allow for ohmic contacts to the back side of the InP samples. The actual mounting first involved heating the copper plate just enough to melt the indium but not hot enough to produce thermal damage on the InP samples. This requirement was met with using a hot plate set so as not to exceed 250°C. After the indium became molten and the samples mounted, the plate was removed from the hot plate and placed on a copper heatsink for cooling.

Determination of Barrier Height from Current/Voltage Measurements

We calculated the barrier height from the measured value of the saturation current and assumed values of temperature, diode area, and Richardson constant. The saturation current became an ideal parameter for detecting barrier enhancement. As we will show later, the barrier enhancement we expected required orders of magnitude decrease in the saturation current. Such an expected dramatic change in the saturation current gave us confidence in using the HP4145A semiconductor parameter analyzer to obtain the saturation current.

The HP4145A semiconductor parameter analyzer.—Several features made the HP4145A very useful for doing current/voltage measurements. The first important feature was the four programmable units; only two were required to characterize the diodes in these experiment. Each unit could be programmed to provide or measure voltage from 0 to 100 V and current from 1 pA to 100 mA. An extremely useful feature was the HP4145's ability to create parameters (called "user defined functions") that are constructed from mathematical operators and voltage and current variables. One HP4145 operator that we found particularly helpful was the Δ operator, which allowed the construction of a parameter that became useful in determining the series resistance; we discuss this parameter, which was based on the measurement of the differential current and differential voltage, in the next subsection. We were also able to obtain linear and logarithm plots of the user-defined functions by using the graphics routine on the HP4145A. The graphics package also contained a very useful straight line curve-fitting routine that allowed us to obtain the series resistance, ideality factor, and saturation current with few computations. Lastly, the most convenient feature of the HP4145 was the ability to store measurement configurations which eliminated the need to reprogram the HP4145 for each measurement. For this experiment, three programs were developed: one program to perform a typical forward biased *I vs. V* measurement from 0 to 0.25 V, another program to extract the series resistance from these measurements, and a third program to correct the *I vs. V* data in order to obtain the saturation current. Since the last two programs use the HP4145 in an unusual manner, the salient features of these programs are discussed in the next two subsections.

Using the HP4145 to determine series resistance.—The theoretical basis for this procedure began with an extension to the thermionic emission model for Schottky diodes to include the influence of series resistance, Eq. 1

$$I = I_o[e^{\frac{q(V-IR_s)}{\eta KT}} - 1] \quad [1]$$

Equation 2 below demonstrates the relationship between the saturation current to the barrier height

$$I_o = A^{**}T^2Ae^{\frac{-q\phi_b}{KT}} \quad [2]$$

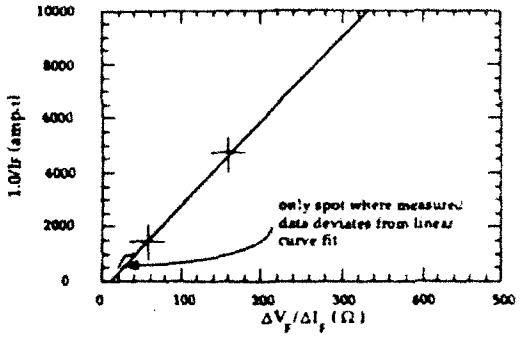
We thought a measurement scheme for the series resistance that was independent of the barrier height would be beneficial for our analysis. Hence we developed a method to eliminate the saturation current from consideration. To begin, we restricted our analysis to the forward-biased region of the diode since, except for small applied voltages (i.e., below 60 mV), the exponential term would dominate the −1 term in Eq. 1. Therefore, in this region, Eq. 1 can be written as

$$I = I_oe^{\frac{q(V-IR_s)}{\eta KT}} \quad [3]$$

Since the barrier height is contained in the expression for the saturation current (Eq. 2) and is independent of voltage, we were able to eliminate the barrier height from consideration and retain the series resistance by taking the natural logarithm of Eq. 3 and then differentiating the result with respect to I and V. After some algebraic manipulation, Eq. 4 below resulted

$$\frac{1}{I} = \frac{q}{\eta KT}\left(\frac{dV}{dI} - R_s\right) \quad [4]$$

Equation 4 is not only independent of the barrier height but also matches the straight line equation $y = m(x - b)$; where y equals $1/I$, x equals dV/dI, m relates to the ideality factor, and most important, b, the X intercept, gives the series resistance. Equation 4 was programmed into the HP4145 using the user-defined function feature to define the x and y variables. In particular, the definition of the x variable made use of the Δ operator to obtain dV and dI. Figure 2



	GRAD	1/GRAD	Xintercept	Yintercept
LINE1	35.8E+00	28.1E-03	17.0E+00	-604E+00

Fig. 2. HP4145 plot used to obtain series resistance. Region between crosses marks the range of measured data used to perform the linear curve fit.

demonstrates how well the model matched data from an actual diode. The excellent fit to a straight line made the determination of the series resistance easy. For the example, in Fig. 2, the X intercept from the curve fitting routine gave a series resistance of 17 Ω. The inverse slope, 1/GRAD, related to an ideality factor of 1.09, which we obtained by dividing 1/GRAD by the room temperature value of KT/q (25.8E-03).

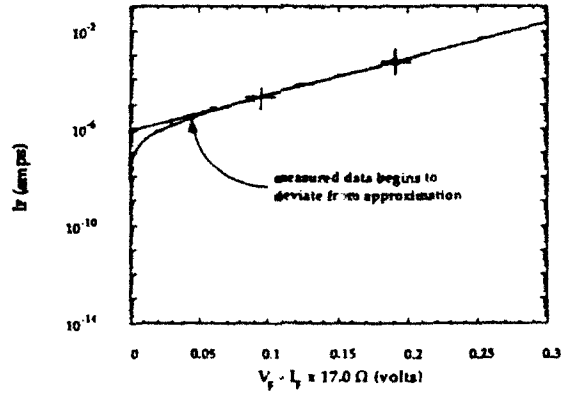
Using the HP4145 to determine saturation current.—Before extracting the saturation current from the measurements, we programmed the HP4145A to remove the effects of series resistance. The user-defined function feature allowed us to subtract $I \cdot R_s$ from the measured voltage (Eq. 5)

$$V_D = V - I \cdot R_s \quad [5]$$

With this adjustment, a slightly modified version of Eq. 3 resulted (Eq. 6)

$$I = I_oe^{\frac{qV_D}{\eta KT}} \quad [6]$$

Although this change is small, a plot of the $\log(I)$ *vs.* V_D demonstrates a significant benefit (Fig. 3). While a plot not corrected for series resistance would maintain some curva-



	GRAD	1/GRAD	Xintercept	Yintercept
LINE1	15.6E+00	64.1E-03	338E-03	896E-09

Fig. 3. HP4145 plot used to obtain saturation current. Region between crosses marks the range of measured data used to perform the linear curve fit.

ture throughout the forward-biased region, the corrected plot shown in Fig 3 exhibited straight line behavior above 60 mV Below 60 mV, the exponential term no longer dominates the diode characteristics, instead, the more precise Eq 1 describes the performance in this region The logarithm of Eq 6 not only predicts the linear behavior but also provides a means of determining the saturation current (Eq 7)

$$\log(I) = \log(I_o) + 0.4343 \frac{q}{\eta KT} V_D \quad [7]$$

Equation 7 matches a slightly different linear equation than the one used to obtain the series resistance Rather than matching the form $y = m(x - b)$ Eq 7 matches the form $y = mx + b$, where y equals $\log(I)$, x equals V_D, m relates to the ideality factor, and most important, b, the Y intercept relates to the saturation current, the saturation current is actually the antilog of the Y intercept For the example in Fig 3, which has a series resistance of 17 Ω, a saturation current of 0 896 μA is obtained

Computation of barrier height —Once the saturation current is known, a rearranged version of Eq 2 was obtained to obtain the barrier height (Eq 8)

$$\Phi_B = \frac{KT}{q} \ln\left(\frac{A^{**}T^2}{I_o}\right) \quad [8]$$

The actual computation for the barrier height was performed using the spread sheet Excel for the MacIntosh For the value of the saturation current obtained earlier, a typical InP barrier height of 0 48 V was obtained

Results and Discussion

Before starting this work, a barrier enhancement of 0 8 V was established as a target With this in mind, we used Eq 8 to estimate the change in the saturation current to obtain this level of barrier enhancement The result of this analysis predicted a drop of 10^5 This expected large drop reinforces the benefit of using the saturation current as the indicator of barrier enhancement

To demonstrate to ourselves that the UV radiation might have a dramatic effect on an InP surface, we performed a series of experiments at a relatively high temperature InP substrates were exposed to a 700°C environment for 2 min in three different atmospheres H_2, N_2, and O_2 For comparison, some samples were additionally exposed to radiation from a 1 kW Hg-Xe ozone free lamp Since the ambient temperature is much larger than the congruent sublimation temperature of InP, we expected severe erosion of the InP surface regardless of the ambient However due to the short exposure time, only samples exposed to the H ambient showed any noticeable erosion Samples exposed to the N_2 ambient regardless of UV exposure illustrated no surface damage, while the samples exposed to the O_2 ambient exhibited a significant oxide growth for the sample exposed to UV radiation Possible explanations for these observations are discussed in the next few paragraphs

Figure 4 shows the results of a sample exposed to an H_2 ambient without UV radiation The severe erosion was expected but, as supported by data presented later, this erosion was not caused by the sublimation of phosphorus as we initially thought, but more likely caused by a reaction between the phosphorus just above the substrate and H As shown with Eq 9 and 10, this reaction most likely leads to formation of phosphine As the PH, in Eq 10 is swept from the process chamber, Eq 9 will continue in the forward direction, thus siphoning more phosphorus from the InP substrate and leaving indium droplets shown in Fig 6 on the surface

$$InP(s) \rightarrow In(s) + \frac{1}{2} P_2(g) \quad [9]$$

$$\frac{1}{2} P_2(g) + \frac{3}{2} H_2(g) \rightarrow PH_3(g) \quad [10]$$

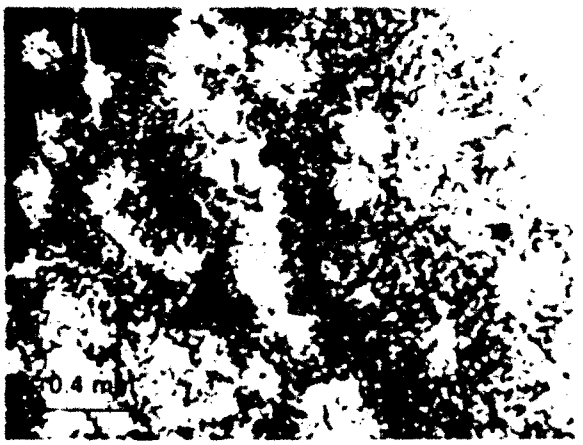


Fig 4 H_2 without UV radiation Exposed for 2 min at 700 C Nomarski photomicrograph taken at 50 times magnification Exposure produced expected deterioration of the surface

Figure 5 illustrates the outcome of another sample exposed to the same conditions as the first sample except for the addition of UV radiation from a 1 kW Hg Xe ozone free lamp The radiation seems to make the erosion of the surface more uniform This led us to believe that the UV radiation contributed significantly to the surface reactions occurring on the InP substrate

We next examined the behavior of the InP in a N_2 atmosphere If indeed H played little role in the thermal erosion of the InP surface we would observe the same damage as that illustrated in Fig 4 and 5 Surprisingly as Fig 6 and 7 show, no erosion occurred for a sample placed in an N_2 ambient regardless of the presence of UV radiation Hence these observations support the notion that while N_2 acts as an inert gas to InP, H plays a very active role in the thermal erosion of InP This reasoning is lent further support by the observations of Greene and Truth

Next we investigated the behavior of these samples in an O_2 atmosphere A comparison of Fig 8 and 9 leads to an unusual observation Without the use of UV radiation, as shown in Fig 8, the surface is well preserved suggesting that, for these short exposure times O_2 acts more like an inert gas However with the addition of the UV radiation (Fig 9) a thin film is produced on the surface which Auger measurements revealed as composed of In and O Fig 10 thus suggesting the formation of the oxide In_2O_3 This scenario is possible since the phosphorus liberated is probably too volatile to participate in any surface reactions Using the index of refraction for In_2O_3 ellipsometer measure

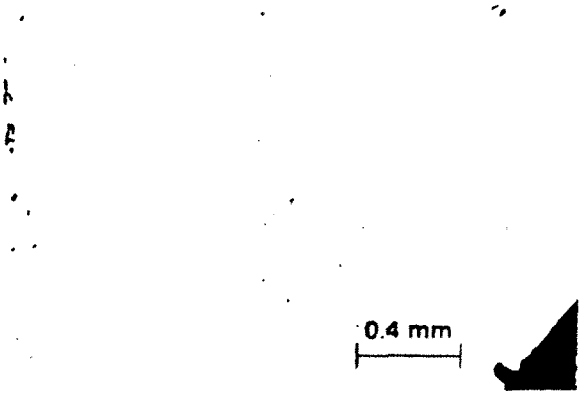


Fig 5 H_2 with UV radiation Exposed for 2 min at 700 C Nomarski photomicrograph taken at 50 times magnification Surface deterioration was expected but uniformity of damage was surprising

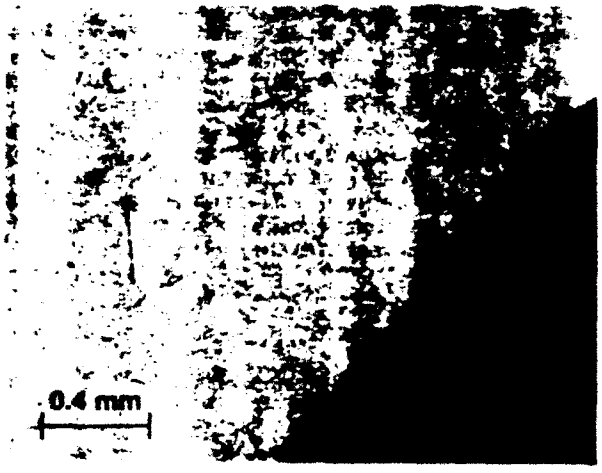


Fig. 6. N_2 without UV radiation. Exposed for 2 min at 700°C. Nomarski photomicrograph taken at 50 times magnification. Preservation of surface was a surprise.

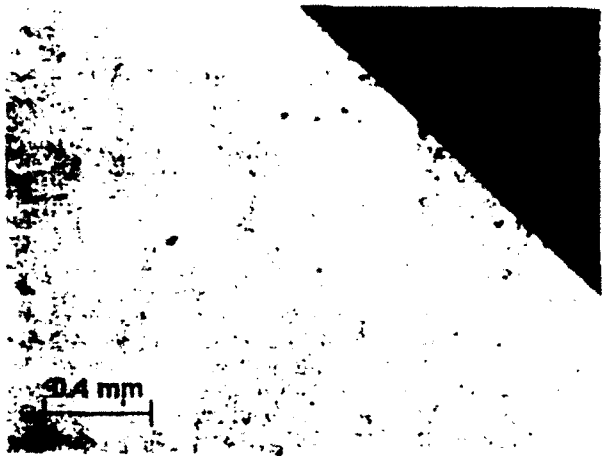


Fig. 7. N_2 with UV radiation. Exposed for 2 min at 700°C. Nomarski photomicrograph taken at 50 times magnification. Preservation of surface was a surprise.

ments were made to estimate the thickness at 500 Å. Since the ozone-free UV lamp was used for this process, the reaction of ozone with the InP surface cannot be responsible for this reaction. Instead, in addition to the catalyst role noticed in the reaction with H_2, a photoinduced electron transfer (PET) process is more likely responsible for producing O_2^- molecules which then participate in the oxidation of the surface to produce indium oxide.

We believe that there are two possible PET reactions that result in excess electrons that convert the InP surface into an oxidizing agent which, due to O_2 relatively high electron affinity (0.45 eV), reacts easily with oxygen to eventually produce an indium oxide film.

The first possible PET reaction, described by Channon and Eberson,[13] relies on the generation of excess electrons in the conduction band of *n*-type by photostimulation of electrons from the valence band and into the conduction band, resulting in bandbending at the surface to reflect the increase concentration of electrons and holes. Equation 11 illustrates how the activated surface converts (S.C.) O_2 to O_2^-

$$S.C. + h\nu \rightarrow S.C.^* + O_2 \rightarrow S.C.^+ + O_2^- \qquad [11]$$

The second possible reaction, described by Fox[14] and illustrated in Eq. 12, generates excess electrons by photostimulation of surface atoms (S.A.)

$$S.A. + h\nu \rightarrow S.A.^* + O_2 \rightarrow S.A.^+ + O_2^- \qquad [12]$$

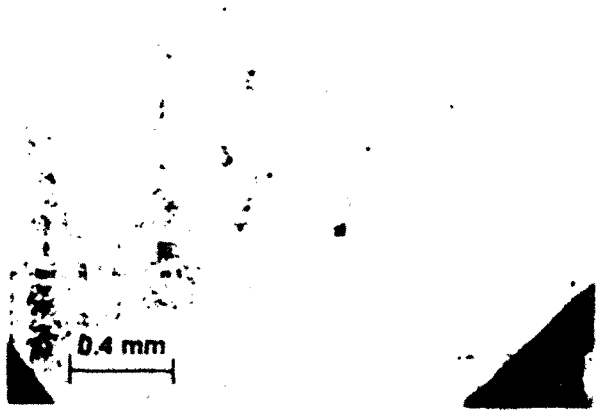


Fig. 8. O_2 without UV radiation. Exposed for 2 min at 700°C. Nomarski photomicrograph taken at 50 times magnification. Preservation of surface was a surprise.

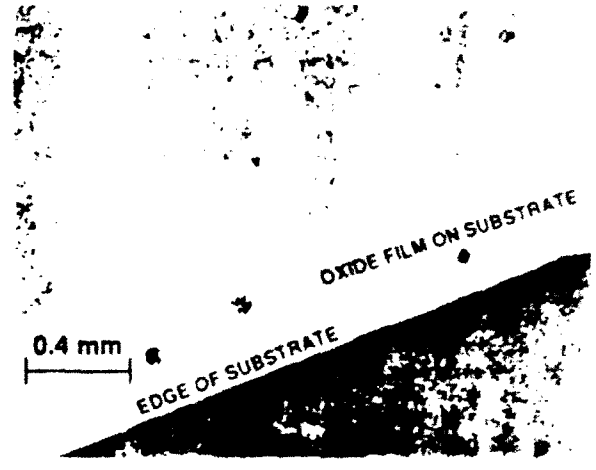


Fig. 9. O_2 with UV radiation. Exposed for 2 min at 700°C. Nomarski photomicrograph taken at 50 times magnification. Most surprising was the formation of thin brownish film, indicated by the abrupt change in shade just before the blemishes on the surface, which covered most of the surface.

With indium as the surface atom for the second type of PET reaction, Eq. 13 and 14 illustrate the possible reaction and

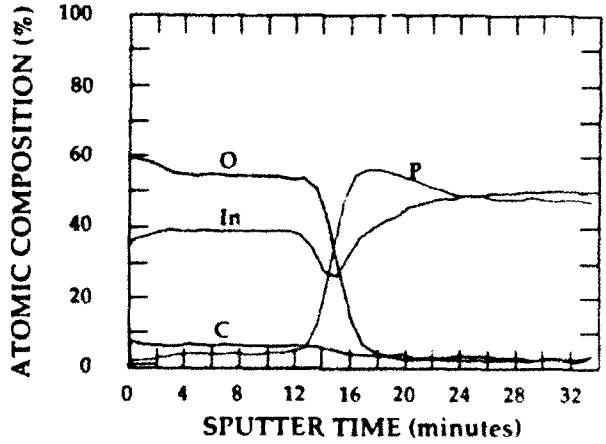


Fig. 10. Auger depth profile. In, P, and O are relatively strong signals. The C signal is within the background noise level of the instrument.

the activation energies based on electrode potential energies[15]

$$In \rightarrow In^{+} + e^{-} \qquad E_a = 0.14\ eV \qquad [13]$$
$$e^{-} + O_2 \rightarrow O_2^{-}$$

$$In \rightarrow In^{3+} + 3e^{-} \qquad E_a = 0.34\ eV \qquad [14]$$
$$3e^{-} + 3O_2 \rightarrow 3O_2^{-}$$

Because the energy for UV photons is above 3.0 eV and the energy gap of InP is 1.35 eV, both mechanisms and both ionization reactions most likely occur to produce radical O_2^-.

Once the radical species is produced, one possible reaction to produce indium oxide is shown in Eq. 15

$$In^{+} + O_2^{-} \rightarrow \frac{1}{2} In_2O_3 + \frac{1}{4} O_2 \qquad [15]$$

Hence, thermodynamic arguments can be made to support the claim that ozone is not primarily responsible for the layer formation but instead oxidation of a surface enhanced by the photoinduced transfer of electrons.

After the high-temperature experiment, we attempted to reproduce the room-temperature Schottky enhancement achieved by Iliadis. However, we met with little success. Less than 1% of the samples processed demonstrated any evidence of barrier enhancement. Furthermore, for those samples exhibiting enhancement, the effect was very localized, at most three diodes from an array of 144 exhibited any enhancement. At this point, the idea of optimizing the process at the congruent sublimation temperature became more appealing. In pursuing this line of investigation, the importance of other parameters, which were needed to achieve the greatest possible enhancement, became clear.

In presenting our results, several effects are presented on the same table. This was not done to suggest any connection between the parameters, although future studies might reveal such relationship, but instead, to present the results in a more compact form and demonstrate how the experiments proceed chronologically.

The first set of experiments involved analysis of temperature and wavelength (Table I). Since the dichoric mirror was still in place from the room-temperature experiments, the first three experiments only used the ozone producing wavelengths. Three different temperatures were examined, with no enhancement occurring. Next, the dichoric mirror was replaced with the Solar Simulator's normal mirror, and the temperature ranges repeated. This time enhancement occurred at 368°C. The results of these experiments gave the first indication that the congruent sublimation temperature was a critical parameter for this process. The results also indicated that the ozone producing wavelength had little influence on the enhancement process.

The sensitivity of this process to the congruent sublimation temperature appears to make sense. Since the enhanced layer most likely contains insulating phosphorus oxides, either $InPO_4$ or P_2O_5 or both,[15] and because the substrate is the only source of phosphorus, the ability to trap, into the enhancement layer, the phosphorus being liberated from the substrate becomes important. Unlike the high volatility at elevated temperatures, the phosphorus liberated at the congruent sublimation temperature should be less mobile, thus allowing the phosphorus to interact with the O_2^- radical produced by a photoinduced electron transfer (PET) effect. The observed PET effect at this temperature is consistent with the high-temperature results discussed earlier in this paper. Equations 16 and 17 describe the initial reactions which might involve the radical O_2^-. The surface atom (S.A.) is most likely indium. As a result, Eq. 18, 19, and 20 describe possible terminal reactions which produce the surface oxides. Note that the ionized surface sites may play an important role in attracting the phosphorus oxides back to the surface, since electrostatic attraction between surface sites and the gas molecules may be strong enough to prevent phosphorus oxides from being swept from the surface.

$$S.C. + h\nu \rightarrow S.C.^{*} + O_2 \rightarrow S.C.^{+} + O_2^{-} \qquad [16]$$

$$S.A. + h\nu \rightarrow S.A.^{*} + O_2 \rightarrow S.A.^{+} + O_2^{-} \qquad [17]$$

$$In^{+} + O_2^{-} \rightarrow \frac{1}{2} In_2O_3 + \frac{1}{4} O_2 \qquad [18]$$

$$In^{3+}(s) + In^{+}(s) + P_2(g) + 4\ O_2(g) \rightarrow 2InPO_4(s) \qquad [19]$$

$$5S.C.^{+}(s) + P_2(g) + 5\ O_2(g) \rightarrow 2P_2O_5(s) + 5S.C. \qquad [20]$$

After the success of the first set of experiments, we next addressed the questions of whether the etch solution HCl was critical to the process and whether there was an upper limit to the exposure time. As mentioned in the previous chapter, HCl corrosively etches InP. From a device processing point of view, the use of a more benign and controllable etch becomes important. The mixture of HCl and H_3PO_4 appeared the best choice, since this mixture not only produces an excellent morphology but also exhibits an adjustable etch rate based on the portion of H_3PO_4. By adding larger portions of H_3PO_4 to the mixture, the etch rate could be decreased from 12 to 0.5 μm/min.[17] For our experiments, a 1:4 ($HCl:H_3PO_4$) mixture was used, which produced a modest etch rate of 1 μm/min. An investigation of the effects of exposure time was conducted since Iliadis demonstrated in his work that the barrier height saturated at an exposure time of 40 min. Since our experiments used a higher intensity (100 mW/cm² vs. 15 mW/cm²), we expected to observe a similar effect within 6 min of exposure time.

Table II summarizes the results of the second set of experiments, which demonstrate that the $HCl:H_3PO_4$ etch is a suitable substitute for HCl and the barrier height saturates at 5 min. The result of the barrier-height saturation time seems to be consistent with Iliadis's results, thus suggesting that an energy density limit might exist for this enhancement process. We believe this is most likely linked to the penetration depth of the UV radiation into the substrate during the PET process. Since the InP extinction coefficient in the UV range is at least an order of magnitude greater than the extinction coefficient in any wavelength region emitted from the Solar Simulator,[18] UV-driven reactions would be limited to the few monolayers close to the surface. Therefore, the number of activated sites in Iliadis's and our experiment would be approximately the same; hence, the shorter saturation time we observed would be consistent with the higher irradiance available with our apparatus.

Having identified a suitable etch solution and exposure time, we returned to refining the temperature effect. Table III illustrates the results of these experiments. Significant barrier enhancement occurs from 350 to 380°C. These results reaffirm the dynamics of the process discussed earlier. In addition, the highest barrier height, 0.69 V, gave indications of a flow-rate effect since that par-

Table I. Wavelength and temperature effects.

	204-211°C	368°C	628°C
Ozone	0.47 V	0.47 V	0.47 V
Full	0.47 V	0.55 V	0.47 V

Sample preparation: HCl etch. Growth parameters: flow rate = 1540 sccm, intensity = 98 mW/cm², exposure time = 2 min.

Table II. Etch solution and time effects.

Temp (°C)	Etch	Time (min)	Barrier height (V)
368	HCl	2	0.55
366	$HCl:H_3PO_4$	5	0.59
366	$HCl:H_3PO_4$	30	0.60

Growth parameters: flow rate = 1540 sccm, intensity = 98 mW/cm², wavelength = full spectrum

Table III. Refinement of temperature effect.

Temp (C)	Barrier height (V)
394	0 47
380	0 59
366	0 59
353	0 51
348	0 69

Sample preparation etch: $HCl:H_3PO_4$. Growth parameters intensity = 98 mW/cm², exposure time = 5 min, wavelength = full spectrum, flow rate = 1540 sccm (last sample in Table processed at flow rate of 614 sccm).

Table IV. Refinement of flow rate effect.

Flow (sccm)	Percent full scale	Barrier height (V)
1540	33 3	0 51
920	20.1	0 50
614	13.3	0 67
307	6 7	0 48
0	0.0	0 49

Sample preparation etch: $HCl:H_3PO_4$. Growth parameters intensity = 98 mW/cm², exposure time = 5 min, wavelength = full spectrum, temperature = 348°C.

ticular experiment was performed at the lower flow rate than the other samples, 614 sccm.

The results at 348°C prompted another set of experiments (Table IV) to isolate the proper flow rate. The earlier conditions were repeated and produced a barrier height of 0.67 V. Other experiments were performed at flows higher and lower than 614 sccm. However, none of these experiments resulted in a barrier height greater than that achieved at 614 sccm. We believe one of two reasons could be used to explain this unusual result. One reason could be that the kinetics of the process may require the contribution of unreacted O_2 at a particular speed to enhance an intermediate chemical reaction. Another reason, as discussed earlier, could be an unusual flow pattern influence by the layout of the process chamber. Further study is needed to identify a plausible reason.

The sample with the highest barrier in the previous set of experiments was also studied to determine whether the barrier height remained stable in air. Table V illustrates the results of these measurements. Unfortunately, we measured a decay of the barrier height to 0.57 V within 48 h, after which the barrier decayed to 0.55 V in 45 days. We believe this decay was caused by the reaction of P_2O_5 with water vapor in the atmosphere since this oxide is one of the most efficient drying agents used in desiccants (0.5 grams of water removed per gram of P_2O_5).[19] To solve this problem, a process to encapsulate and/or anneal these devices will be necessary to guarantee their long-term stability. Thus what initially appeared as a simple process to enhance InP's Schottky barrier is becoming more complex.

We thought a close examination of the changes in the series resistance and ideality factor would reveal whether these devices are MES Schottky diodes. For MES Schottky diodes, we believe that at best, the series resistance should remain constant as the barrier height increases, and at worst, the resistance should relate to the length of the depletion region generated by the Schottky barrier. The theoretical model we used for the latter case views the depletion region as a linear resistor. From this standpoint, we derived Eq. 21

Table V. Stability of highest barrier.

Time	Barrier height (V)
Initial	0.67
48 h	0.57
72 h	0.57
45 days	0.55

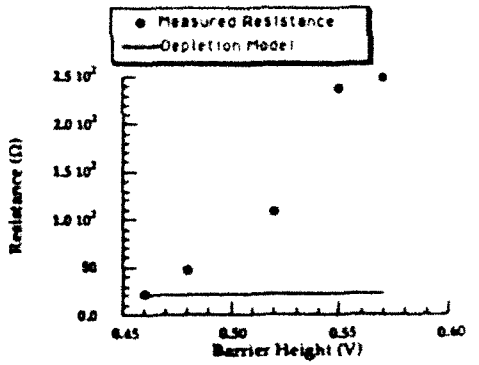


Fig. 11. Resistance comparison used to determine Schottky character of diodes.

$$R_{dep} \approx \frac{\rho}{A}\sqrt{\frac{2\varepsilon}{qN_D}\phi_B} \qquad [21]$$

The approximation in Eq. 21 reflects the fact that the expression ignores the few millielectron volts difference between the bottom of the conduction band and the Fermi level within the neutral region of the semiconductor material. These differences are typically an order of magnitude less than the barrier height and as such have little influence on the analysis of the experimental results.

Since the resistivity of the depletion region is difficult to obtain, we developed an expression for resistance which eliminated the resistivity by considering the percent increase in the enhanced resistance from the resistance of a normal Schottky diode (see Eq. 22)

$$\%\ \text{increase} = \frac{\sqrt{\phi B_{enhanced}} - \sqrt{\phi B_{normal}}}{\sqrt{\phi B_{normal}}} \times 100 \qquad [22]$$

For the size of the Schottky diodes used in this experiment, the resistance of a normal Schottky diode, which has a barrier height of 0.45 V, is 21.1 Ω; as a result, the expected resistance of an enhanced layer is expressed by Eq. 23

$$R_{dep} = \frac{\sqrt{\phi B_{enhanced}} - \sqrt{0.45\ \text{V}}}{\sqrt{0.45\ \text{V}}} \times 21.1\ \Omega + 21.1\ \Omega \qquad [23]$$

Figure 11 demonstrates how, even for the lower barrier heights, the measured series resistance varies drastically from the expected variation. For the higher barrier heights, the discrepancy is much worse. We believe these results indicate that a thin insulating barrier is being formed between the diode metal gate and the semiconducting surface as the barrier is increased. This would strongly suggest the formation of a MIS structure rather than a MES Schottky diode. The evidence becomes more compelling in this direction if the variation in the ideality factor is also considered. If the enhancement represented a MES Schottky diode, the ideality factor would remain close to 1.00. But, an examination of Fig. 12 shows that this is not the case. Instead the ideality factor increases as the barrier height increases, which is an indication that the diode characteristics are moving further and further away from ideal behavior. Since the variations in the series resistance and ideality factor indicate that these devices are not MES-enhanced Schottky barriers, the expected unpinning of the Fermi level is not a by-product of our enhancement process. Furthermore, for gate regions fabricated from this technology, we would expect lower cutoff frequencies due to larger than expected RC time constants.

Conclusions

A number of important observations were made with this set of experiments. To begin, we established that the HCl

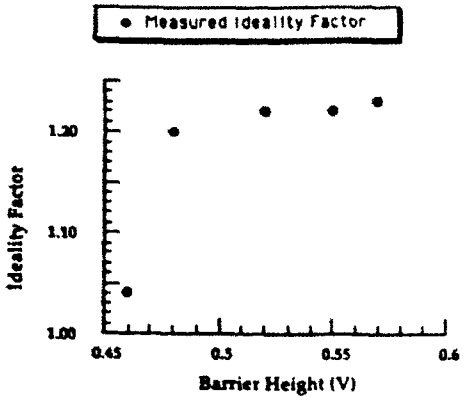


Fig. 12. Ideality factor comparison used to determine Schottky character of diodes.

etch is not critical to the enhancement process. As a result, we were able to replace this etch with the more benign mixture of $HCl:H_3PO_4$. The optimal growth parameters include growth temperatures in the 340 to 380°C range, a flow rate of $O_2 \approx 600$ sccm, a saturation exposure time of 5 min. We also determined that the ozone producing wavelengths are not critical to the process. Finally, the device characteristics indicate that the barrier height is susceptible to the water vapor in the air, and the variation of the series resistance and ideality factor as the barrier height increases suggest that the devices are not MES Schottky diodes.

Although not MES Schottky diodes, the advantage of a lower saturation current would be beneficial for a number of device applications. For instance, this thin insulating layer could be used as an intermediate layer between the InP surface and a SiO_2 layer in order to enhance the stability of SiO_2 MIS devices. Therefore, future work will continue in order to answer the questions raised by the experiments in this paper and to alleviate the susceptibility of the devices to water vapor.

Acknowledgments

We would like to thank Saied Tadayon for the useful conversation concerning the use of the HP4145A to measure the series resistance measurements of a diode. Financial support was provided by the Joint Services Electronics Program under Contract No. F49620-87-C-0044.

Manuscript submitted May 17, 1991; revised manuscript received May 27, 1992.

Cornell University assisted in meeting the publication costs of this article.

LIST OF SYMBOLS

A	area of Schottky diodes, 1.27×10^{-4} cm^2 for diodes used in this paper
A^{**}	Richardson's constant, 9.2 $A/cm^2 - K^2$ for InP
I	current, A
I_o	saturation current, A
K	Boltzmann's constant, 1.38×10^{-23} J/K
N_D	carrier concentration of substrate, cm^{-3}
q	electronic charge, 1.602×10^{-19} C
R_{dep}	equivalent resistance of depletion region, Ω
R_s	series resistance, Ω
S A	surface atom involved in PET process
S C	n-type semiconductor material involved in PET process
T	temperature, K
V	voltage across diode and series resistance, V
V_D	voltage across the diode, V
ε	permittivity, F/cm
η	ideality factor
ρ	resistivity, Ω-cm
ϕ_b	Schottky barrier height, V
$\phi_{enhanced}$	barrier height of enhanced Schottky junction, V
ϕ_{normal}	barrier height of normal Schottky junction, V

REFERENCES

1. H. H. Wieder, *J. Vac. Sci. Technol.*, **17**, 1009 (1980).
2. M. Armand, D. V. Bui, J. Chevrier, and N. J. Linh, *Electron Lett.*, **19**, 433 (1983).
3. L. Messick, D. A. Collins, R. Nguyen, A. R. Clawson, and G. E. McWilliams, *IEDM Tech. Dig.*, 767 (1986).
4. A. Antreasyan, P. A. Garbinski, V. D. Mattera, Jr., H. Temkin, and J. H. Abeles, *Appl. Phys. Lett.*, **51**, 1097 (1987).
5. R. M. Corlett, I. Griffith, and J. J. Purcell, in *Proceedings of 5th European Microwave Conference*, pp. 695-698, Hamburg (1975).
6. I. Weinberg, C. K. Swartz, R. E. Hart, Jr., and M. Yamaguchi, in *Proceedings of 5th European Symposium. Photovoltaic Generators in Space*, Scheveningen, The Netherlands, Sept. 30-Oct. 2, 1986 CESA SP-267, pp. 415-420 (1986).
7. D. L. Lile and M. J. Taylor, *J. Appl. Phys.*, **54**, 260 (1983).
8. S. Loualiche, A. Ginoudi, H. L'Haridon, M. Salvi, A. LeCorre, D. Lecroshier, and P. N. Favennce, *Appl. Phys. Lett.*, **54**, 1238 (1989).
9. A. A. Iliadis, *Inst. Phys. Conf. Ser. No. 96*, pp. 413-417, paper presented at International Symposium on GaAs and Related Compounds, *Atlanta, GA (1988)*.
10. *Oriel Corporation Light Sources Monochromators and Detection Systems Catalog*, Vol. II, pp. 83-84, Oriel Corp. (1989).
11. Oriel Corporation Advanced Exposure Group, Private communication.
12. P. D. Greene and E. J. Thrush, *J. Crystal Growth*, **72**, 563 (1985).
13. M. Chanon and L. Eberson, *Photoinduced Electron Transfer: Part A. Conceptual Basis*, pp. 409-597, Elsevier Science Pub. Co. Inc., New York (1988).
14. M. A. Fox, *Photoinduced Electron Transfer. Part D Photoinduced Electron Transfer Reactions: Inorganic Substrates and Applications*, pp. 1-27, Elsevier Sci. Pub. Co. Inc., New York (1988).
15. P. W. Atkins, *Physical Chemistry*, 3rd ed., p. 825, W. H. Freeman and Co., Boston (1986).
16. *Physics and Chemistry of III-V Compound Semiconductor Interfaces*, C. W. Wilmsen, Editor, p. 182, Plenum Press, New York (1985).
17. S. E. H. Turley and P. D. Greene, *J. Crystal Growth*, **58**, 409 (1982).
18. *Handbook of Optical Constants of Solids*, E. D. Palik, Editor, pp. 509-510, Academic Press, Inc., New York (1985).
19. G. J. Shugar and J. A. Dean, *The Chemist's Ready Reference Handbook*, p. 254, McGraw-Hill Inc., New York (1990).

Accepted for publication in Applied Physics Letters,
February 1993.

GAS PHASE REACTIONS OF TRIMETHYLAMINE ALANE IN LOW PRESSURE ORGANOMETALLIC VAPOR PHASE EPITAXY OF AlGaAs

B.L. Pitts, D.T. Emerson and J.R. Shealy
OMVPE Facility, School of Electrical Engineering
Cornell University
Ithaca, N.Y. 14853

Abstract

We have investigated the effects of gas phase reactions between trimethylamine alane (TMAA), triethylgallium (TEG) and arsine on $Al_xGa_{1-x}As$ films grown by low pressure Organometallic Vapor Phase Epitaxy. The reactor used in this study provides for independent observation of the effects of TEG-TMAA and TMAA-arsine gas phase reactions. Gas phase reactions involving TMAA and TEG result in the formation of nonvolatile compounds upstream, which condense on the reactor wall, resulting in a reduction of growth rate and a degradation of the deposition uniformity. The TMAA-arsine reaction produces a compositional dependence on the gas phase stoichiometry (V/III ratio). Both of these effects are more severe for higher TMAA fluxes. High quality AlGaAs with excellent thickness and compositional uniformity was produced by spatially separating the TMAA and TEG in the gas phase which minimizes the parasitic reactions.

The growth of aluminum containing III-V compound semiconductors grown by Organometallic Vapor Phase Epitaxy (OMVPE) has traditionally been plagued with high oxygen and carbon incorporation. A major reason for these problems is due to the widely used aluminum source, trimethylaluminum (TMA). TMA has a strong aluminum-carbon bond and the ability to form volatile aluminum alkoxide compounds resulting in oxygen contaminated AlGaAs layers.[1] Recently, trimethylamine alane (TMAA) has received much attention as a viable alternative aluminum source in both OMVPE and Chemical Beam Epitaxy (CBE).[2-6] Previous reports indicate that using TMAA along with triethylgallium (TEG) and arsine (AsH_3), under the appropriate growth conditions (very high V/III ratios and gas velocities), can result in the highest purity OMVPE grown AlGaAs.[2,6] This is believed to be due to a lack of direct aluminum-carbon bond in TMAA and also its ability to form involatile Al–O compounds when reacted with oxygen and H_2O, resulting in reduced oxygen contamination. Improved photoluminescence (reduced donor-to-acceptor related transition) and mobilities (77 K mobility exceeding 14,000 $cm^2/V \cdot sec$ for $Al_{0.14}Ga_{0.86}As$) have been achieved.[2]

Earlier reports using TMAA in OMVPE suggest that a requisite for producing high quality AlGaAs epitaxial layers is to avoid prereactions between TEG and TMAA upstream from the susceptor.[2,7] TMAA has a low thermal decomposition temperature (~100 °C), allowing predeposition on the side walls of the reaction cell. A solution to these problems has been to increase the gas velocity which reduces the residence time of the reactants in the growth chamber. In addition, high V/III ratios are necessary to achieve high purity results. The growth chemistry using these precursors in OMVPE must be understood in order to optimize film quality. Although studies investigating the effects of gas-phase reaction between TMAA and TEG in CBE have been reported,[8,9] no previous study exists for OMVPE.

An investigation of gas phase reactions involving TMAA in low pressure OMVPE of AlGaAs is reported. We have observed two predominant effects: one

due to a TMAA-AsH_3 reaction yields a strong influence of film composition with the V/III ratio, and the other resulting from a TMAA-TEG reaction which degrades the deposition uniformity. The effects of each of these gas phase reactions in the upstream portion of the reaction cell were identified by spatially separating TMAA and TEG in the gas phase using a multichamber reaction cell.[10] The TMAA-TEG reaction has severe effects on the quality of the AlGaAs films especially at low V/III ratios. Using the separated TMAA and TEG reactant flux approach, high quality AlGaAs structures were produced by at much lower V/III ratios than have been previously reported.

AlGaAs layers were grown using Flow Modulation Epitaxy (FME)[11] at low pressure in a vertical barrel, multichamber OMVPE system,[10] illustrated schematically in Figure 1a. In this system substrates are rotated through group III rich spatially separated zones in a uniform group V background flux without valve switching. An inner quartz ampoule (diameter-*d*) separates the reactant fluxes of each deposition zone. Figure 1b shows the flow modulation exposure cycle for each growth mode. In the conventional growth mode, the TEG and TMAA are premixed prior to injection into the reaction cell while the susceptor is rotated at 0.1 rev/sec. In the spatially separated growth mode, the TEG and TMAA are injected into separate growth zones, minimizing the TEG-TMAA gas phase reactions. For the group III flux used in this study, susceptor rotation speeds greater than 1 rev/sec are needed to produce sub-monolayer exposure cycles which result in mixed alloys. Rotation speeds ranging from 0.1 to 0.7 rev/sec were used when the reactant fluxes were separated in the vapor. Raman spectroscopy confirmed the existence of short period superlattices (confined LO GaAs and AlAs vibrations) on all samples produced with this method. The degree of deposition zone separation (indicated by the set of arrows in Figure 1b) is proportional to the amount of hydrogen carrier gas injected between each zone. Because a small amount of zone intermixing occurs in the spatially separated growth mode (see Figure 1b), the short period superlattices have graded interfaces. In both growth

schemes, the total gas flow was 30 slm, while the gas velocity was maintained at 30 cm/s.

Undoped AlGaAs layers were grown using TMAA, adduct-purified TEG and 100% AsH_3. Layer thicknesses ranged from 0.5-2 μm. The substrates, (100) Si-doped n^+ GaAs and (100) semi-insulating GaAs, were first rinsed in organic solvents and then etched in $5H_2SO_4$:$1H_2O_2$:$1H_2O$ prior to growth. The TMAA was held at 23 °C while a H_2 flow of 57 sccm was passed through the bubbler. The TEG was also held at 23 °C, while the flow varied from 18 to 50 sccm. Both TEG and TMAA were maintained at 100 torr. The growth temperature varied from 635 to 750 °C, and the reactor cell pressure was 76 torr. AlGaAs films were characterized by Hall measurements, Raman spectroscopy and photoluminescence (PL). Low temperature (1 K) PL was carried out with samples submerged in superfluid He with photoexcitation provided by the 514.5 nm line of an Ar^+ laser. Raman spectroscopy was used to determine the aluminum composition[12] and the structure features of the superlattices.[13] Thickness measurements were made by a combination of angle bevelling and staining and from analysis of reflectance spectra.

The V/III ratio and growth temperature criteria for good surface morphology were investigated over the range from 635 to 750 °C. Good surface morphology was realized for a V/III ratio as low as unity over the entire temperature range. All layers were n-type and net carrier concentrations were in the low 10^{15} cm^{-3} range. With growth temperature (670 °C) and group III flux constant, the Al mole fraction as determined by Raman scattering[12] was found to vary with V/III ratio in the conventional premixed growth mode. As shown in Figure 2, more Al is incorporated in the film as the V/III ratio is decreased. For low TMAA fluxes, corresponding to alloy compositions of $\approx$15%, AsH_3 appears to prevent the TEG-TMAA reaction which is shown to reduce the TEG tranport to the growth surface. As can be seen in the PL spectra in the Figure 4 inset, the sample quality degrades with decreasing V/III ratio. At a V/III ratio of 80,

a full width at half maximum exciton linewidth of 2.2 meV was observed for $Al_{0.15}Ga_{0.85}As$. This compares favorably with the narrowest linewidth ever reported at that composition by Reynolds *et al.*[14] The exciton line broadened but was still clearly identifiable when the V/III ratio was lowered to 7.5. Finally, at a V/III ratio of 1, the exciton feature was absent. The need for large arsine flows may imply that the TMAA-AsH_3 reaction inhibits the TMAA-TEG reaction which is demonstrated to severely degrade the quality of the AlGaAs films.

Gas phase reactions between TEG and TMAA have major effects on the growth rate. As Figure 3 illustrates, when the TMAA and TEG are premixed, the AlGaAs growth rate is approximately half that of GaAs with same TEG reactant flux. A relatively high V/III ratio was used (V/III=80) to eliminate the effects of AsH_3 flows described earlier. The Al composition for AlGaAs grown using premixed sources was 79% whereas that for the spatially separated sources was nominally 40%. Assuming that the Al and Ga incorporation in the AlGaAs layer in the two growth modes are equal, an estimated 70% of the TMAA reacts in the gas phase to produce nonvolatile compounds. The effect of growth rate reduction was also reported for CBE for premixed TMAA and TEG.[8] In addition, color fringes were observed downstream along the wafer, indicating severe thickness nonuniformity (±16% over a 20 mm diameter). In contrast, excellent thickness uniformity was realized (±1% over a 20 mm diameter) when the TEG and TMAA are separated in the vapor. Although the growth rate was roughly doubled by separating the group III reactant fluxes, it was still lower than that for GaAs. This is likely due to the partial intermixing of the growth zones.

A comparison of PL spectra was made between layers grown by premixed and spatially separated growth modes for constant reactant flux. These experiments were performed at a growth temperature of 670 °C and a V/III ratio of 80. As shown in Figure 4, the material grown with spatially separated group III fluxes exhibited three orders of magnitude higher PL intensity than the premixed grown material. All material grown by spatially separating the TMAA and TEG

had strong room temperature PL, which was difficult to observe when premixed sources were used. A possible explanation for this effect is that in addition to the TMAA and TEG forming nonvolatile compounds, volatile compounds are also present which participate in the growth process and incorporate non-radiative centers in the AlGaAs.

In conclusion, we have demonstrated the effects of parasitic gas phase reactions between TMAA, TEG and AsH_3 in low pressure OMVPE of AlGaAs. The TMAA-TEG reaction the decreases the growth rate, degrades thickness uniformity and luminescence efficiency particularly at moderately high Al compositions. This reaction results in the formation of nonvolatile compounds, dramatically reducing the TEG transport to the substrate surface. These effects were greatly reduced by spatially separating the TMAA and TEG to minimize parasitic gas phase reactions. The effects of V/III ratio on film quality and Al composition have also been determined. High V/III ratios are necessary to inhibit the TMAA-TEG reaction likely due to a pre-reaction with TMAA and AsH_3. The AsH_3 flow requirement for acceptable quality AlGaAs films is sharply reduced using the multichamber flow modulation technique.

The authors wish to thank B. Butterfield, A. Schremer and K. Whittingham for technical assistance. This work was supported by the Joint Services Electronics Program under grant No. F49620-90-C-0039, the Strategic Defense Initiative Objective under contract No. N00014-89-J-1311, and the Defense Advanced Research Projects Agency under contract No. MDA97290C0058 Optoelectronics Technology Center.

FIGURE CAPTIONS

1. **Figure 1:** (a) Schematic illustration of implementation of Flow Modulation Epitaxy in the multichamber cell. Two TEG sources, one on each growth zone, allow for conventional premixed injection or spatially separated group III sources. The arsine is uniformly injected around the cell. The inner quartz ampoule (diameter - *d*) serves to separate the reactant fluxes of each deposition zone. (b) The exposure cycle for premixed and spatially separated TMAA and TEG. The arsine flow is uniformly distributed around the cell. Dotted lines represent the reactant flux zero reference. The degree of deposition zone separation is indicated schematically by set of arrows in the lower diagram of the figure.
2. **Figure 2:** Dependence of the aluminum composition (determined from Raman scattering) on V/III ratio for contant TEG and TMAA fluxes at 670 °C. The inset is the corresponding low temperature PL spectra for various V/III ratios. The luminescence intensity is magnified by the factors shown. Excitation conditions are as indicated.
3. **Figure 3:** The growth rate of undoped AlGaAs downstream along the wafer when TMAA and TEG are premixed prior to injection into the growth chamber and spatially separated in the gas phase. The nominal aluminum composition of the superlattice is 0.40. The growth rate is normalized to GaAs. The experimental conditions are as indicated.
4. **Figure 4:** Low temperature (1 K) photoluminescence of undoped AlGaAs grown with TEG, TMAA and AsH_3 using FME. The TMAA and TEG were either premixed prior to their injection into the reaction cell or spatially separated in the multichamber cell, as indicated. The luminescence is magnified by the factors shown. Excitation conditions, growth conditions and superlattice periods are as indicated.

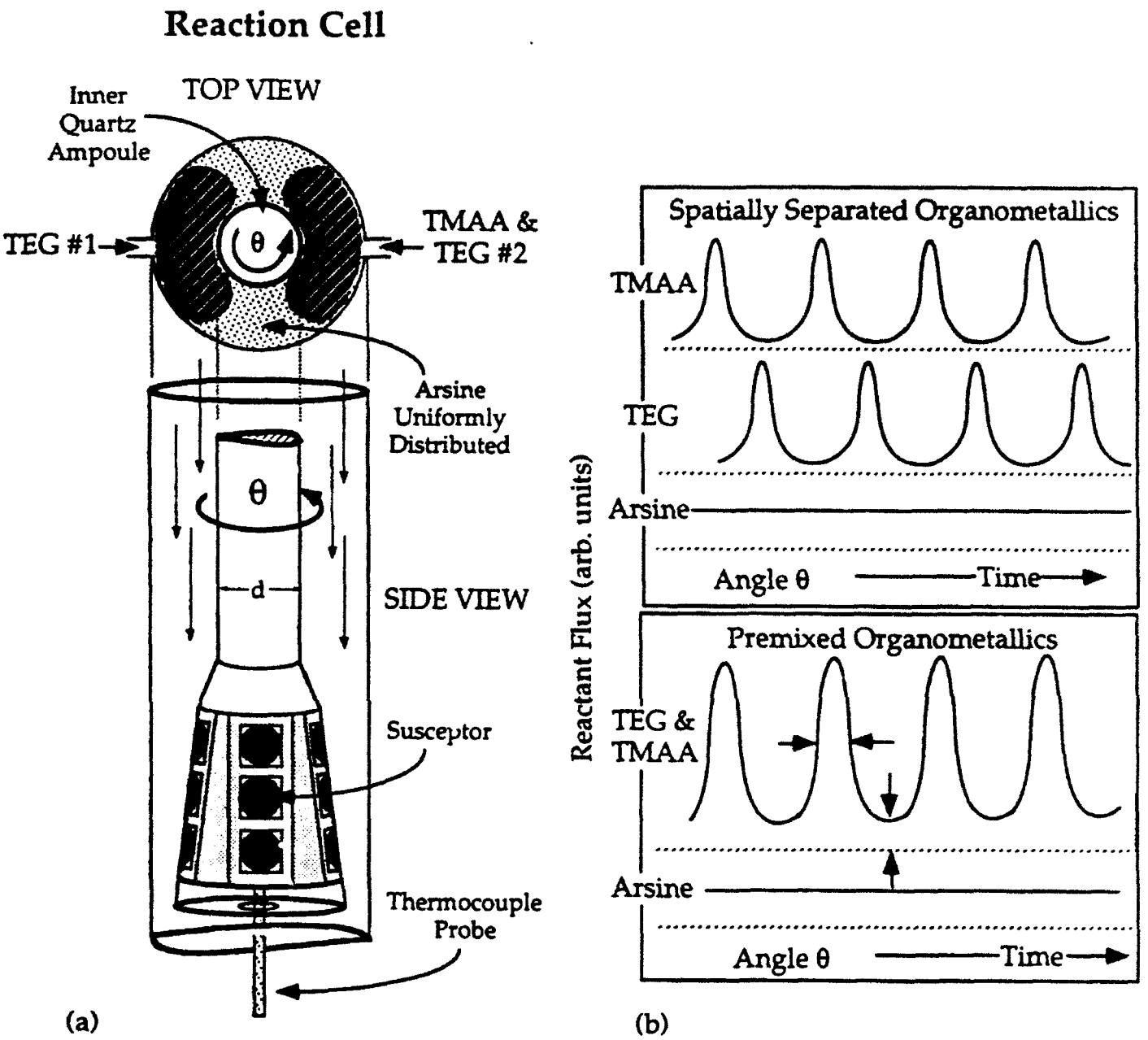

Reaction Cell
Inner Quartz Ampoule
TOP VIEW
TEG #1
TMAA & TEG #2
θ
Arsine Uniformly Distributed
θ
d
SIDE VIEW
Susceptor
Thermocouple Probe
(a)
Spatially Separated Organometallics
TMAA
TEG
Arsine
Angle θ
Time
Premixed Organometallics
TEG & TMAA
Arsine
Angle θ
Time
Reactant Flux (arb. units)
(b)

Figure 1

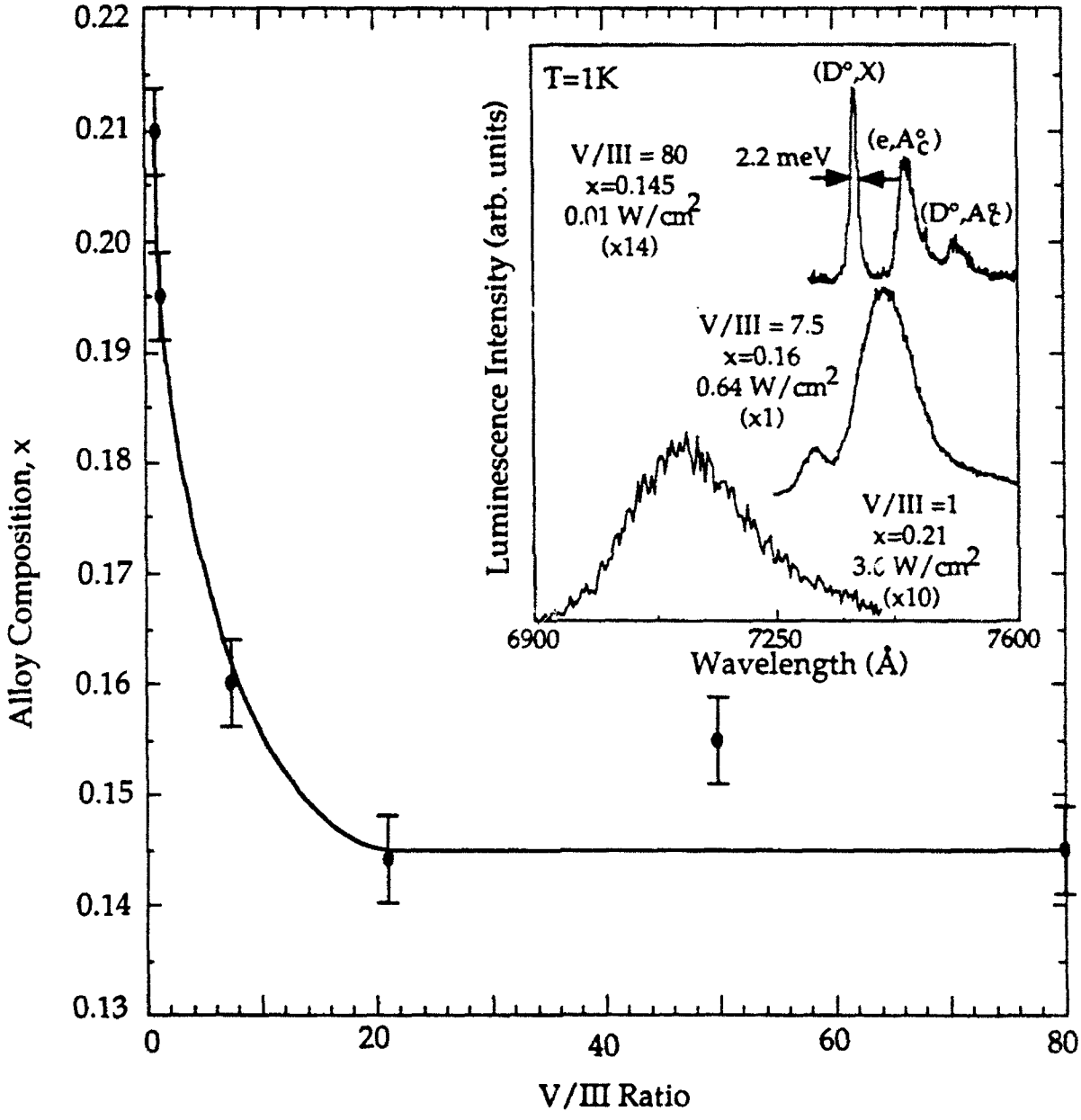

0.22
0.21
0.20
0.19
0.18
0.17
0.16
0.15
0.14
0.13
Alloy Composition, x
0
20
40
60
80
V/III Ratio
T=1K
(D°,X)
V/III = 80
x=0.145
0.01 W/cm2
(x14)
2.2 meV
(e,A°C)
(D°,A°C)
V/III = 7.5
x=0.16
0.64 W/cm2
(x1)
V/III =1
x=0.21
3.0 W/cm2
(x10)
Luminescence Intensity (arb. units)
6900
7250
7600
Wavelength (Å)

Figure 2

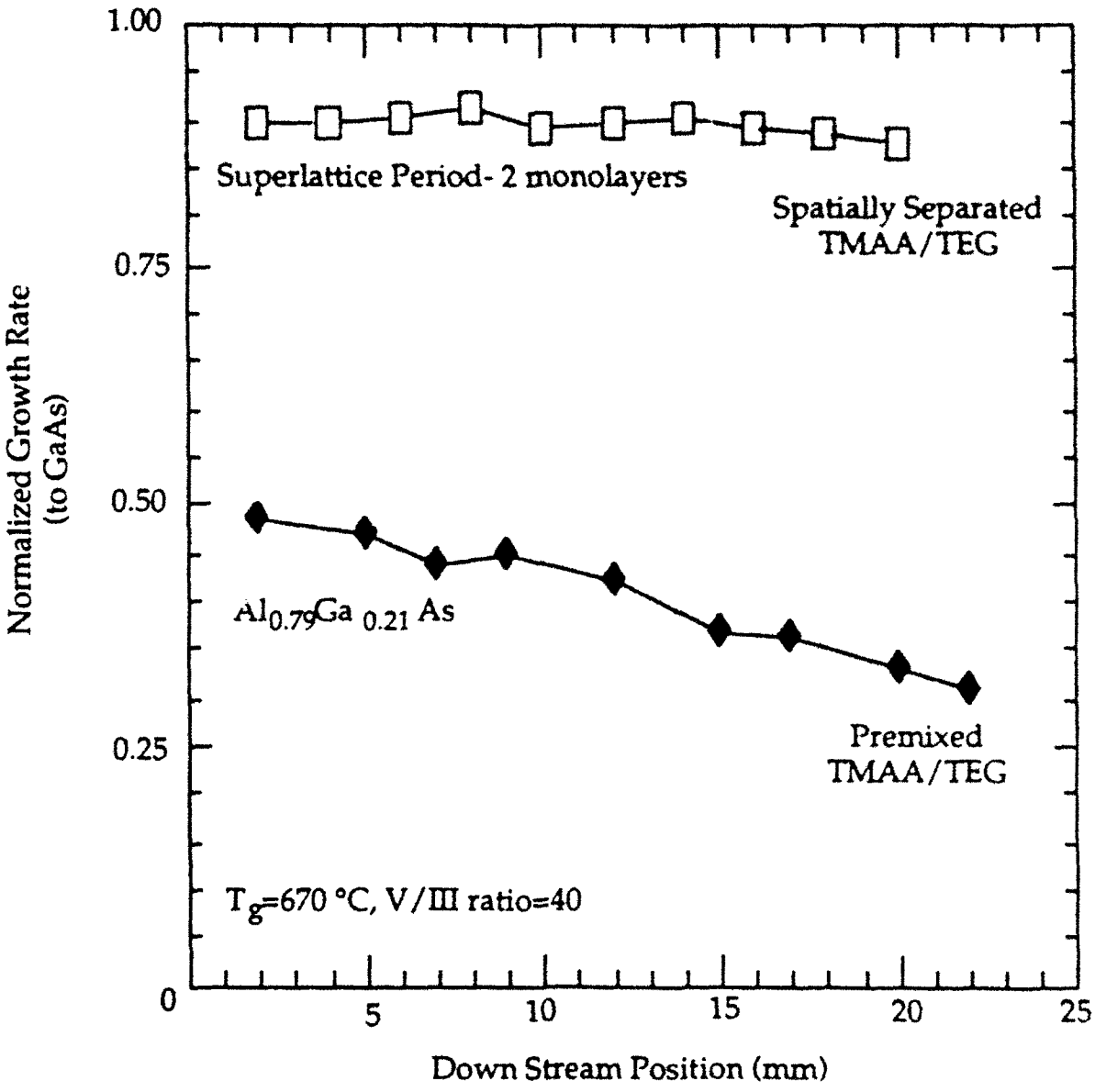

1.00
0.75
0.50
0.25
0
Normalized Growth Rate
(to GaAs)
Superlattice Period- 2 monolayers
Spatially Separated
TMAA/TEG
$Al_{0.79}Ga_{0.21}As$
Premixed
TMAA/TEG
T_g=670 °C, V/III ratio=40
5
10
15
20
25
Down Stream Position (mm)

Figure 3

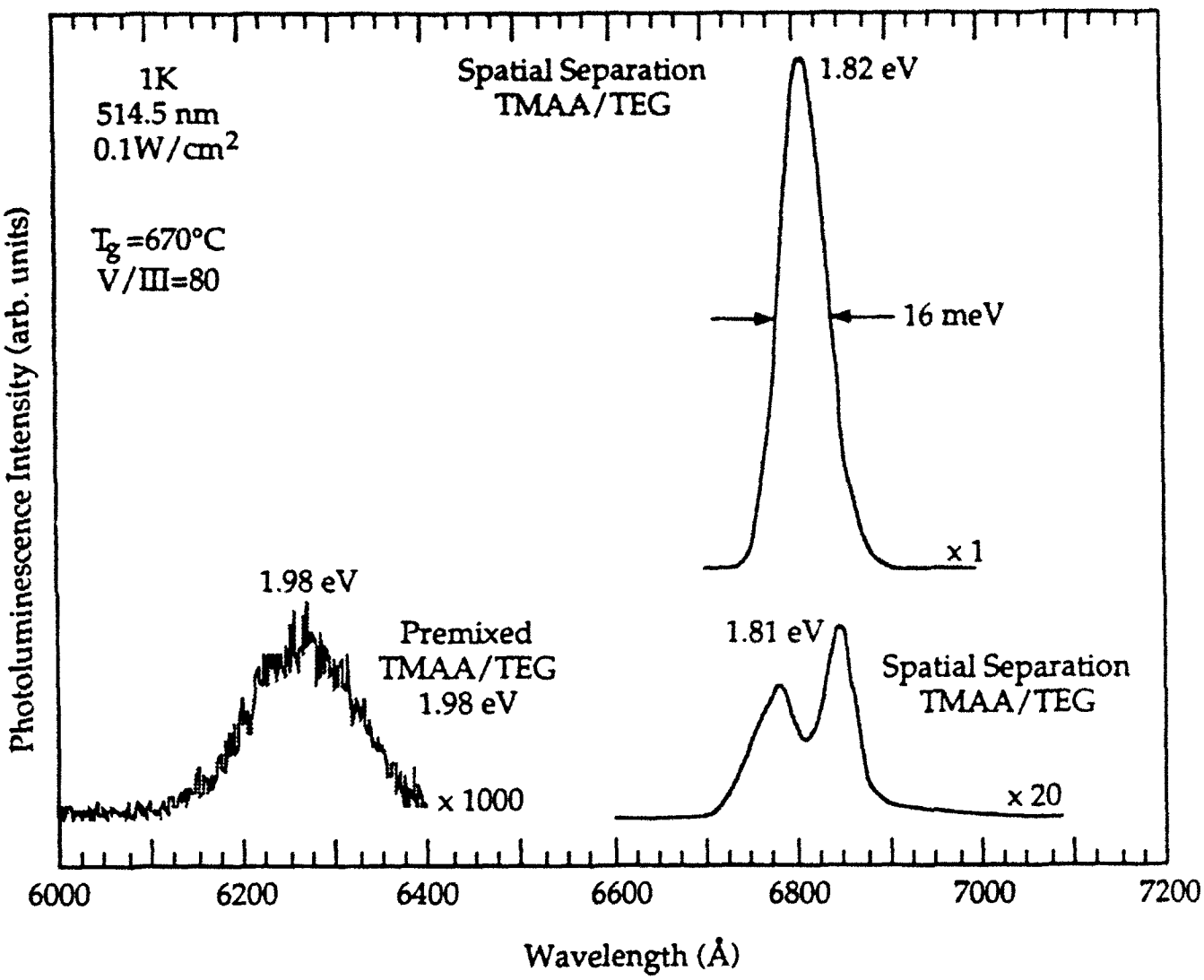

1K
514.5 nm
0.1W/cm2
Tg =670°C
V/III=80
Spatial Separation
TMAA/TEG
1.82 eV
16 meV
x 1
1.98 eV
Premixed
TMAA/TEG
1.98 eV
x 1000
1.81 eV
Spatial Separation
TMAA/TEG
x 20
Photoluminescence Intensity (arb. units)
6000
6200
6400
6600
6800
7000
7200
Wavelength (Å)

Figure 4

REFERENCES

1. T.F. Kuech, D.J. Wolford, E. Veuhoff, V. Deline, P.M. Mooney, R. Potemski and J. Bradley, J. Appl. Phys. **62**, 632 (1987).
2. A.C. Jones and S.A. Rushworth, J. Crystal Growth 106, 253 (1990).
3. J.S. Roberts, C.C. Button, J.P.R. David, A.C. Jones and S.A. Rushworth, J. Crystal Growth **104**, 857 (1990).
4. W.S. Hobson, T.D. Harris, C.R. Abernathy and S.J. Pearton, Appl. Phys. Lett. 58, 77 (1991).
5. C.R. Abernathy, A.S. Jordan, S.J. Pearton, W.S. Hobson, D.A. Bohling and G.T. Muhr, Inst. Phys. Conf. Ser. No. 112: Chapter 3, 149 (1991).
6. C.R. Abernathy, A.S. Jordan, S.J. Pearton, W.S. Hobson, D.A. Bohling and G.T. Muhr, Appl. Phys. Lett. **56**, 2654 (1991).
7. W.S. Hobson, F. Ren, M. Lamont Schnoes, S.K. Sputz, T.D. Harris, S.J. Pearton, C.R. Abernathy and K.S. Jones, Appl. Phys. Lett. 59, 1975 (1991).
8. F. Kobayashi, S. Iio, T. Kuwahara and Y. Sekiguchi, Jap. J. Appl. Phys. 30, 1971 (1991).
9. C.R. Abernathy, S.J. Pearton, F.A. Baiocchi, T. Ambrose, A.S. Jordan, D.A. Bohling and G.T. Muhr, J. Crystal Growth 110, 457 (1991).
10. J.R. Shealy, J. Crystal Growth 87, 350 (1988).
11. N. Kobayashi, T. Makimoto, Y. Yamauchi and Y. Horikoshi, J. Appl. Phys. 66, 640 (1989).
12. N. Saint-Cricp, G. Landa, J.B. Renucci, I. Hardy and A. Munoz-Yague, J. Appl. Phys. 61, 1206 (1987).
13. C. Colvard, T.A. Gant, M.V. Klein, R. Merlin, R. Fischer, H. Morkoç and A.C. Gossard, Phys. Rev. B 31, 2080 (1985).
14. D.C. Reynolds, K.K. Bajaj, C.W. Litton, P.W. Yu, J. Klem, C.K. Peng, H. Morkoç and J. Singh, Appl. Phys. Lett. 48, 727 (1986).

THE EFFECTS OF GAS PHASE REACTIONS OF TRIMETHYLAMINE ALANE ON AlGaAs FILMS GROWN BY ORGANOMETALLIC VAPOR PHASE EPITAXY

B.L. Pitts, D.T. Emerson, M.J. Matragrano* and J.R. Shealy
OMVPE Facility, School of Electrical Engineering
Cornell University
Ithaca, N.Y. 14853

Abstract

The effect of gas reaction between trimethylamine alane, triethylgallium and arsine on AlGaAs films grown by Organometallic Vapor Phase Epitaxy is reported. Using a multichamber reaction cell, we have been able to independently observe the effects of TMAA-TEG and TMAA-arsine gas phase reactions. The effects of TMAA-TEG gas phase reactions were identified by comparing films that were grown by premixing the TMAA and TEG prior to injection, to those that where the TMAA and TEG were spatially separated. The TMAA-TEG reactions results in the formation of nonvolatile compounds which condense upstream from the reaction cell, resulting in a severe reduction in growth rate, as well a depletion of the gallium species. The TMAA-arsine reaction produces compositional dependence on V/III ratio. The arsine flow requirement for attaining good surface morphology has been identified. Under the appropriate growth conditions we demostrate that acceptable purity AlGaAs can be grown using low V/III ratios.

* Department of Material Science, Bard Hall, Cornell University, Ithaca, N.Y. 14853

I. Introduction

The ability to produce high purity AlGaAs material has led to the realization of many high performance optical and high-speed electronic devices. The growth of AlGaAs grown by Organometallic Vapor Phase Epitaxy (OMVPE) has traditionally been plagued with high oxygen and carbon incorporation. Although many attempts have been made to reduce these effects, relatively high concentrations of carbon, and to a lesser extent oxygen, still persist using conventional sources[1,2]. A major reason for these problems is due to the widely used aluminum source, trimethylaluminum (TMA). TMA has a strong aluminum-carbon bond and the ability to form volatile aluminum alkoxide compounds resulting in oxygen contaminated AlGaAs layers[3]. Triethylaluminum (TEA) is also used as an aluminum source, and it has demonstrated lower carbon incorporation in AlGaAs than TMA. Low temperature (<5 K) mobilities near 500,000 $cm^2/V \cdot sec$ have been reported for AlGaAs/GaAs modulation doped heterostructure (sheet electron density - $8(10^{11})$ cm^{-2}) using TEA and triethylgallium (TEG)[4]. Comparable results do not yet exist for structures grown with TMA or trimethylamine alane (TMAA). However, some residual oxygen still remains using TEA. Also, TEA has a low vapor pressure (0.5 torr at 55 °C) which is inconvenient for OMVPE.

Trimethylamine alane (TMAA) has received much attention as a viable aluminum source in both OMVPE and Chemical Beam Epitaxy (CBE)[5-16]. TMAA does not have a direct aluminum-carbon which is expected to reduce the carbon comtamination. Also, when TMAA reacts with O_2 and H_2O involatile Al-O compounds form thereby reducing the oxygen contamination. Reports indicate that using TMAA along with TEG or trimethylgallium (TMG) and arsine (AsH_3), under the appropriate growth conditions (very high V/III ratios and gas velocities), can result in the highest purity OMVPE grown AlGaAs[5,7]. Improved photoluminescence (reduced donor-to-acceptor related transition) and mobilities (77 K mobility exceeding 14,000 $cm^2/V \cdot sec$ for $Al_{0.14}Ga_{0.86}As$) have been achieved[5].

Recently, high quality AlInAs/GaInAs structures have also been attained using TMAA. Low threshold lasers and high transconductance selectively doped field effect transistors have been demonstrated using TMAA in both AlGaAs/GaAs and AlInAs/GaInAs material systems[11,12,15].

Previous investigators have reported parasitic reactions between TMAA and metal-alkyl compounds in OMVPE[5,12,15]. Inferior compositional and thickness uniformity was realized, probably due to gas phase reactions between TMAA and other reactants[5]. Grady *et al.* performed Fourier transform infrared (FTIR) spectroscopy on TMAA/TMG vapor mixture and reported the presence of strong gas phase reactions between TMAA and TMG resulting in a depletion of gallium species[17]. TMAA also has a low thermal decomposition temperature (~100 °C), allowing predeposition on the side walls of the reaction cell. A remedy to these problems has been to increase the gas velocity which reduces the contact time between the reactants in the growth chamber. Hobson *et al.* used gas velocities greater than 1 m/sec to overcome these effects[11]. In addition, high V/III ratios are necessary to achieve high purity results. Studies have been made investigating the growth chemistry of CBE using TMAA with other organometallic compounds[13,14]. Notably, Kobayashi *et al.* reported the effects of gas-phase reactions between TMAA and TEG in CBE. They concluded that TMAA-TEG reactions produced non-volatile compounds which decreases the growth rate and reduces gallium incorporation[13].

This paper investigates the effects of gas phase reactions between TMAA, TEG and AsH_3 on AlGaAs films grown by low pressure OMVPE. The reactor used in this study provides for independent observation of the effects of TEG-TMAA and TMAA-AsH_3 gas phase reactions. Gas phase reactions involving TMAA and TEG result in the formation of nonvolatile compounds upstream, which condense on the reactor wall, resulting in a reduction of growth rate and a degradation of the deposition uniformity. The TMAA-TEG reaction has severe effects on the quality of the AlGaAs films especially at low V/III ratios. The

TMAA-AsH_3 reaction produces a compositional dependence on the gas phase stoichiometry (V/III ratio). High quality AlGaAs with excellent thickness and compositional uniformity was produced by spatially separating the TMAA and TEG in the gas phase which minimizes the parasitic reactions. Applying flow modulation techniques[18,19] dramatically reduces the arsine flow requirements for producing acceptable quality AlGaAs.

II. Experimental

AlGaAs layers were grown on (100) Si-doped n^+ GaAs and (100) semi-insulating GaAs substrates in a vertical barrel, multichamber OMVPE system[20]. The reaction cell is made of 6-inch high purity quartz. The graphite susceptor, which can hold up to 18-1.5 inch wafers, is inductively heated by RF radiation. Each organometallic line has independent pressure control to enhance transport to the reaction cell. The system is also equipped with an in-situ quadrapole mass analyzer to detech gas leaks before experiments. Figure 1a illustrates the gas flow in the reaction chamber. The reaction chamber has two growth zones that are spatially separated by large hydrogen fluxes. The substrates are rotated through the growth zones without valve switching. An inner quartz ampoule (diameter-*d*) separates the reactant fluxes. Arsine is uniformly injected into the entire growth chamber. During the group III exposure cycle the local V/III ratio is estimated to be 25% of the average value. The V/III ratio quoted throughout represents the average V/III ratio determined by the total injected reactant fluxes. The flow modulation exposure cycle for each growth mode is shown in Figure 1b. The group III reactants are modulated while the AsH_3 exposure remains constant. In the conventional growth mode, the TEG and TMAA are premixed prior to injection into the reaction cell while the susceptor is rotated at 0.1 rev/sec. In the spatially separated growth mode, the TEG and TMAA are injected into separate growth zones, minimizing the TEG-TMAA gas phase reactions. For the group III flux used in this study, susceptor rotation speeds greater than 1 rev/sec are needed to produce sub-monolayer exposure cycles which result in mixed alloys.

Rotation speeds ranging from 0.1 to 0.7 rev/sec were used when the reactant fluxes were separated in the vapor. Raman spectroscopy confirmed the existence of short period superlattices (confined LO GaAs and AlAs vibrations) on all samples produced with this method. The degree of deposition zone separation (indicated by the set of arrows in Figure 1b) is proportional to the amount of hydrogen carrier gas injected between each zone. Due to a small amount of zone intermixing occurs in the spatially separated growth mode (see Figure 1b), the short period superlattices have graded interfaces. The total gas flow was 30 slm, while the gas velocity was maintained at 30 cm/s.

The sources used were TMAA, adduct-purified TEG[23] and 100% Phoenix Research Grade AsH_3. Arsine was passed through Al-Ga-In melt to reduce the oxygen and H_2O contamination[1]. Palladium diffused H_2 was used as a carrier gas. The growth pressure was 76 torr. The TMAA was held at 23 °C (vapor pressure~2 torr) while a H_2 flow of 57 sccm was passed through the bubbler. The TEG was also held at 23 °C (vapor pressure-5 torr), while the flow varied from 18 to 50 sccm. Both TEG and TMAA were maintained at 100 torr. The substrates were first degreased in organic solvents, then etched for 10 minutes in $5H_2SO_4$:$1H_2O_2$:$1H_2O$ prior to growth. The growth temperature varied from 635 to 750 °C and the V/III ratio was varied from 1 to 80. Layer thicknesses ranged from 0.5-2 μm.

Films were characterized by Hall measurements, Raman spectroscopy, photoluminescence (PL) and double crystal X-ray diffractometry. Raman spectroscopy was used to determine the aluminum composition[22] and the structure features of the superlattices[23]. Optical quality was assessed using low (1 K) and room temperature photoluminscence (PL). Low temperature PL was carried out with samples submerged in superfluid He with photoexcitation provided by the 514.5 nm line of an Ar^+ laser. Thickness measurements were made by a combination of angle bevelling and staining and from analysis of reflectance spectra. A double crystal X-ray diffractometer with a computer controlled X-Y stage was

used to determine the layer composition[24] and map the compositional uniformity across the wafer. The X-ray beam of Cu $K\alpha_1$ monochromatized by (111) reflections from a perfect Si crystal.

III. Results and Discussion

All layers were n-type and background carrier concentration in the $10^{15}cm^{-3}$ range. This is believed to be due to Si impurities in TMAA[5]. The V/III ratio and growth temperature criteria for good surface morphology were investigated over the temperature range 635 to 750 °C for $Al_xGa_{1-x}As$ ($x < 0.2$). When the substrate temperature is below 675 °C, more AsH_3 must be supplied to maintain specular surfaces. As the growth temperature is increased beyond 675 °C, the good morphology/bad morphology transition approaches a constant value of unity. An analagous study has been reported using TEG and AsH_3[25].

With growth temperature (670 °C) and group III flux constant, the Al mole fraction as determined by X-ray diffraction and Raman scattering was found to vary with V/III ratio in the conventional premixed growth mode. As shown in Figure 2, more Al is incorporated in the film as the V/III ratio is decreased. For low TMAA fluxes, corresponding to alloy compositions less than 20%, AsH_3 appears to prevent the TEG-TMAA reaction which is shown to reduce the TEG tranport to the growth surface. As the PL spectra in Figure 3 reveals, the sample quality degrades with decreasing V/III ratio. At a V/III ratio of 80, a full width at half maximum exciton linewidth of 2.2 meV was observed for $Al_{0.15}Ga_{0.85}As$. This compares favorably to the narrowest linewidth ever produced using OMVPE[26] As the V/III ratio decreased to 50, the exciton linewidth was 6.5 meV. The linewidth continue to broaden but was clearly identifiable when the V/III ratio was lowered to 7.5. Finally, at a V/III ratio of 1, the exciton feature was absent, indicating that even though morphology was good for these growth conditions, material purity was relativley poor. The need for large arsine flows may suggest that the TMAA-AsH_3 reaction inhibits the TMAA-TEG reaction which is demonstrated to severely degrade the quality of the AlGaAs

films.

Gas phase reactions between TEG and TMAA have major effects on the growth rate. As Figure 4 illustrates, when the TMAA and TEG are premixed, the AlGaAs growth rate is approximately half that of GaAs with same TEG reactant flux. A relatively high V/III ratio was used (V/III=40) to eliminate the effects of AsH_3 flows described earlier. The effect of growth rate reduction was also reported for CBE for premixed TMAA and TEG[13]. The Al composition for AlGaAs grown using premixed sources was 79% whereas that for the spatially separated sources was nominally 40%. Assuming that the Al and Ga incorporation in the AlGaAs layer in the two growth modes are equal, an estimated 70% of the TMAA reacts in the gas phase to produce nonvolatile compounds. The reduction of gallium incorporation seems to be consistent with FTIR results on TMAA/TMG gas mixture[15]. In addition, color fringes were observed downstream along the wafer, indicating severe thickness nonuniformity (±16% over a 20 mm diameter). In contrast, excellent thickness uniformity was realized (±1% over a 20 mm diameter) when the TEG and TMAA are separated. Although the growth rate was roughly doubled by separating the group III reactant fluxes, it was still lower than that for GaAs, likely due to the partial intermixing of the growth zones.

Figures 6 and 7 are compositional uniformity maps for the premixed and spatially separated growth modes, respectively. The compositional uniformity is similar for both growth modes (±2%). In both cases, the aluminum concentration decreases downstream along the wafer. The compositional uniformity is approximately the same for both growth modes (~±2% over 40 mm^2). This is consistent with other reports using TMAA and TEG[5].

The X-ray rocking curves for the premixed and spatially separated growth modes are compared in Figure 8. The premixed grown material exhibited a broad peak from the epitaxial layer, indicative of poor structural quality. In addition, long tail on the substrate is present probably due to the to compositional grading

in the layer. In constrast, layers produced by the spatially separating the TMAA and TEG had a peak that were comparable to that of the substrate. Both curves exhibited extra peaks which is possibly due to compositional grading. This is believed to be due to irratic transport of the TMAA which commonly occurs in solid organometallic sources such as trimethylindium[27].

PL spectra was crmpared between layers grown by premixed and spatially separated growth mod. . for constant reactant flux. These experiments were performed at a growth temperature of 670 °C and a V/III ratio of 80. As shown in Figure 9, the material grown with spatially separated group III fluxes exhibited much stronger PL intensity than the premixed grown material. Material produce from spatially separating the TMAA and TEG exhibited strong room temperature PL, which was difficult to observe when premixed sources were used. An explanation for this effect is that in addition to the TMAA and TEG forming nonvolatile compounds, volatile compounds are also present which participate in the growth process and reduces the radiative efficiency in the AlGaAs.

VI. Summary

We have described the effects of gas phase reactions between TMAA, TEG and AsH_3 on AlGaAs films grown by OMVPE. The TMAA-AsH_3 produces a compositional dependence on the gas phase stoichiometry (V/III ratio). The TMAA-TEG reaction result in the formation of nonvolatile compounds which reduces the growth rate and degrades the depostion uniformity. Poor luminescence was observed suggesting the presence of volatile compounds which produce non-radiative centers. These effects were dramatically reduced by spatially separating the reactants. The arsine flow requirements has been identified for yielding good surface morphology AlGaAs using TMAA, TEG and AsH_3. Finally, we have demonstrated that under the appropriate growth condition, acceptable quality AlGaAs can be produced using low V/III ratios.

V. Acknowledgements

The authors wish to thank B. Butterfield, A. Schremer and K. Whittingham

for technical assistance. This work was supported by the Joint Services Electronics Program under grant No. F49620-90-C-0039, the Strategic Defense Initiative Objective under contract No. N00014-89-J–1311, and the Defense Advanced Research Projects Agency under contract No. MDA97290C0058 Optoelectronics Technology Center.

FIGURE CAPTIONS

1. **Figure 1:** (a) Schematic illustration of implementation of Flow Modulation Epitaxy in the multichamber cell. Two TEG sources, one on each growth zone, allow for conventional premixed injection or spatially separated group III sources. The arsine is uniformly injected around the cell. The inner quartz ampoule (diameter - *d*) serves to separate the reactant fluxes of each deposition zone. (b) The exposure cycle for premixed and spatially separated TMAA and TEG. The arsine flow is uniformly distributed around the cell. Dotted lines represent the reactant flux zero reference. The degree of deposition zone separation is indicated schematically by set of arrows in the lower diagram of the figure.
2. **Figure 2:** Dependence of the aluminum composition (determined from Raman scattering) on V/III ratio for contant TEG and TMAA fluxes at 670 °C.
3. **Figure 3:** Low temperature PL spectra for various V/III ratios at 670 °C and constant TEG and TMAA fluxes. Compositional variation is due to TMAA-arsine gas phase reaction. The luminescence intensity is magnified by the factors shown. Excitation conditions are as indicated.
4. **Figure 4:** The growth rate of undoped AlGaAs downstream along the wafer when TMAA and TEG are premixed prior to injection into the growth chamber and spatially separated in the gas phase. The nominal aluminum composition of the superlattice is 0.40. The growth rate is normalized to GaAs. The experimental conditions are as indicated.
5. **Figure 5:** Compositional uniformity across a 20X20 mm wafer for AlGaAs grown by premixing the TMAA and TEG at 670°C and a V/III of 40. The map was constructed from X-ray diffraction close to the (004) reflection.
6. **Figure 6:** Compositional uniformity across a 12X16 mm wafer for AlGaAs grown by spatially separated TMAA and TEG at 670°C and a V/III of 40. The map was constructed from X-ray diffraction close to the (004) reflection.

7. **Figure 7:** X-ray rocking curve of AlGaAs grown by either premixed prior to their injection into the reaction cell or spatially separated in the multichamber cell, as indicated. The nominal Al compositions of the layer grown by premixed and spatially separated growth modes were 70% and 26%, respectively. Both layer were grown at 670°C and a V/III of 40. The X-ray diffraction was taken close to the (004) reflection.

8. **Figure 8:** Low temperature (1 K) photoluminescence of undoped AlGaAs grown with TEG, TMAA and AsH_3 using FME. The TMAA and TEG were either premixed prior to their injection into the reaction cell or spatially separated in the multichamber cell, as indicated. The luminescence is magnified by the factors shown. Excitation conditions, growth conditions and superlattice periods are as indicated.

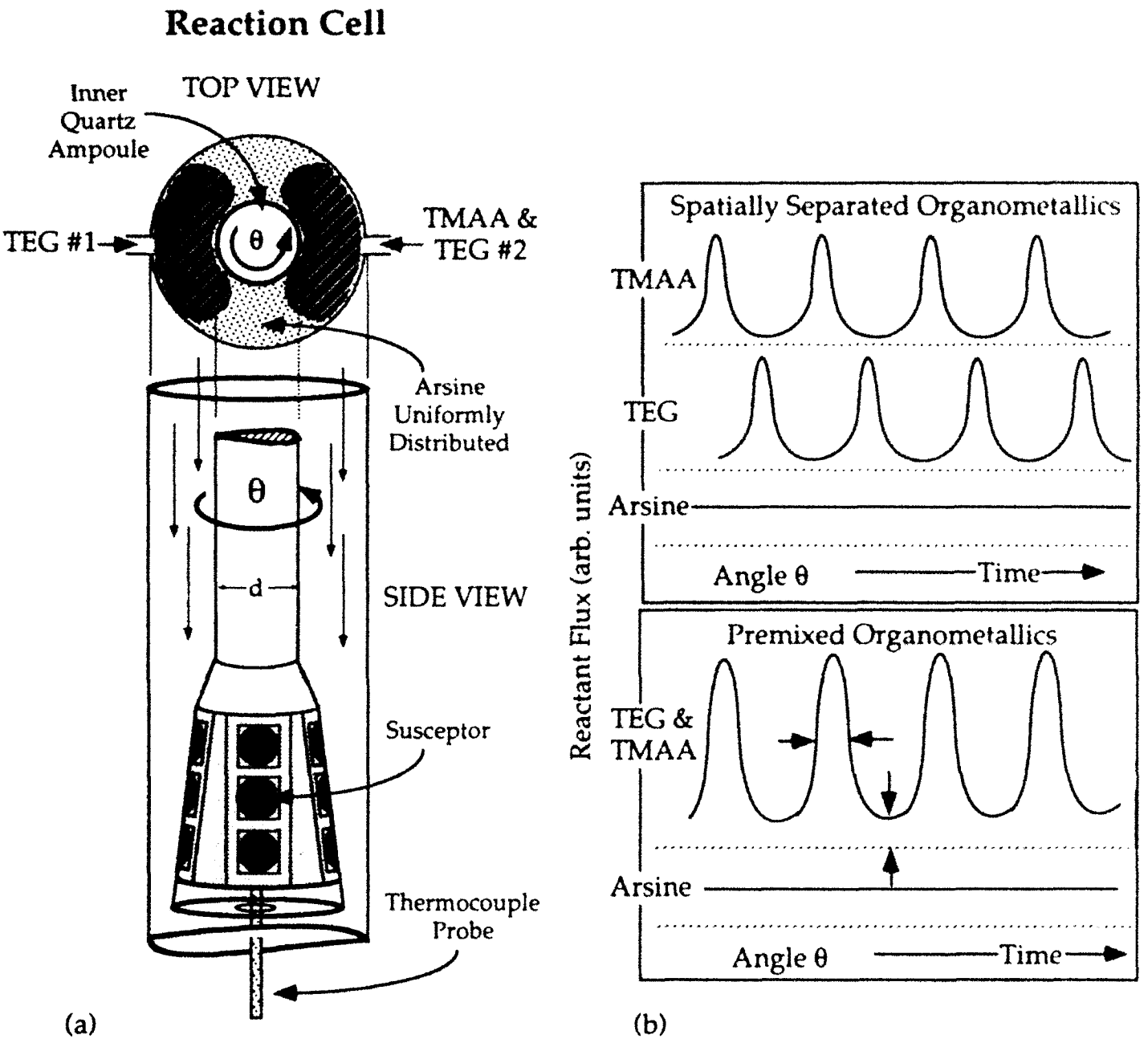

Reaction Cell
TOP VIEW
Inner Quartz Ampoule
TEG #1
TMAA & TEG #2
θ
Arsine Uniformly Distributed
θ
d
SIDE VIEW
Susceptor
Thermocouple Probe
(a)
Spatially Separated Organometallics
TMAA
TEG
Arsine
Angle θ
Time
Reactant Flux (arb. units)
Premixed Organometallics
TEG & TMAA
Arsine
Angle θ
Time
(b)

Figure 1

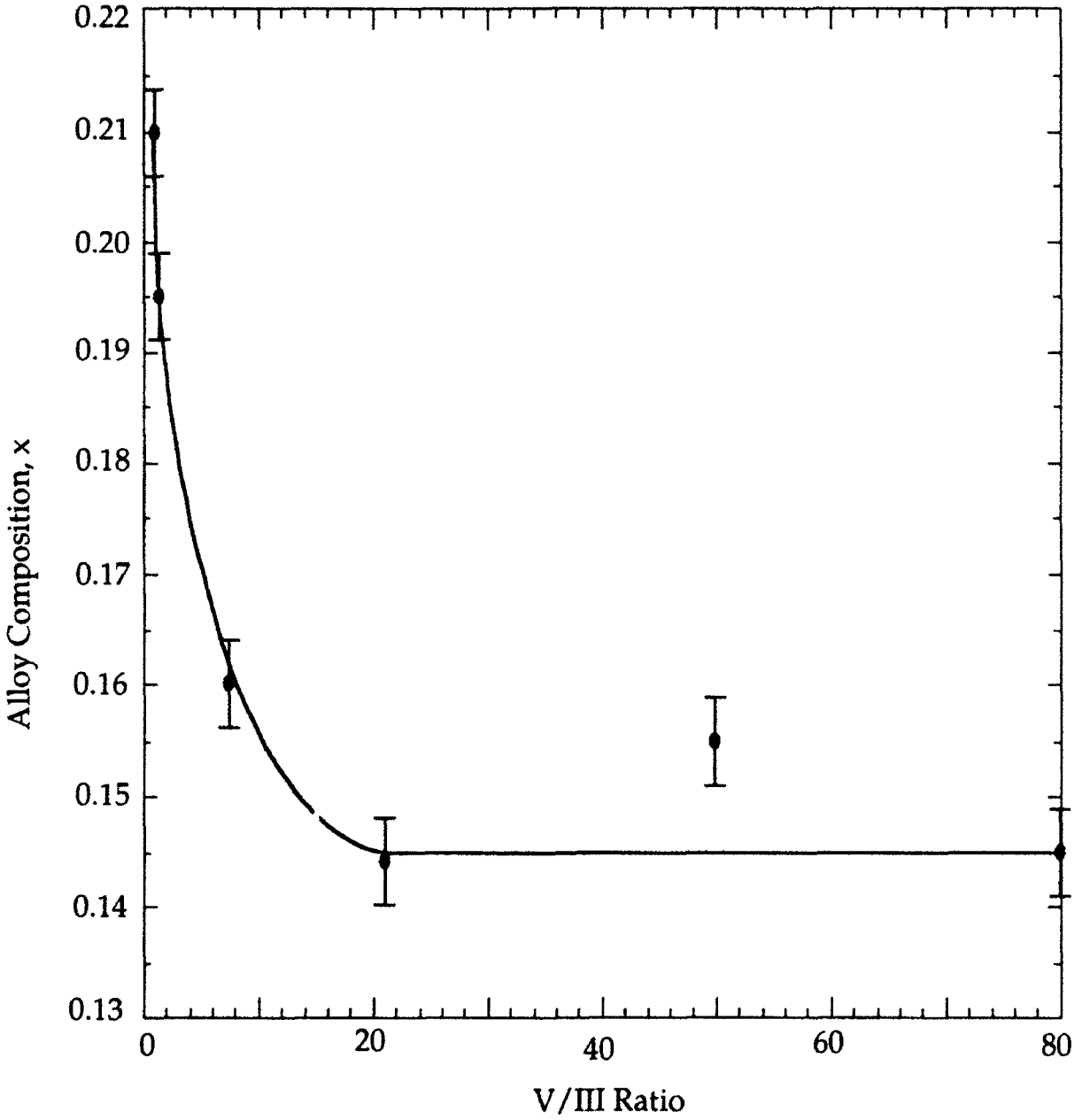

0.22
0.21
0.20
0.19
0.18
0.17
0.16
0.15
0.14
0.13
Alloy Composition, x
0
20
40
60
80
V/III Ratio

Figure 2

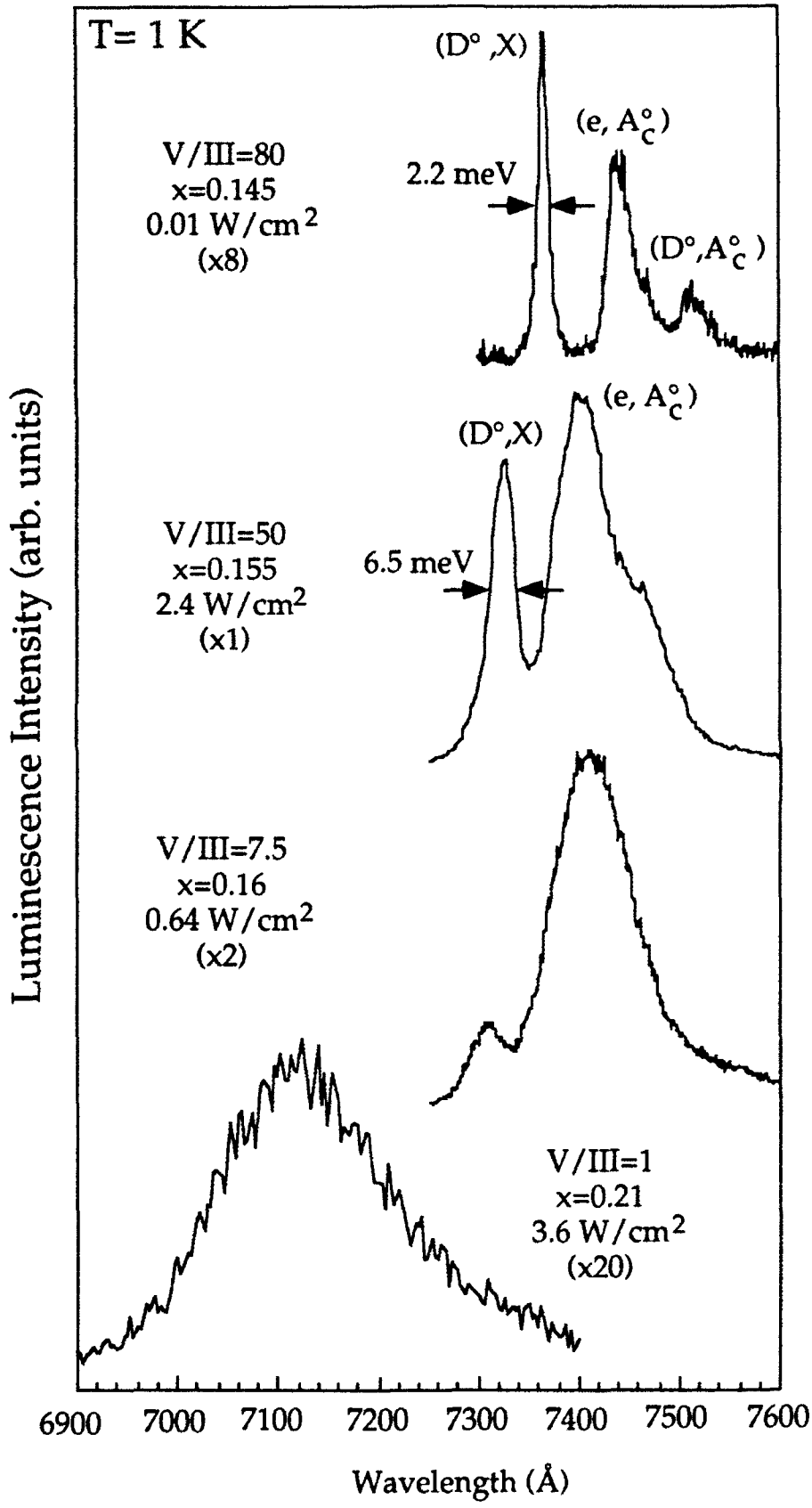

T= 1 K
(D° ,X)
(e, A°c)
2.2 meV
(D°,A°c)
V/III=80
x=0.145
0.01 W/cm2
(x8)
(e, A°c)
(D°,X)
6.5 meV
V/III=50
x=0.155
2.4 W/cm2
(x1)
V/III=7.5
x=0.16
0.64 W/cm2
(x2)
V/III=1
x=0.21
3.6 W/cm2
(x20)
Luminescence Intensity (arb. units)
6900
7000
7100
7200
7300
7400
7500
7600
Wavelength (Å)

Figure 3

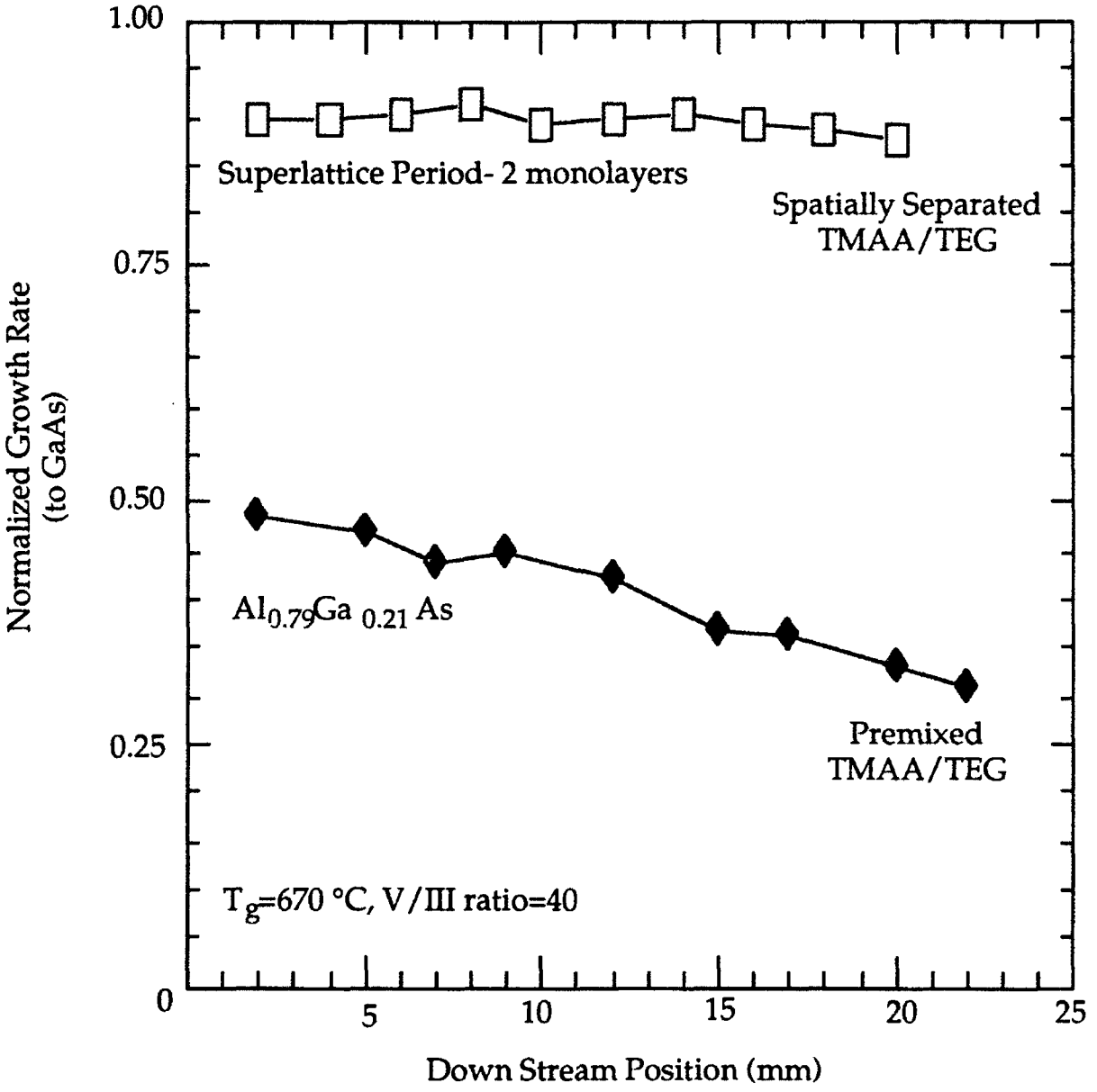

1.00
0.75
0.50
0.25
0
Normalized Growth Rate
(to GaAs)
Superlattice Period- 2 monolayers
Spatially Separated
TMAA/TEG
$Al_{0.79}Ga_{0.21}As$
Premixed
TMAA/TEG
T_g=670 °C, V/III ratio=40
5
10
15
20
25
Down Stream Position (mm)

Figure 4

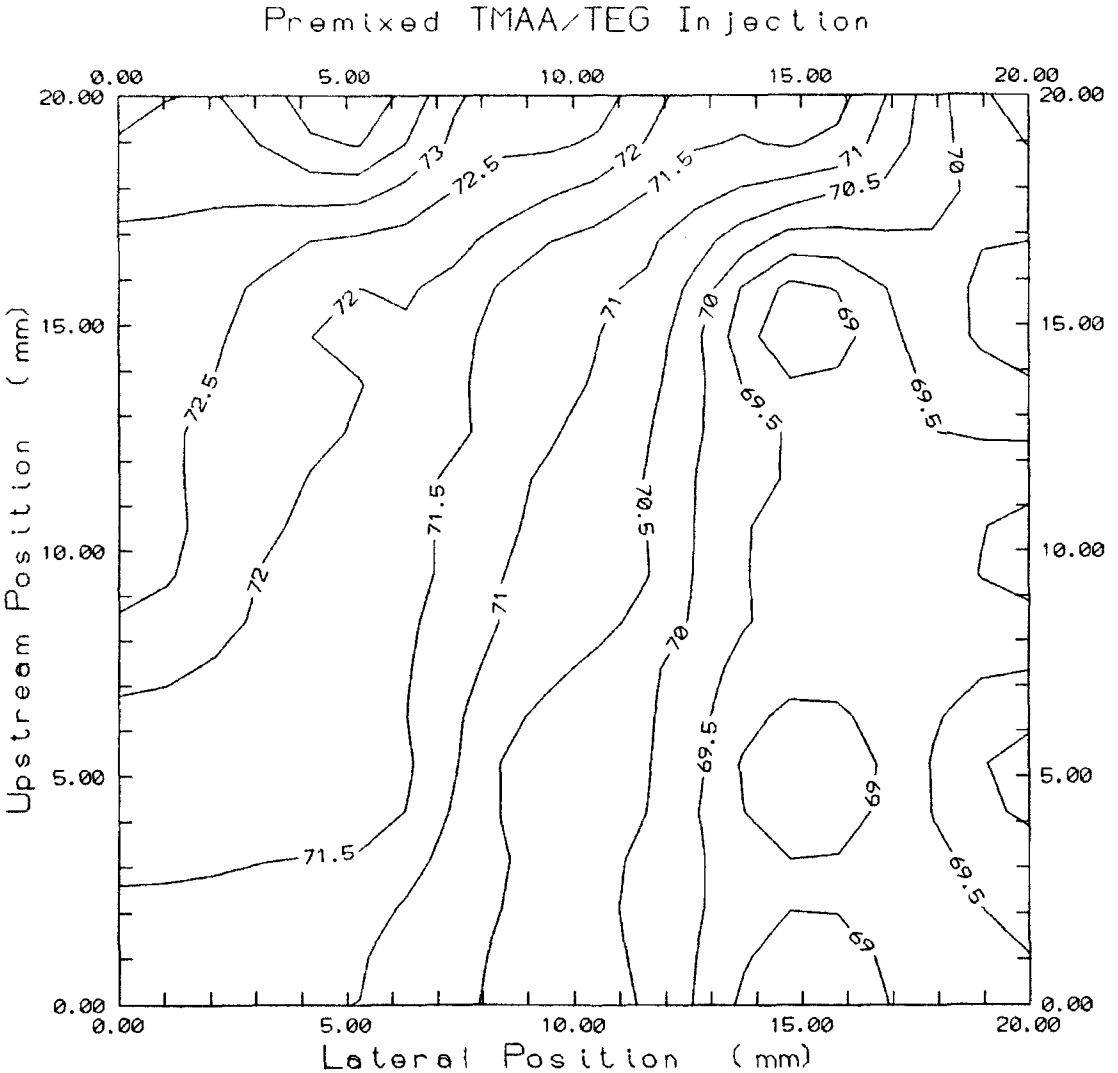

Premixed TMAA/TEG Injection
Upstream Position (mm)
Lateral Position (mm)
0.00
5.00
10.00
15.00
20.00
73
72.5
72
71.5
71
70.5
70
69.5
69

Figure 5

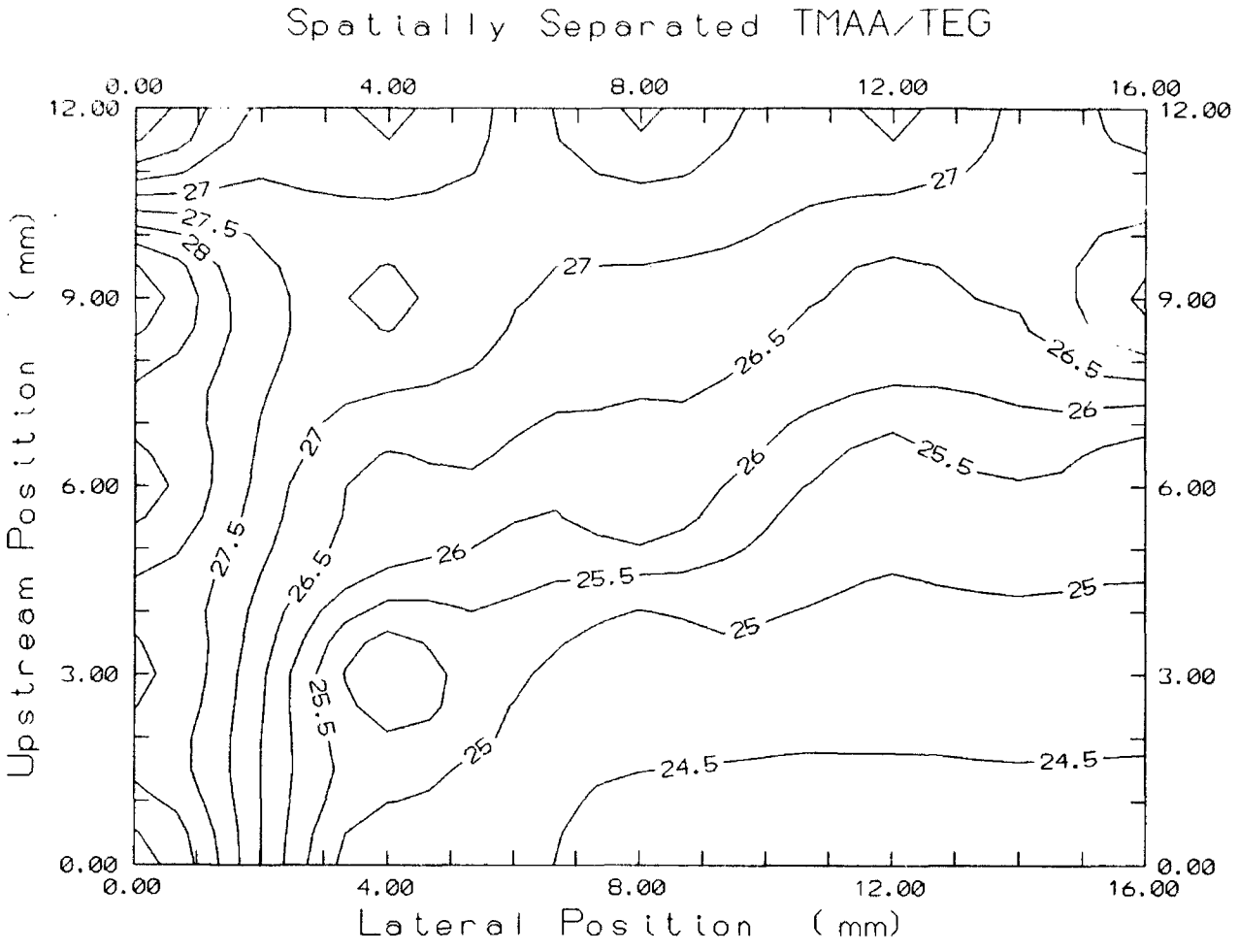

Spatially Separated TMAA/TEG
Upstream Position (mm)
Lateral Position (mm)
0.00
4.00
8.00
12.00
16.00
3.00
6.00
9.00
27
27.5
28
26.5
26
25.5
25
24.5

Figure 6

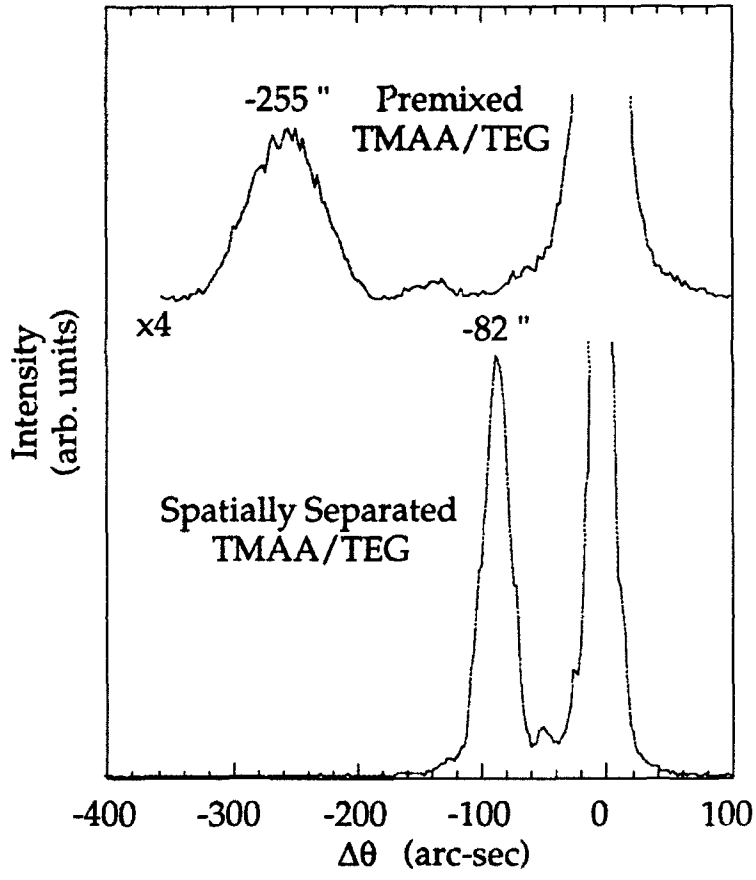

-255 "
Premixed
TMAA/TEG
x4
-82 "
Intensity
(arb. units)
Spatially Separated
TMAA/TEG
-400
-300
-200
-100
0
100
Δθ (arc-sec)

Figure 7

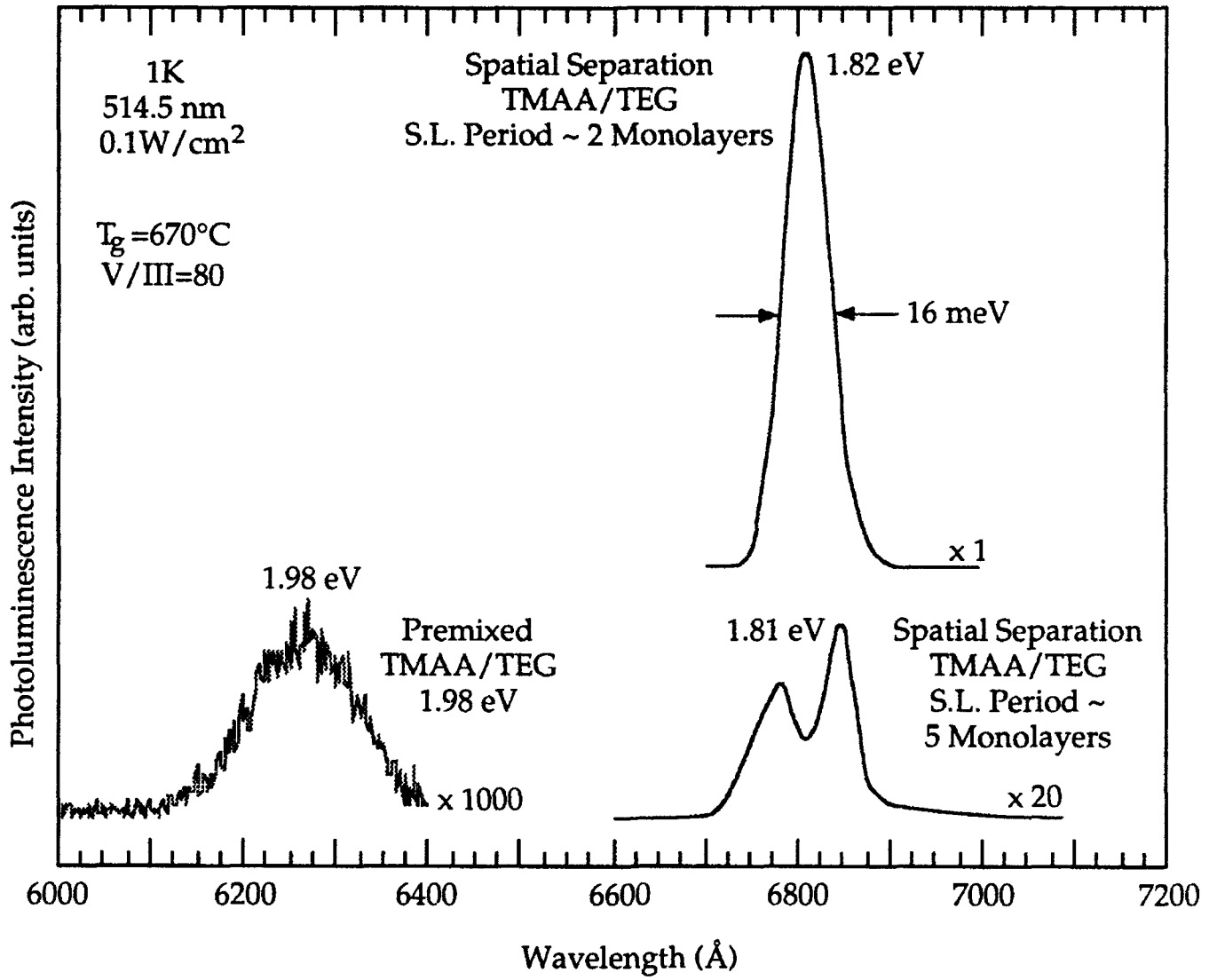

1K
514.5 nm
0.1W/cm2
Tg =670°C
V/III=80
Spatial Separation
TMAA/TEG
S.L. Period ~ 2 Monolayers
1.82 eV
16 meV
x 1
1.98 eV
Premixed
TMAA/TEG
1.98 eV
x 1000
1.81 eV
Spatial Separation
TMAA/TEG
S.L. Period ~
5 Monolayers
x 20
Photoluminescence Intensity (arb. units)
6000
6200
6400
6600
6800
7000
7200
Wavelength (Å)

Figure 8

REFERENCES

1. J.R. Shealy, V.G. Kreismanis, D.K. Wagner and J.M. Woodall, Appl. Phys. Lett. **83**, 246 (1982).
2. S.D. Hersee, M.A. Di Forte-Poisson, M. Baldy and J.P. Duchemin, J. Crystal Growth **55**, 53 (1981).
3. T.F. Kuech, D.J. Wolford, E. Veuhoff, V. Deline, P.M. Mooney, R. Potemski and J. Bradley, J. Appl. Phys. **62**, 632 (1987).
4. N. Kobayashi and T. Fukui, Electron. Lett. **21**, 888 (1984).
5. A.C. Jones and S.A. Rushworth, J. Crystal Growth **106**, 253 (1990).
6. J.S. Roberts, C.C. Button, J.P.R. David, A.C. Jones and S.A. Rushworth, J. Crystal Growth **104**, 857 (1990).
7. W.S. Hobson, T.D. Harris, C.R. Abernathy and S.J. Pearton, Appl. Phys. Lett. **58**, 77 (1991).
8. C.R. Abernathy, A.S. Jordan, S.J. Pearton, W.S. Hobson, D.A. Bohling and G.T. Muhr, Inst. Phys. Conf. Ser. No. 112: Chapter 3, 149 (1991).
9. C.R. Abernathy, A.S. Jordan, S.J. Pearton, W.S. Hobson, D.A. Bohling and G.T. Muhr, Appl. Phys. Lett. **56**, 2654 (1991).
10. N. Okamoto, H. Ando, A. Sandhu and T. Fugii, Jap. J. Appl. Phys. **30**, 3792 (1991).
11. W.S. Hobson, J.P. van der Ziel, A.F.J. Levi, J.O'Groman, C.R. Abernathy, M. Geva, L.C. Luther and V. Swaminathan, J. Appl. Phys. **70**, 432 (1991).
12. W.S. Hobson, F. Ren, M. Lamont Schnoes, S.K. Sputz, T.D. Harris, S.J. Pearton, C.R. Abernathy and K.S. Jones, Appl. Phys. Lett. **59**, 1975 (1991).
13. F. Kobayashi, S. Iio, T. Kuwahara and Y. Sekiguchi, Jap. J. Appl. Phys. **30**, 1971 (1991).
14. C.R. Abernathy, S.J. Pearton, F.A. Baiocchi, T. Ambrose, A.S. Jordan, D.A. Bohling and G.T. Muhr, J. Crystal Growth **110**, 457 (1991).
15. A. Kohzen, Y. Tohmori, Y. Akatsu and H. Kamada, J. Crystal Growth **124**, 70

(1992).

16. V.S. Sundaram, L.M. Fraas and C.C. Samuel, J. Elect. Mat. **21**, 1047 (1992).
17. A.S. Grady, R.D. Markwell, D.K. Russell and A.C. Jones, J. Crystal Growth **106**, 239 (1990).
18. N. Kobayashi, T. Makimoto, Y. Yamauchi and Y. Horikoshi, J. Appl. Phys. **66**, 640 (1989).
20. J.R. Shealy, J. Crystal Growth **87**, 350 (1988).
21. N. Saint-Cricp, G. Landa, J.B. Renucci, I. Hardy and A. Munoz-Yague, J. Appl. Phys. **61**, 1206 (1987).
22. C. Colvard, T.A. Gant, M.V. Klein, R. Merlin, R. Fischer, H. Morkoç and A.C. Gossard, Phys. Rev. B **31**, 2080 (1985).
23. A.C. Jones, Chemtronics **4**, 15 (1989).
24. W.J. Bartels and W. Nijman, J. Crystal Growth **44**, 518 (1978).
25. B.L. Pitts, D.T. Emerson and J.R. Shealy, Appl. Phys. Lett. **61**, 2054 (1992).
26. S.M. Olsthoorn, F.A.J.M. Driessen and L.J. Giling, Appl. Phys. Lett. **58**, 1274 (1991).
27. B.R. Bulter and J.P. Stagg, J. Crystal Growth **94**, 481 (1989).

TASK 2 FEMTOSECOND LASER STUDIES OF ULTRAFAST PROCESSES IN COMPOUND SEMICONDUCTORS

C. L. Tang

High-repetition-rate femtosecond pulse generation in the blue

R. J. Ellingson

School of Applied and Engineering Physics, Cornell University, Ithaca, New York 14853

C. L. Tang

School of Electrical Engineering, Cornell University, Ithaca, New York 14853

Received November 20, 1991

We report the generation of high-repetition-rate femtosecond pulses in the blue by intracavity doubling of a mode-locked Ti:sapphire laser using β-BaB_2O_4. To reduce the pulse-broadening effect of group-velocity mismatch, an extremely thin β-BaB_2O_4 crystal is used. By pumping the Ti:sapphire laser with 4.4 W of power from an Ar^+ laser, as much as 230 mW of 430-nm light is produced at a 72-MHz repetition rate and a 89-fs pulse width. This represents an effective conversion efficiency of ~75% from the typical infrared output to the second harmonic. Pulse widths as short as 54 fs are achieved for the blue output.

Extension of the wavelength range accessible to femtosecond pulses has been a topic of much interest. The two techniques used most frequently to generate <100-fs pulses at otherwise unattainable wavelengths are continuum generation and frequency conversion with the use of crystals. Femtosecond pulse generation techniques based on amplification followed by continuum generation permit tunability from the UV into the IR.[1] However, amplification reduces the pulse repetition rate to the order of a kilohertz, and there is often a loss of time resolution in the final pulse. In contrast, frequency conversion in crystals can maintain the high repetition rate of the femtosecond megahertz-rate laser and requires only a single cw pump laser. The higher repetition rate results in much smaller pulse fluctuation and excellent experimental signal-to-noise ratios.

In recent years, much progress has been made in extending the spectral range of high-repetition-rate femtosecond pulses throughout the visible and IR by using frequency conversion in crystals. The 80-MHz femtosecond optical parametric oscillator permits broad tunability throughout the near IR and mid-IR.[2,3] High-repetition-rate femtosecond pulse generation in the UV and blue–green has been somewhat more limited. Colliding-pulse mode-locked (CPM) lasers have directly generated <100-fs pulses in the range of 493 to 554 nm at milliwatt outputs,[4,5] and intracavity doubling of the Rhodamine 6G/diethyloxadiacarbocyanine iodide (Rh6G/DODCI) CPM dye laser has resulted in a 100-MHz source of femtosecond pulses with milliwatt outputs in the 310–315-nm range. The Rh6G/DODCI CPM laser was first intracavity doubled by using KDP.[6] Soon thereafter, β-BaB_2O_4 (BBO) was used to intracavity double the CPM laser with a per-pass conversion efficiency as high as 5.5%, which generated 20 mW of UV output per arm with <100-fs pulse widths, and pulse widths as short as 43 fs.[7] This gives an effective conversion efficiency of nearly 100% from the typical CPM output in the red to the UV.

While the standard Rh6G/DODCI CPM dye laser operates at a wavelength slightly shorter than the tuning range of the Ti:sapphire laser, the broad tunability, the high average output power, and the obvious advantages of a solid-state laser have made the dispersion-compensated mode-locked Ti:sapphire laser[8] an extremely attractive replacement for the CPM dye laser. At present, the mode-locked Ti:sapphire laser can potentially operate with <200-fs pulse widths and >100-mW average power over the range of 700 to 1053 nm.[9] Frequency doubling over this spectral range provides femtosecond pulses from 350 to 525 nm. Doubling of the Ti:sapphire laser outside the cavity has been reported.[10] The best conversion efficiency of 25% was achieved at 750 nm, although no second-harmonic pulse widths were reported and the length of the doubling crystal was not given. The group-velocity mismatch for type I second-harmonic generation (SHG) in BBO at 750 nm is 225 fs/mm, and in order to maintain the narrowest temporal pulse width a thin doubling crystal is required. Use of a thin crystal therefore necessitates a high peak power to achieve high conversion efficiency, and thus intracavity doubling is required to achieve simultaneously the shortest pulses and the highest power in the second harmonic. As discussed further below, extremely high intracavity conversion efficiency is possible, which would result in UV, blue, or green outputs of hundreds of milliwatts average power. Using an extremely thin (55 μm) crystal of BBO, we demonstrate a 72-MHz repetition-rate source of blue pulses of 89-fs duration (FWHM) and 115 mW of power per arm (two arms of BBO; see Fig. 1). Reducing the pulse width for the blue to 54 fs, we measure 45 mW of power per arm.

Figure 1 shows a schematic of the dispersion-compensated intracavity-doubled Ti:sapphire laser. The SF-10 prisms are spaced 50 cm tip to tip. The

0146-9592/92/050343-03$5.00 0

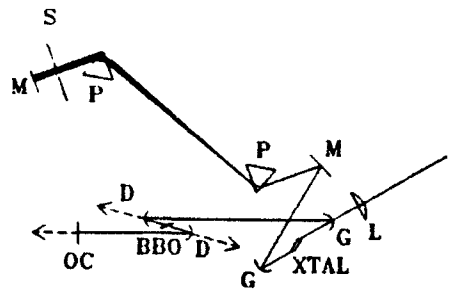


Fig. 1. Schematic of the intracavity doubled Ti:sapphire laser. XTAL, Ti:sapphire crystal; G's, gain mirrors; L, focusing lens; P's, SF-10 prisms; M's, flat mirrors; D, dichroic mirror; BBO, doubling crystal; S, tuning slit; OC, output coupler.

argon pump laser is focused by a 10-cm focal-length lens through one of the $r = 10$ cm gain mirrors onto the 18-mm-long titanium-doped (0.1%) sapphire crystal. The additional intracavity focus at the BBO crystal consists of $r = 5$ cm dichroic mirrors (fused-silica substrates, $R = 100\%$ at 860 nm, $T = 70\%$ at 430 nm). The outcoupler has $T = 1\%$ for the IR and was replaced by a high reflector when the highest power in the blue was generated. Before insertion into the laser cavity, the crystal is aligned for maximum SHG conversion efficiency in the extracavity beam of the mode-locked Ti:sapphire laser operating at the intended doubling wavelength of ~860 nm. The proper alignment of the BBO can be preserved on insertion into the laser cavity.

Pulse-width measurements for both the fundamental (IR) and the second-harmonic light are made by autocorrelation with collinear type I SHG in BBO. The BBO crystal used to measure the IR autocorrelation has a thickness of 0.8 mm and is cut for a phase-matching angle of $\theta = 27.5°$. The BBO crystal used to measure the blue pulse widths has a thickness of 0.67 mm and is cut at $\theta = 69°$. The second harmonic of the blue (215 nm, the fourth harmonic of the Ti:sapphire) is passed through a 0.2-m monochromator and detected by a solar-blind photomultiplier tube. The spectra for the fundamental and second-harmonic outputs from the laser are measured by using a 0.25-m monochromator to disperse the light onto an optical multichannel analyzer.

We point out that the type I SHG cutoff wavelength in the blue for BBO is 409 nm. Below this wavelength, accurate pulse-width measurement requires a more difficult technique such as cross correlating the fundamental beam with the second-harmonic beam by using phase-matched sum-frequency generation. Owing to the significant group-velocity mismatch between the fundamental and second-harmonic pulses for fundamental wavelengths below 820 nm (the group-velocity mismatch is >170 fs/mm for BBO at $\lambda_{IR} = 820$ nm and increases for shorter wavelengths), a thin cross-correlation crystal is required.[7] Thus, for the convenience of using collinear type I SHG autocorrelation to measure the pulse width of the doubled light, we operated the Ti:sapphire laser at $\lambda > 820$ nm.

The intracavity-doubled mode-locked laser is started by a slight mechanical perturbation, usually by a small-amplitude, gentle back-and-forth translation of one prism. Once well aligned, the mode-locked laser operates stably indefinitely (observed for as much as ~6 h), although significant mechanical perturbation can stop mode-locked operation. The mode locking generally is not self-starting. Variation of the intracavity dispersion compensation permits control of the pulse width. On starting, the laser is pushed to shorter pulses simply by adding prism glass and adjusting the focusing slightly to maintain high stability. While the laser stability is excellent even at the longer pulse widths, the oscilloscope trace of the IR mode-locked pulse train indicates somewhat quieter operation as the pulse width is decreased. The spatial mode of the fundamental beam is TEM_{00} with faint, simple higher-order modes superimposed. The blue beam mode is a clean TEM_{00} that shows no sign of higher-order modes, thus verifying that the power of the fundamental lies almost entirely in the TEM_{00} mode.

When the laser is run with a high reflector in place of the outcoupler, 107-fs IR pulses produce 230 mW of second harmonic. Without the intracavity doubling crystal, the maximum output of the mode-locked Ti:sapphire laser operating at 860 nm is ~300 mW for 4.4-W pump power; thus generation of 230 mW of blue power gives an effective conversion efficiency of ~75% from the IR output typical at this pump power. The dichroic mirrors transmit ~72 mW of power per arm of the blue second-

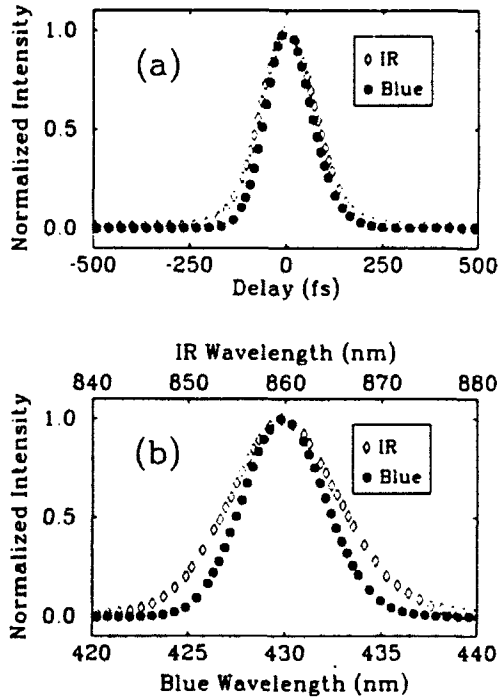


Fig. 2. (a) Autocorrelation data for the fundamental and second-harmonic pulses in the longer-pulse limit. The FWHM for the fundamental is 107 fs, and for the second harmonic it is 89 fs. (b) Spectra for the fundamental and second-harmonic beams. The FWHM for the fundamental is 12.7 nm, which gives $\Delta\nu\Delta t = 0.55$, and for the second harmonic it is 4.9 nm, which gives $\Delta\nu\Delta t = 0.71$.

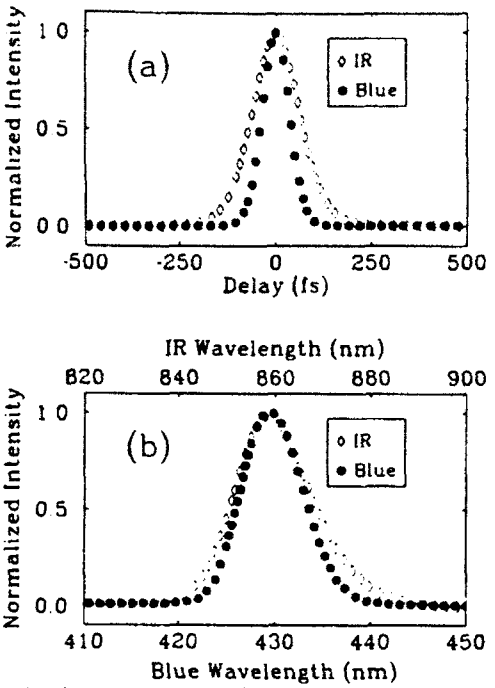


Fig. 3. (a) Autocorrelation data for the fundamental and second-harmonic pulses for the shortest second-harmonic pulses. The FWHM for the fundamental is 93 fs, and for the second harmonic it is 54 fs. (b) Spectra for the fundamental and second-harmonic beams. The FWHM for the fundamental is 18.6 nm, and for the second harmonic it is 7.7 nm. This gives $\Delta\nu\Delta t = 0.70$ for the fundamental and $\Delta\nu\Delta t = 0.67$ for the blue second-harmonic pulses.

harmonic light. On compression of the blue pulses by a dispersion-compensating prism pair, a pulse width of 89 fs is measured (see Fig. 2). The prism pair allows compensation for the dispersion of the dichroic mirror substrate and of other intracavity optics as well as for any upchirp that the pulses may have on generation in the intracavity BBO crystal. The IR pulses are not extracavity dispersion compensated. The spectral FWHM's of the IR and blue are 12.7 and 4.9 nm, respectively, which give $\Delta\nu\Delta t = 0.55$ for the IR and $\Delta\nu\Delta t = 0.71$ for the blue pulses. Pulse widths and time–bandwidth products are determined assuming a $\mathrm{sech}^2(t)$ intensity envelope.

We achieved the shortest blue pulses when running the laser with a 1% outcoupler in place of the high reflector and operating closer to net zero intracavity group-velocity dispersion (see Fig. 3). The power of the IR coupled out is 27 mW, whereas the blue power transmitted by the dichroic mirrors is ~31 mW per arm. The extracavity dispersion-compensated blue pulses have a FWHM of 54 fs and a spectral FWHM of 7.7 nm, which gives $\Delta\nu\Delta t = 0.67$. The IR pulses (which again are not extracavity dispersion compensated) have a pulse width of 93 fs and a spectral FWHM of 18.6 nm, which yields $\Delta\nu\Delta t = 0.70$. It is believed that the IR pulses may be compressed by an extracavity two-prism sequence, and we hope to verify this in the near future. Again, a $\mathrm{sech}^2(t)$ intensity envelope is assumed.

The observed intracavity SHG conversion efficiency of 3.2% per pass for the shortest blue pulses agrees well with the theory (3.5%) for conversion by a nondepleted pump wave.[11] Without the intracavity BBO crystal, we have observed stable mode-locked operation for <100-fs pulses at intracavity powers as high as 8 W. For the same focusing and BBO crystal length presented here, 8 W of intracavity power at a 110-fs pulse width would yield a more than fourfold increase in the output of the second harmonic, or ~500 mW of blue light. For this case, the peak intracavity intensity at the focus would approach the reported single-shot damage threshold for BBO of 50 GW/cm^2.[12] However, this threshold pertains to pulses of 8-ns duration, and we expect the threshold to increase by orders of magnitude for the 100-fs pulse-width regime. The average intensity is orders of magnitude below the long-term damage threshold for BBO.[12]

In conclusion, we have demonstrated highly efficient intracavity doubling of a mode-locked Ti:sapphire laser that yields a source of femtosecond pulses in the blue with the same high repetition rate of 72 MHz, short pulse width, excellent beam quality, and power in the blue representing appreciable recovery of the typical IR output at this 4.4-W pump level. This research represents an extension of intracavity doubling to solid-state mode-locked lasers and results in a source of femtosecond pulses potentially tunable from the near UV into the green, thus broadly expanding the potential spectral range for femtosecond pulses.

The authors thank W. S. Pelouch, P. E. Powers, and D. C. Edelstein for helpful conversations. This research was supported by the Joint Services Electronics Program and the National Science Foundation.

References

1. R. L. Fork, C. V. Shank, C. Hirlimann, R. Yen, and W. J. Tomlinson, Opt. Lett. **8**, 1 (1983).
2. D. C. Edelstein, E. S. Wachman, and C. L. Tang, Appl. Phys. Lett. 54, 1728 (1989).
3. E. S. Wachman, W. S. Pelouch, and C. L. Tang, J. Appl. Phys. **70**, 1893 (1991).
4. P. M. W. French and J. R. Taylor, Opt. Lett. **13**, 470 (1988).
5. P. M. W. French, M. M. Opalinska, and J. R. Taylor, Opt. Lett. **14**, 217 (1989).
6. G. Focht and M. C. Downer, IEEE J. Quantum Electron. **24**, 431 (1988).
7. D. C. Edelstein, E. S. Wachman, L. K. Cheng, W. R. Bosenberg, and C. L. Tang, Appl. Phys. Lett. **52**, 2211 (1988).
8. D. E. Spence, P. N. Kean, and W. Sibbett, Opt. Lett. **16**, 42 (1991).
9. For example, the Coherent MIRA laser.
10. Y. Ishida, N. Sarukura, and H. Nakano, in *Digest of Conference on Lasers and Electro-Optics* (Optical Society of America, Washington, D.C., 1991), paper JMB2.
11. A. Yariv, *Quantum Electronics* (Wiley, New York, 1975), p. 431, Eq. 16.7-3, where this equation is divided by 4 for P^{ω} representing the total pump beam rather than by one half of the pump for each mixing wave, and the author has included the factor of ϵ_0 in d_{eff}.
12. H. Nakatani, W. R. Bosenberg, L. K. Cheng, and C. L. Tang, Appl. Phys. Lett. **53**, 2587 (1988).

Ti:sapphire-pumped, high-repetition-rate femtosecond optical parametric oscillator

W. S. Pelouch, P. E. Powers, and C. L. Tang

Department of Applied Physics, Cornell University, Ithaca, New York 14853

Received March 31, 1992

A broadly tunable femtosecond optical parametric oscillator (OPO) based on $KTiOPO_4$ that is externally pumped by a self-mode-locked Ti:sapphire laser is described. Continuous tuning is demonstrated from 1.22 to 1.37 μm in the signal branch and from 1.82 to 2.15 μm in the idler branch by using one set of OPO optics. The potential tuning range of the OPO is from 1.0 to 2.75 μm and requires three sets of mirrors and two crystals. Without prisms in the OPO cavity, 340 mW (475 mW) of chirped-pulse power is generated in the signal (idler) branch for 2.5 W of pump power. The total conversion efficiency as measured by the pump depletion is 55%. With prisms in the cavity, pulses of 135 fs are generated, which can be shortened to 75 fs by increasing the output coupling.

Optical parametric oscillators (OPO's) have recently been exploited in the femtosecond time domain as a source of broadly and continuously tunable radiation. The lack of suitable pump sources has hampered the development of femtosecond OPO's that operate with short pulse widths, a high repetition rate, and high output powers. The high peak power at the intracavity focus of a colliding-pulse mode-locked dye laser was exploited to develop the first femtosecond OPO.[1-3] This resulted in ≥105-fs, 80-MHz pulses at approximately 3 mW of output power. Other researchers resorted to a Q-switched and mode-locked laser (300 pulses at 15 Hz) to pump an OPO producing ≥160-fs pulses (65 fs at one wavelength) at 4.5 mW of average power.[4] More recently a femtosecond OPO was reported that was externally pumped by a hybridly mode-locked dye laser producing 220-fs pulses at 30 mW of average power.[5] In this Letter we describe a Ti:sapphire-pumped OPO capable of producing 75-fs pulses at a high repetition rate (90 MHz) and hundreds of milliwatts of average output power. We believe that these are the shortest tunable pulses ever generated from an OPO.

The Ti:sapphire pump laser is configured in a linear cavity with a 18-mm titanium-doped (0.1%) sapphire crystal and SF-14 prisms (spaced at 40 cm) for dispersion compensation. The crystal is mounted in a copper block and cooled by using a thermoelectric cooler with temperature feedback to maintain a constant 20°C temperature. The laser is self-mode locked as described elsewhere in the literature[6] and produces 2.5 W of 125-fs pulses in a TEM_{00} mode when pumped by a 15-W argon-ion laser. A schematic of the OPO cavity is shown in Fig. 1. The Ti:sapphire laser beam is focused onto a 1.15-mm KTP crystal with polarization along the y axis using a $r = 15$ cm curved high reflector. The pump suffers approximately a 5% transmission loss for each side of the crystal. The KTP crystal is cut at $\theta = 47.5°$ and $\phi = 0°$ for type II phase matching ($o \rightarrow e + o$) and coated with a 250-nm layer of MgF_2 on both sides for high transmission centered at 1.3 μm. The OPO cavity uses two $r = 10$ cm curved mirrors that are aligned for oscillation in the x-z plane of the crystal to provide compensation for walk-off between the Poynting vectors of the pump and the resonated signal branch.[3] The cavity may be aligned with or without the SF-14 prism sequence simply by lowering or raising the prism assembly. The output coupler is 1%, and the other flat mirror is mounted on a piezoelectric transducer for fine length adjustment. A linear cavity design was chosen so that the pump can be retroreflected for double-pass pumping of the KTP crystal.[4] This would result in parametric gain for the signal in both directions through the crystal when the retroreflected pump pulses overlap the signal pulses in the crystal. However, this requires that an optical isolator be inserted between the pump laser and the OPO to reject feedback into the Ti:sapphire cavity.

The OPO is aligned by monitoring the spontaneous parametric scattering using a liquid-nitrogen-cooled germanium photodiode [the peak detectivity is $\sim 10^{13}$ cm $Hz^{1/2}$ W^{-1} at 1.5 μm]. This signal is maximized by adjusting the OPO mirrors and focusing such that the spontaneous parametric scattering makes many round trips in the cavity. Oscillation occurs when the cavity length of the OPO is matched to that of the pump laser cavity; the length mismatch becomes more sensitive near threshold.

With 2.5 W of pump power (125 fs) the OPO produces as much as 340 mW of power in the signal branch through the 1% output coupler. We have measured 60 mW of signal energy reflected from the KTP crystal in one direction (120-mW loss per round trip), which implies a transmission loss of 0.2%. Thus 460 mW of power is generated in the signal branch with an effective output coupler of 1.4%. In the idler branch we have coupled out 475 mW of power, but this may be limited by the physical constraints of collecting and collimating the diverging

0146-9592/92/151070-03$5.00/0

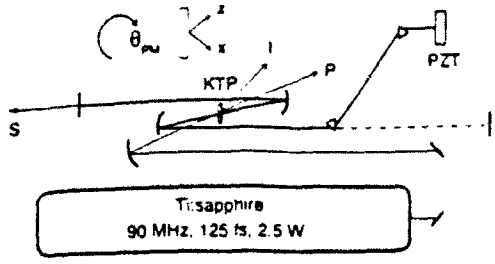


Fig. 1. Schematic of the OPO cavity in the vertical plane. The Ti:sapphire pump (P) is focused onto the 1.15-mm KTP crystal. An enlarged view of the crystal is depicted above and shows the orientation for type II phase matching at the phase-matching angle θ_{PM}. The signal branch (S) is resonated by using a 1% output coupler and a piezoelectric transducer (PZT) for fine length adjustment. The idler (I) exits from the crystal at ~6 deg from the signal. The prism sequence may be raised to allow oscillation without the prisms.

idler radiation that is generated at ~6 deg (external to the crystal) from the signal. The pump is depleted by 55% when the OPO is oscillating and is a measure of the actual conversion efficiency; this value agrees well with the measured power output of the OPO if the crystal reflections and the pump transmission losses are taken into account. Double-pass pumping has not yet been implemented in the OPO since excellent conversion efficiency has already been achieved. If only one pass of the pump were used, then a ring cavity would provide less loss than the linear cavity.

Interestingly, the OPO also produces output at two other non-phase-matched[7] wavelengths that correspond to collinear second-harmonic generation of the signal branch ($e + e \rightarrow e$) and noncollinear sum-frequency generation between the pump and the signal ($o + e \rightarrow o$). For a pump wavelength of 780 nm and a signal wavelength of 1300 nm the second-harmonic wavelength is 650 nm and the sum-frequency wavelength is 485 nm. A total of almost 100 mW of second-harmonic power is generated (50 mW in each direction), but only 10 mW gets transmitted through the infrared optics and output coupler. The collinear second harmonic could be utilized for experimental purposes and is also useful for aligning the signal through extracavity optics, after which it can be easily filtered out. 100 μW of sum-frequency light was measured after the output coupler. In all, the OPO system produces synchronized femtosecond radiation at five different wavelengths.

Without prisms in the OPO cavity the autocorrelation and spectra show signs of significant chirp. The pulse width as measured from the intensity autocorrelation is approximately 500 fs owing to the long decay time of the wings. With prisms in the OPO cavity two regimes are encountered. For net negative group-velocity dispersion (GVD) the pulses are unchirped with a minimum pulse width of 135 fs (fit to a sech^2 shape) and have a smooth spectrum ($\Delta\nu\Delta\tau = 0.45$) [see Figs. 2(a) and 2(b)]. For net positive GVD the pulses are slightly chirped with a broader pulse width and a split spectrum [see Figs. 2(c) and 2(d)]. Near zero GVD the OPO may abruptly flip into either the chirped or unchirped mode. This behavior is in contrast to the observed smooth transition between operation with net negative and positive GVD of the OPO reported in Ref. 2. Therefore a nonlinear chirp must be generated in the KTP, which accounts for the runaway condition in the positive-GVD regime. This would also explain why the time–bandwidth product is 45% greater than the transform limit for the minimum pulse width. This effect is most likely due to self-phase modulation of the signal in the crystal as a result of the high intracavity intensity and large nonlinear index of KTP. Self-phase modulation in KTP was identified as a source of broadening of the pump laser in Ref. 1 and is consistent with the shape of the signal spectrum in Fig. 2(c).[8] It is expected that the pulse widths are approximately constant over the tuning range owing to the relatively constant inverse group-velocity mismatch between the pump and the signal. The larger mismatch for the idler suggests pulse widths approximately 50% greater than the signal.

It was also observed in the unchirped regime that a slight detuning of the length shortened the pulse widths to approximately 75 fs (and reduced the output power by 25%). The pulse width was also decreased to 75 fs by increasing the output coupling at constant zero detuning. This was achieved by inserting a thin glass flat in the OPO cavity and rotating it away from Brewster's angle, effectively reducing the intracavity power by increasing the output coupling to 1.5% (plus 0.4% from the crystal). Therefore this pulse shortening results from a decrease in intracavity power as the OPO is operated closer to threshold, as predicted by theory.[9] The reduction in intracavity power reduces the magnitude of self-phase modulation (both linear and nonlinear chirp) so that less dispersion compensation is required from the prism sequence.

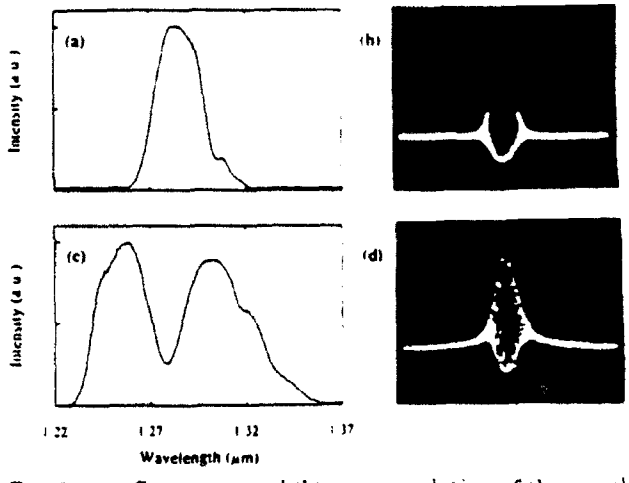


Fig. 2. (a) Spectrum and (b) autocorrelation of the signal pulse for net negative GVD. The time–bandwidth product is 0.45. (c) Spectrum and (d) autocorrelation of the chirped signal pulse for net positive GVD. The abrupt transition between these two regimes suggests a self-phase-modulation process in the crystal.

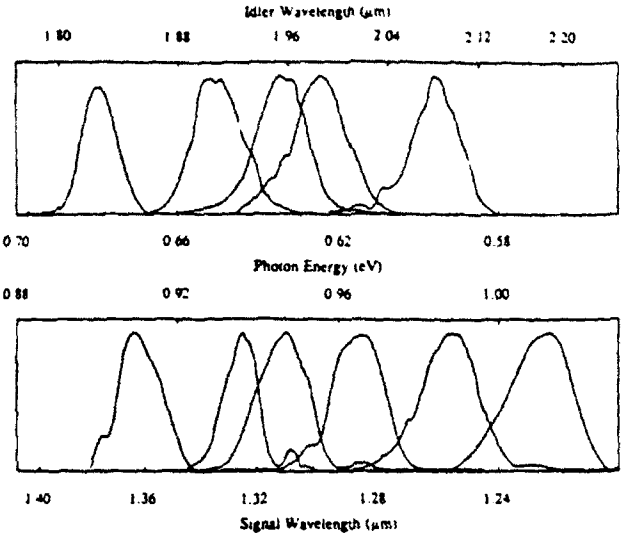


Fig. 3. OPO signal (bottom) and idler (top) spectra obtained by angle tuning the OPO over a range of one set of mirrors. Broad tuning may also be achieved by changing the pump wavelength without rotating the KTP crystal or altering the OPO alignment.

The insertion of the Brewster-cut prism sequence reduces the output power of the signal to 300 mW in the chirped regime, but we believe that with a more careful alignment full recovery of the 340 mW is possible. This loss is primarily due to a small rotation of the signal polarization in the KTP crystal, which is oriented slightly away from $\phi = 0°$. The output power for the unchirped pulses is reduced to approximately 180 mW. This loss of power is not due to simple alignment since the prism is only translated.

Tuning of the OPO is straightforward and may be accomplished by three different means. Adjustment of the length mismatch of the OPO cavity results in a wavelength shift as reported previously[1] and may be used to stabilize the OPO length at a fixed wavelength. The wavelength range over which the OPO will oscillate while the length is adjusted is a measure of how sensitive the OPO is to length variations. The OPO can withstand a 5-μm length variation, which results in a wavelength shift of almost 50 nm. Second, a change in the pump wavelength will tune the OPO without changing the crystal orientation or OPO alignment—only the length of the OPO cavity must be adjusted to match the new pump cavity length. We can tune our Ti:sapphire laser from 765 to 815 nm while maintaining mode locking and cavity alignment. This results in tuning of the signal branch from 1.22 to 1.34 μm and from 2.05 to 2.08 μm in the idler branch. Note that the wavelength of the idler remains relatively fixed, whereas the signal tunes over 120 nm as the pump wavelength is varied over 50 nm. Typically this type of tuning will also result in a change in pump power. Third, the OPO may be tuned in the traditional manner by adjusting the phase-matching angle of the KTP crystal. We can tune over a 100-nm range by freely rotating the KTP crystal and adjusting the cavity length. Beyond this range the OPO alignment needs to be modified. The operation of the OPO is quite robust so that broad tuning is accomplished by iterating between rotating the crystal and adjusting the OPO alignment while maintaining oscillation. Representative spectra are displayed in Fig. 3 for both the signal and the idler. The demonstrated tuning is limited by the optics available in our laboratory, but with appropriate optics the full tuning range will be accessible.

No alignment of the OPO is necessary on a day-to-day basis; length adjustment is all that is required to regain oscillation. Furthermore the OPO is not extremely sensitive to pump steering. Alignment of the pump through two pinholes suffices to recover oscillation if the Ti:sapphire alignment is considerably altered. The output of the OPO is an excellent TEM_{00} mode that is made possible by the tight Z focus shown in Fig. 1. Thus the OPO is a practical laser source for experimental ultrafast research. A feedback circuit to maintain length matching would be useful to maximize stability, although all the data presented in this Letter were obtained without any length stabilization.

In summary, we have reported the development of a high-power, high-repetition-rate femtosecond OPO externally pumped by a self-mode-locked Ti:sapphire laser. More than 1.0 W of the pump laser power is converted to tunable OPO radiation for a conversion efficiency of 55%. Unchirped pulses of 135 fs can be generated across the demonstrated tuning range of the device. Pulse shortening to 75 fs is achieved by increasing the output coupling at the expense of output power.

This research was supported by the Joint Service Electronics Program and the National Science Foundation. We are grateful to L. K. Cheng and J. D. Bierlein of E. I. DuPont de Nemours & Company for providing the KTP material.

Note added in proof: We recently generated nearly transform-limited 57-fs signal pulses at an output power of 115 mW.

References

1. D. C. Edelstein, E. S. Wachman, and C. L. Tang, Appl. Phys. Lett. **54**, 1728 (1989).
2. E. S. Wachman, D. C. Edelstein, and C. L. Tang, Opt. Lett. **15**, 136 (1990).
3. E. S. Wachman, W. S. Pelouch, and C. L. Tang, J. Appl. Phys. **70**, 1893 (1991).
4. R. Laenen, H. Graener, and A. Laubereau, Opt. Lett. **15**, 971 (1990).
5. G. Mak, Q. Fu, and H. M. van Driel, Appl. Phys. Lett. **60**, 542 (1992).
6. See, for example, D. E. Spence, P. N. Kean, and W. Sibbett, Opt. Lett. **16**, 42 (1991).
7. The non-phase-matched process was previously observed by D. C. Edelstein, Ph.D. dissertation (Cornell University, Ithaca, NY., 1990).
8. E. M. Wright, J. Opt. Soc. Am. B **7**, 1142 (1990).
9. E. C. Cheung and J. M. Liu, J. Opt. Soc. Am. B **7**, 1385 (1990).

TASK 3 ULTRAFAST INTERACTION OF CARRIERS AND PHONONS IN NARROW BANDGAP SEMICONDUCTOR STRUCTURES

C. R. Pollock

Femtosecond pulse generation by using an additive-pulse mode-locked chromium-doped forsterite laser operated at 77 K

Alphan Sennaroglu, Timothy J. Carrig, and Clifford R. Pollock

School of Electrical Engineering, Cornell University, Ithaca, New York 14853

Received April 13, 1992

Using an acousto-optically mode-locked chromium-doped forsterite laser, operated at 77 K and coupled to a nonlinear resonator containing a single-mode fiber, we have produced femtosecond pulses of 150-fs duration at 1.23 μm with useful output powers of approximately 60 mW. This represents what is to our knowledge the first demonstration of femtosecond pulse generation from this laser system using the coupled-cavity mode-locking scheme.

The chromium-doped forsterite laser ($Cr:Mg_2SiO_4$) is based on the Cr^{4+} ion in a tetrahedrally coordinated lattice site serving as the laser-active center. First demonstrated by Petričević *et al.*,[1] the laser emission, centered at 1.23 μm, was shown to be tunable over as broad a range as from 1.13 to 1.37 μm.[2] This feature, in conjunction with ample output powers, makes the Cr:forsterite laser a useful source for the optical characterization of fiber-optic systems at 1.3 μm and spectroscopic studies of narrow band-gap semiconductors. To date, room-temperature Q-switched,[1,2] cw,[3] flash-lamp-pumped,[2,4] cw acousto-optically mode-locked,[5] synchronously pumped,[5] and cw cryogenic[6] operations have been demonstrated with various optical pumping mechanisms. In particular, our previous experiments revealed an approximately threefold increase in cw output power when the gain medium was cooled to 77 K (Ref. 6) (for pump powers well above threshold), resulting in cw output powers as high as 2.8 W at 1.23 μm when the system was pumped by a cw Nd:YAG laser.[7] The Cr:forsterite laser used in the mode-locking experiments described in this Letter was also operated cryogenically to achieve increased power outputs. Furthermore the broad emission bandwidth of this laser can also be utilized for generating ultrashort light pulses on a femtosecond scale. Such pulses are ideal for applications in short-pulse propagation experiments and femtosecond time-resolved spectroscopy.

In this Letter we report what is to our knowledge the first demonstration of additive-pulse mode-locked operation of an actively mode-locked Cr:forsterite laser. Using this technique, we have produced pulses of 150-fs duration (FWHM) at 1.23 μm with useful output powers of approximately 60 mW.

Additive-pulse mode locking (APM), a well-established scheme for generating ultrashort light pulses, has been successfully applied to many solid-state laser systems (see Ref. 8 and references therein for a thorough discussion). Briefly, in its most commonly practiced form, this technique, also known as coupled-cavity[9,10] or interferential[11] mode locking, involves coupling the master laser resonator to an external nonlinear cavity containing an optical fiber. The auxiliary fiber cavity, in which propagating light pulses acquire a Kerr-effect-induced phase shift, can be regarded as a nonlinear termination equivalent to a mirror with an intensity-dependent reflectivity. Once the nonlinear phase shift is adjusted to give constructive interference at the center and destructive interference in the wings of the master cavity and coupled-cavity pulses when they combine at the output coupler of this composite optical resonator, a dramatic reduction in the output pulse width results provided that the two cavities are interferometrically matched in length. To date, APM has been demonstrated in KCl:Tl°,[12,13] $LiF:F_2^+$,[12] $NaCl:OH^-$,[14] Ti:sapphire,[15,16] Nd:YAG,[17] Nd:YLF,[18] and Nd:glass[19] lasers. As described in what follows, we have applied this scheme to generate femtosecond pulses from the Cr:forsterite laser.

Figure 1 shows the experimental setup of the coupled-cavity Cr:forsterite laser used for the APM experiments. The master cavity, consisting of a flat high reflector (HR1), a flat 11% transmitting output coupler (O.C.), an acousto-optic prism mode locker (M.L.), and a pair of 5-cm focal-length antireflection-coated plano–convex lenses (L1 and L2) around the gain medium, was end pumped by a cw Nd:YAG laser (Quantronix Model 416). The gain medium was a 20 mm × 5 mm × 5 mm piece of forsterite crystal cut along the a, b, and c axes (using the P_{nma} crystallographic notation), with the longest dimension along the c axis. The estimated laser-active center concentration was 4×10^{18} cm^{-3}. To prevent deleterious étalon effects, the normal-cut crystal was polished with a slight wedge between the 5-mm-sided square faces, which were also broadband antireflection coated at 1.28 μm. The crystal was maintained in an evacuated Dewar (pressure $\sim 10^{-6}$ Torr) at 77 K with the plano–convex lenses L1 and L2 serving as the Dewar windows. Using 5 W of input pump power and a 70-cm focal-length mode-matching lens between the pump and the Cr:forsterite laser, we obtained 625 mW of cw TEM_{00} output power with the output field polarized

0146-9592/92/171216-03$5.00 0

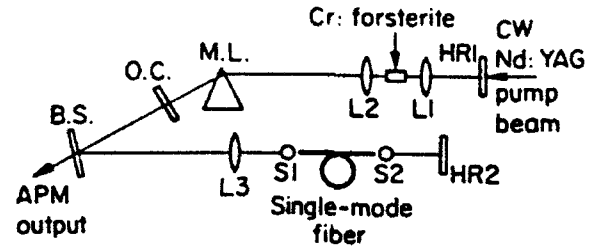


Fig. 1. Experimental setup of the APM Cr:forsterite laser. The Cr:forsterite crystal was maintained at 77 K inside an evacuated Dewar with lenses L_1 and L_2 serving as the Dewar windows.

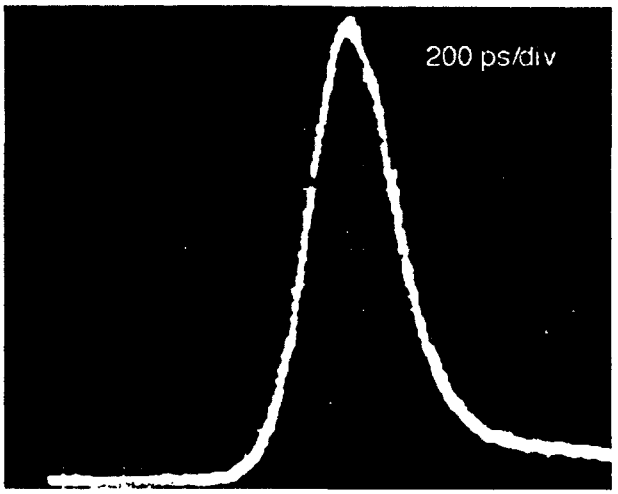


Fig. 2. Oscilloscope trace of the actively mode-locked pulses from the master laser resonator using the 11% transmitting output coupler.

along the *a* axis. The Cr:forsterite crystal had 78% absorption at the pump wavelength of 1.06 μm at 77 K. An asymmetric cavity configuration used to prevent possible double pulsing effects of the mode-locked laser together with a choice of comparably shorter focal-length lenses around the gain medium resulted in less than optimum mode matching between the pump and the laser cavities and hence lower output power than what was reported in Ref. 6.

Before employing the nonlinear coupled-cavity scheme to produce femtosecond pulses, we actively mode locked the master laser resonator, using a quartz acousto-optic modulator in the form of a Brewster-cut prism (Crystal Technology), placed within 3 cm of the output coupler. With approximately 2 W of absorbed rf power at 40.999 MHz, the laser was acousto-optically mode locked and generated output pulses at a 82-MHz repetition rate. The individual acousto-optically mode-locked pulses were monitored by a high-speed InGaAs detector with a response time of approximately 80 ps connected to a sampling oscilloscope with a response time of less than 30 ps. With a 1% transmitting output coupler, detector-limited pulse widths of 80 ps (FWHM) were measured, indicating that the actual pulses were shorter and comparable with what was reported by Seas *et al.*[5] However, with the 11% transmitting output coupler, used in the APM experiments, the minimum pulse width obtained was 320 ps (FWHM), as shown in Fig. 2.

The nonlinear coupled cavity was established by using an 85% reflecting beam splitter (B.S.). The single-mode fiber (Corning 1521) of length 50.8 cm placed in this external cavity had zero group-velocity dispersion at 1.3 μm and a mode-field diameter of 9 μm. The fiber ends were cleaved with tilt angles of less than 0.5 deg to the surface normal. Using coupling spheres (S1 and S2) with antireflection coating on the input side and index-matching gel between the output side and the fiber surface, together with an antireflection-coated mode-matching lens (L3), we obtained coupling efficiencies of approximately 70%. A flat high reflector mirror (HR2) placed a distance from the output end of the fiber provided the nonlinear feedback with retroreflection efficiencies approaching 90%. The output of the APM Cr:forsterite laser was monitored by using three separate diagnostics: a Michelson interferometer with a $LiIO_3$ nonlinear crystal to measure the collinear and background-free intensity autocorrelations of the pulses, a Ge photodiode with a 2-ns rise time to investigate the pulse train over the 50-ns to 200-μs time scale, and a scanning spectrometer (Monolight Model 6000) to measure the bandwidth of the output pulses.

When the two cavity lengths were interferometrically matched, enhanced mode-locked operation of the Cr:forsterite laser was observed. Figures 3 and 4 show the background-free intensity autocorrelation and the spectrum of the APM pulses, respectively. With the assumption of a $sech^2$ intensity profile, the width (FWHM) of the pulses was measured to be 150 fs. A simultaneous measurement of 18-nm bandwidth gave a time–bandwidth product of approximately 0.55, roughly 1.7 times larger than the theoretical limit of 0.32 for the assumed pulse shape. The output of the laser was at 1.23 μm.

It was found that with the above mirror reflectivities and coupling retroreflection efficiencies, a threshold power level of 40 mW coupled through the

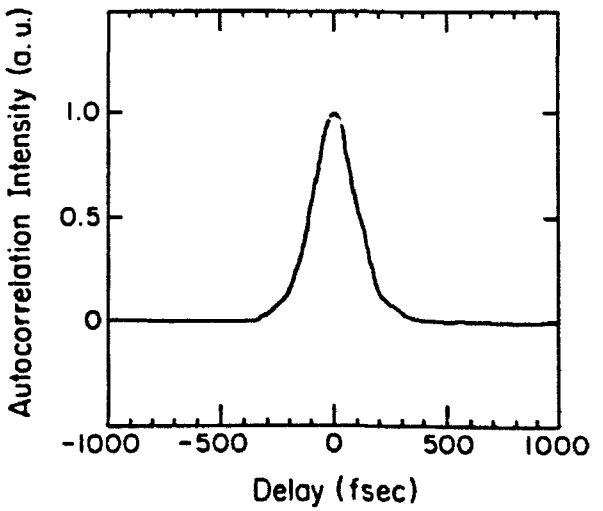


Fig. 3. Background-free intensity autocorrelation of the APM Cr:forsterite pulses. The measured FWHM is 150 fs.

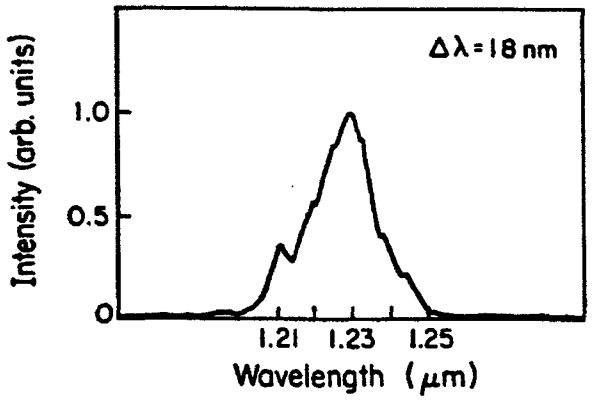


Fig. 4. Spectrum of the APM Cr:forsterite pulses. The resulting time–bandwidth product is 0.55.

fiber was required for observation of the onset of APM operation. Close to and somewhat above this threshold, the external cavity length had to be perturbed about its correct value for APM action to be observed. When the fiber power was increased to well above 40 mW, the pulses became more stable and could be sustained without the use of active cavity-length stabilization. By simultaneously monitoring the output using the Ge photodiode, we found that near the 40-mW threshold the APM output pulse train appeared as a series of repetitively Q-switched pulses, each of 700-ns duration occurring at 143-kHz repetition rate. For power levels three times above the threshold, repetitive Q switching gave way to a quiet, stable pulse train with occasional weak relaxation oscillations. Similar turn-on behavior has been observed by Spielmann *et al.*[20] regarding the self-starting APM Nd:glass laser. Coupled fiber power levels of as much as 280 mW were tried, higher power levels being avoided to prevent possible damage to fiber ends. This resulted in 63 mW of useful output power at 1.23 μm. Using the relevant laser and fiber parameters, we found the calculated peak nonlinear phase shift[21] between the center and the wings of the external cavity pulses to change from 2.7π at the 40-mW threshold to 19π when the fiber power was 280 mW. Within the 10% error associated with the measurements, the pulse width of the APM Cr:forsterite laser remained essentially insensitive to the variation in this nonlinear phase shift.

It was observed that incomplete mode locking, in the form of pulses with excess amplitude noise or spiky structure, would give rise to unsatisfactory APM action, resulting in broader pulses. It was therefore essential to get clean, spike-free mode-locked operation of the master laser resonator as depicted in Fig. 2 to obtain femtosecond pulses.

In conclusion, we have demonstrated, for what is to our knowledge the first time, additive-pulse mode-locked operation of the Cr:forsterite laser that produces 150-fs pulses at 1.23 μm with useful output powers of approximately 60 mW. It was also observed that near the threshold of the APM action, the mode-locked pulse train came as a series of repetitively Q-switched pulses of 700-ns duration occurring at 143-kHz repetition rate. For power levels sufficiently above the threshold, however, a stable, quiet pulse train of femtosecond pulses was produced.

This research was supported by the National Science Foundation under grant ECS-9111838, the New York State Science and Technology Foundation, and the Materials Science Center at Cornell University.

References

1. V. Petričević, S. K. Gayen, R. R. Alfano, K. Yamagishi, H. Anzai, and Y. Yamaguchi, Appl. Phys. Lett. **52**, 1040 (1988).
2. V. G. Baryshevskii, M. V. Korzhik, A. E. Kimaev, M. G. Livshitz, V. B. Pavlenko, M. L. Meilman, and B. I. Minkov, Zh. Prikl. Spectrosk. **53**, 7 (1990).
3. V. Petričević, S. K. Gayen, and R. R. Alfano, Opt. Lett. **14**, 612 (1989).
4. A. Sugimoto, Y. Segawa, Y. Yamaguchi, Y. Nobe, K. Yamagishi, P. H. Kim, and S. Namba, Jpn. J. Appl. Phys. **28**, L1833 (1989).
5. A. Seas, V. Petričević, and R. R. Alfano, Opt. Lett. **16**, 1668 (1991).
6. T. J. Carrig and C. R. Pollock, Opt. Lett. **16**, 1662 (1991).
7. T. J. Carrig and C. R. Pollock, in *Digest of Conference on Advanced Solid-State Lasers* (Optical Society of America, Washington, D.C., 1992), pp. 23–25.
8. H. A. Haus, J. G. Fujimoto, and E. P. Ippen, J. Opt. Soc. Am. B **8**, 2068 (1991).
9. W. Sibbett, in *Ultrafast Phenomena VII*, C. B. Harris, E. P. Ippen, G. A. Mourou, and A. H. Zewail, eds., Vol. 53 of Springer Series in Chemical Physics (Springer-Verlag, Berlin, 1990), pp. 2–7.
10. P. A. Belanger, J. Opt. Soc. Am. B **8**, 2077 (1991).
11. M. Morin and M. Piché, Opt. Lett. **14**, 1119 (1989).
12. P. N. Kean, X. Zhu, D. W. Crust, R. S. Grant, N. Langford, and W. Sibbett, Opt. Lett. **14**, 39 (1989).
13. J. Mark, L. Y. Liu, K. L. Hall, H. A. Haus, and E. P. Ippen, Opt. Lett. **14**, 48 (1989).
14. C. P. Yakymyshyn, J. F. Pinto, and C. R. Pollock, Opt. Lett. **14**, 621 (1989).
15. P. M. W. French, J. A. R. Williams, and J. R. Taylor, Opt. Lett. **14**, 686 (1989).
16. J. Goodberlet, J. Wang, J. G. Fujimoto, and P. A. Schulz, Opt. Lett. **14**, 1125 (1989).
17. J. Goodberlet, J. Jacobson, J. G. Fujimoto, P. A. Schulz, and T. Y. Fan, Opt. Lett. **15**, 504 (1990).
18. J. M. Liu and J. K. Chee, Opt. Lett. **15**, 685 (1990).
19. F. Krausz, Ch. Spielmann, T. Brabec, E. Wintner, and A. J. Schmidt, Opt. Lett. **15**, 737 (1990).
20. Ch. Spielmann, F. Krausz, T. Brabec, E. Wintner, and A. J. Schmidt, IEEE J. Quantum Electron. **27**, 1207 (1991).
21. G. P. Agrawal, *Nonlinear Fiber Optics* (Academic, San Diego, Calif., 1989), Chap. 4.

Femtosecond electron relaxation in InGaAs lattice-matched to InP

David Cohen and Clifford R. Pollock

Cornell University, School of Electrical Engineering
Ithaca, NY 14853

ABSTRACT

Carrier energy relaxation times have been measured in $In_{0.53}Ga_{0.47}As$ grown by MBE on InP. Layer thicknesses from 0.5 to 3 microns have been studied. An NaCl color center laser using additive pulse modelocking supplied 150 femtosecond pulses with photon energies between 780 and 806 meV. These were used for time resolved optical saturation measurements near the 750 meV material bandgap. Carrier densities between 0.4×10^{18} and 5.7×10^{18} were achieved. Lifetimes of about 150 femtoseconds are reported. These are observed to decrease with increasing carrier density and with decreasing photon energy.

1. INTRODUCTION

The bandgap of the InGaAs/InP sytem at 1.55 microns has made it a useful material for optoelectronic device fabrication. In addition, its high mobility suggests the possibility of fabricating extremely fast devices. This has been done, for example, in a heterojunction bipolar transistor[1]. Fast pin photodiodes are also being developed in InGaAs. It is therefore useful to characterize the carrier lifetimes. Previously, photoluminescence upconversion has been applied to measure longer time scale relaxation rates[2,3,4], as well as femtosecond pump-continuum probe methods[5,6]. Both used photon energies well above the bandgap. In this work, near-bandgap measurements were made well below the intervalley scattering threshold, using the equal pulse correlation technique to extract the lifetime from the transient optical saturation of different samples. Exploiting the tuneability of the NaCl color center laser, these experiments were performed at several photon energies.

InGaAs is a direct gap material, and the bandgap InGaAs/InP is well known to be 750 meV at 300K. The next lowest transition occurs at more than 3 times the photon energy [7]. for which the split-off band separation is about 343 meV. Therefore only the direct Γ transition to the conduction band from the light- and heavy-hole valence bands is significant in these experiments; split-off holes cannot participate in the low-energy transitions excited by our laser. For this system, the longitudinal optical phonon energy is 34 meV.

Experimental results from 3 samples are reported here. 3, 1, and 0.5um films of $In_{0.53}Ga_{0.47}As$ grown by MBE on an Fe-doped InP substrate were studied. Note that the InP substrate's bandgap of 1.4 eV makes it quite transparent to the wavelengths of the NaCl laser. The substrate was, however, lightly polished to minimize scattering from the substrate. The transparency of the InP was verified with a Cary 5 spectrophotometer. For wavelengths longer than 1 micron, it revealed a smooth, resonance-free transmission spectrum for an InP sample taken from the same wafer as that used to grow our samples. All experiments were carried out at 300K.

2. EQUAL PULSE CORRELATION SPECTROSCOPY

Equal pulse correlation spectroscopy uses two identical excitation pulses derived from the same source but delayed relative to each other . The time-averaged absorption in the sample is then a symmetrical function of delay. More precisely, the experiment measures the convolution of the material response with the second order autocorrelation of the laser pulse[8]. This assumes that the sample is optically thin, that is,

$$(L/\alpha) << 1 \qquad (1)$$

where L is the sample thickness, and α is the absorption depth.

For a linear response function R(t), and a second-order pulse autocorrelation function f(t), the equal pulse correlation signal takes the form

$$S(\tau) = \int_0^\infty ds\ R(s)\,[f(\tau - s) + f(\tau + s)] + \int_0^\infty ds\ R(s)\,[c(s, \tau) + c(s, -\tau)] \qquad (2)$$

where $c(t, \tau)$ models the coherent response to the rapidly varying electric fields. For transform-limited pulses, the coherent response term is negligible for delays longer than one and a half pulse widths. Most of the useful information about the sample is contained within the first term. Note that the response is symmetrical in delay τ. The nature of the equal pulse correlation signal is illustrated in figures 1 and 2 for 100 femtosecond pulses convolved with 50 and 200 femtosecond decay functions.

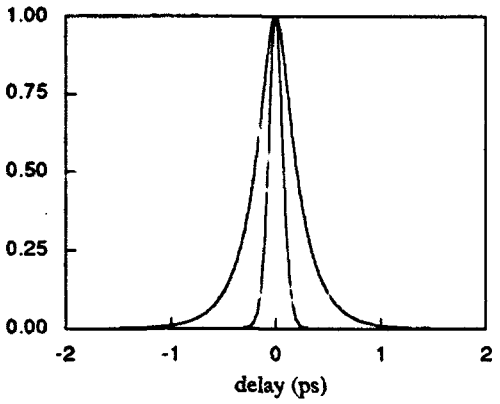


Fig. 1. The convolution of eqn. 2, with the autocorrelation function superimposed. Pulsewidth is 100 fs, and decay time is 200 fs.

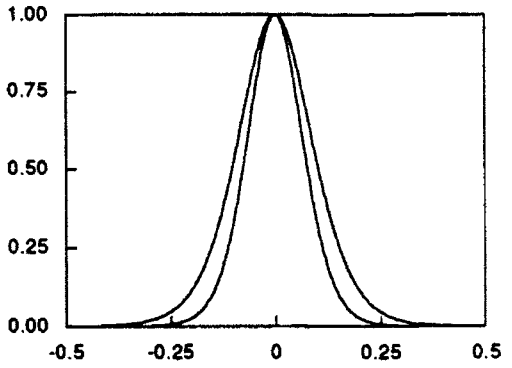


Fig. 2. Same as Fig. 1, but with 50fs lifetime.

Typical pump probe spectroscopy also gives a convolution of autocorrelation and response function, but information near zero delay is distorted by the order reversal of saturating and probing pulses. Equal pulse correlation allows simpler fitting when the decay times observed are close to the excitation pulse width.

3. Experiment

The experiment is laid out as a Michelson interferometer, with one arm mounted on a galvanometer-driven taut band translator[9] which gives a smooth sinusoidal variation of the optical delay. Both pulses had parallel polarizations. Only the linear region of the delay, near zero crossing, is used to collect data. The spatially overlapped pulse trains from each Michelson arm are then attenuated by a rotatable antireflection coated linear polarizer followed by a fixed polarizing beamsplitter used to define a constant polarization state for our experiments. The light is then split into a reference and signal beam. The latter is focussed onto a germanium photodiode for use in noise suppression. The first beam is focussed to an 8 micron spot on the sample. The light transmitted through is collected by a lens and focussed onto a second, identical germanium photodiode. Both detectors are preceded by neutral density filters to balance the photocurrents and also to minimize detector nonlinearity.

The large amount of amplitude modulation (5-10%) present on the output of the additive pulse modelocked laser requires that some form of noise cancelation be used. A well-known passive approach has worked best to date: subtracting from the

nonlinear response signal a signal proportional to the instantaneous laser intensity. This is easily accomplished using the photodiodes sampling optical intensity before and after the sample, as described above. The two photodiodes are directly connected so as to subtract their photocurrents. Using a variable neutral density filter to balance the average photocurrents results in excellent subtraction of laser amplitude fluctuations. Little decrease in cancellation efficiency occurs near zero delay, since the nonlinearity is small, only about 2%.

The difference photocurrent is used as the input to a transimpedance amplifier (Ithaco model 1211) which provides a voltage proportional to the difference current. It also serves to filter out fast interferometric oscillations in the data. This voltage is averaged synchronously with the delay variation, using a 12 bit a/d converter. Five hundred to fifteen hundred averages were used to obtain the traces presented here.

The source used in these experiments was a sodium chloride color center laser using additive pulse modelocking. The characteristics of this laser have been reported elsewhere[10]. This laser was used to produce 100 - 200 femtosecond pulses from 1.54 microns to 1.59 microns in this experiment. Up to 100 miliwatts of output power is available, at a repetition rate of 164 MHz. In the wavelengths reported in this paper, the laser pulses are approximately transform limited.

4. Fitting

Extracting the carrier lifetime proved challenging, since the carriers clearly relaxed on a time scale comparable to the pulsewidth. Simply fitting the data tails (data well separated from zero delay) to a sum of exponentials is problematic in this case. It was decided instead to fit to the convolved model dexcribed earlier. The dominant decay is clearly on the order of 150 femtoseconds or less, so only one exponential was used in the fit. This is not meant to imply that longer decays are not present - work is still progressing on refining the analysis. The sodium chloride additively pulse modelocked laser has previously been demonstrated to produce transform limited pulses near 1.55 micron wavelengths. The second harmonic autocorrelation trace of these pulses fit a hyperbolic secant function rather well. Therefore the measured second harmonic autocorrelation trace width was used as a fixed fit parameter. All fitting was done starting 300 femtoseconds after the zero delay point. This ensured that the coherent artifact did not distort the results. Good fits to the data were obtained for a ratio of fit function peak to data peak of 1: 2. This ratio, which corresponds to the amount of coherent artifact present, was therefore fixed in our analysis at this value. Data was taken out to a delay of 1.7 picoseconds on either side of zero delay.

5. Results

All of our experiments show fast decay times. A representative equal pulse correlation trace is shown in figures 3 and 4.

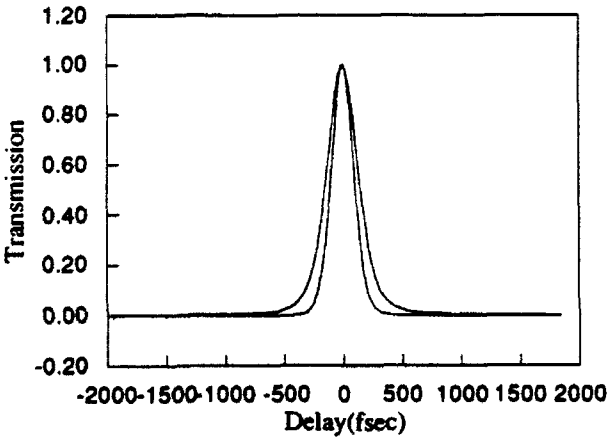


Fig. 3. A representative equal pulse autocorrelation trace taken with InGaAs, taken at a wavelength of 1.596 um .

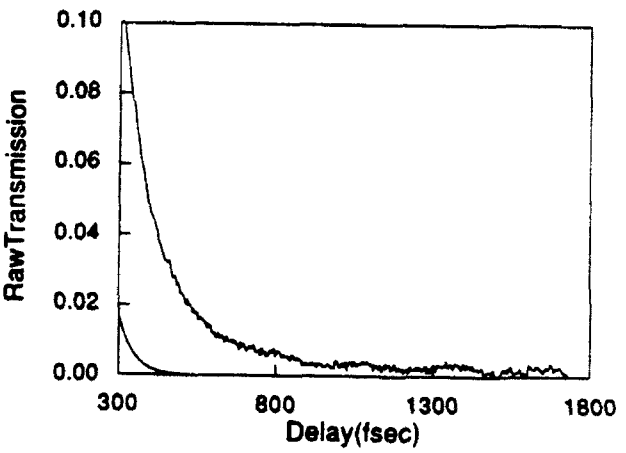


Fig. 4. An expanded view of the part of the trace used for fits. The modulation is residual laser noise.

Lifetimes in the 0.5 micron and 3 micron samples range between 100 and 200 femtoseconds. There is a very clear decrease in lifetime with increasing carrier density. This is attributable to carrier-carrier scattering. There is also a decrease in lifetime with decreasing photon energy. At the same time, the slope of the lifetime-carrier density curves decreases with decreasing photon energy. The lifetime is almost constant at 120 femtoseconds for 780 meV excitation, the lowest photon energy reported here. The results are summarized in figures 5 and 6 below.

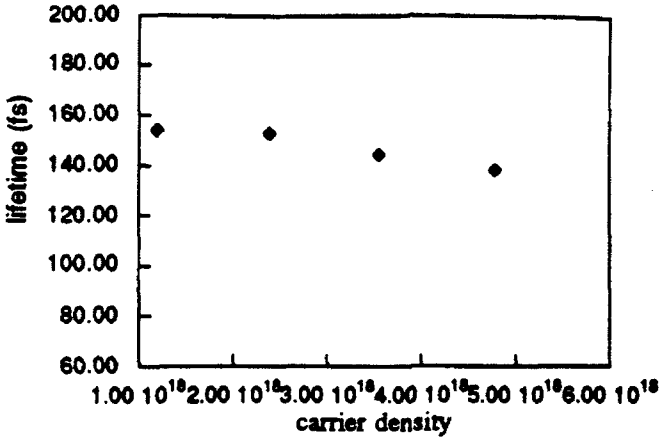


Fig. 5. Convolution fit results for the 0.5 micron thick sample. The incident pulsewidth was 153 fs, and the excitation wavelength was 1.549 μm.

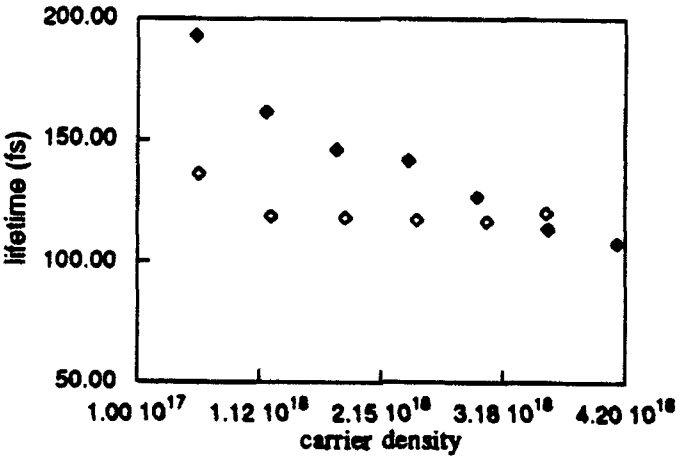


Fig. 6. Convolution fits for a 1 μm sample, with 137 fs pulsewidths incident. The upper curve corresponds to an excitation wavelength of 1.539 μm, while the lower curve corresponds to a wavelength of 1.592 μm.

The 3 micron sample begins to violate the assumption of an optically thin sample (the Beer's law absorption depth in InGaAs is 2.5 um). Somewhat faster lifetimes are returned by the convolution fit, which may be attributable to the sample acting like a saturable absorber [11]. These lifetime fit results are in figure 7.

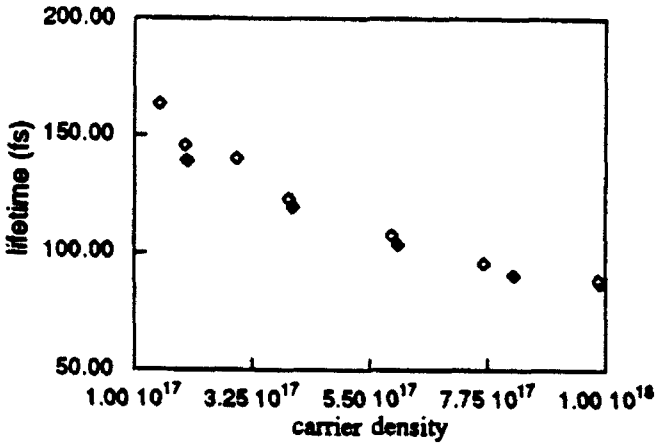


Fig. 7. Convolution fits for a 3 μm thick sample. The closed circles are the results for a wavelength of 1.544 μm, and the open diamonds are the results for a wavelength of 1.574 μm.

6. Summary

We have optically measured carrier lifetimes in $In_{.53}Ga_{.47}As$ grown by MBE on InP. Measurements were made at several photon energies just above the bandgap. Carrier densities ranging from 3×10^{17} to 4×10^{18} were created in the sample. Lifetimes of about 150 femtoseconds were found, with carrier-carrier scattering appearing to increase with increasing carrier densities. Lifetimes were found to decrease somewhat with decreasing distance of the photoexcitation energy from the band edge, and the dependence on carrier density also decreased.

7. Acknowledgements

This research was supported by the Joint Services Electronics Program, contract number F49620-87-C-0044. The authors especially wish to thank Dr. Uwe Happek, for his many invaluable contributions. We also appreciate useful interactions with Janet Machol, Jim Bair, and Dr. Bill Schaff.

8. References

1. T. Carruthers, I. Duling III, O. Aina, M. Mattingly, M. Serio, "Responses of an InP/InGaAs/InP heterojunction bipolar transistor to 1530 and 620 nm ultrafast optical pulses", Appl. Phys. Lett., 59, 3, p. 327, 15 July 1991.
2. J. Shah, R.F. Leheny, R.E. Nahory, and M.A. Pollack, "Hot-carrier Relaxation in photoexcited InGaAs", Appl. Phys. Lett., 37, 5, p. 475, 1 Sept. 1991.
3. K. Kash and J. Shah, "Carrier Energy Relaxation in InGaAs as Determined from Picosecond Luminescence Studies", Appl. Phys. Lett. 45, 4, p. 401, 15 Aug. 1984.
4. K. Kash and J. Shah, "Hot Electron Relaxation in InGaAs", J. Lumin., 30, p. 333, 1985.
5. H. Lobentanzer, W. Stolz, K. Ploog, "Cooling of Carriers in Three- and Two-Dimensional GaInAs", Ultrafast Phenomena VI, T. Yajima, K. Yoshihara, C. Harris, S. Shionaya, eds., p. 297, Springer-Verlag, Heidelberg, 1988.
6. W. Kutt, K. Seibert, H. Kurz, "High Density Femtosecond Excitation of Hot Carrier Distributions in InP and InGaAs", Ultrafast Phenomena VI, T. Yajima, K. Yoshihara, C. Harris, S. Shionaya, eds., p. 233, Springer-Verlag, Heidelberg, 1988.

7. E.W, Williams and V. Rehn, "Electroreflectance Studies of InAs, GaAs, and (Ga,In)As Alloys", Phys. Rev., **172**, 3, p. 798, 15 Aug. 1968.
8. A.J. Taylor, D.J. Erskine, C.L. Tang, "Equal-Pulse Correlation Technique for Measureing Femtosecond Excited State Relaxation Times", Appl. Phys. Lett., **43**, 11, p. 989, 1 Dec. 1983.
9.
10. C.P. Yakymyshn, J.F. Pinto, C.R. Pollock, "Additive Pulse Modelocked NaCl $F_2^+O^{2-}$ Laser", Opt. Lett., **14**, p. 621, 1989.
11. Our thanks to J. Bair for suggesting this effect.

Generation of 48 fsec pulses and measurement of crystal dispersion by using a regeneratively-initiated self-mode-locked chromium-doped forsterite laser

Alphan Sennaroglu and Clifford R. Pollock

School of Electrical Engineering, Cornell University, Ithaca, NY 14853

Howard Nathel

Lawrence Livermore National Laboratory, Livermore, California 94550

Tel: 510-423-3262 Fax: 510-422-2252 E-mail: HNathel@llnl.gov

Abstract

Regeneratively-initiated, self-sustained, mode-locked operation of a chromium-doped forsterite laser operated at 3.5 ^{o}C is described. By employing intracavity, negative group velocity dispersion compensation, nearly transform-limited femtosecond pulses of 48 fsec (FWHM) duration were generated with average TEM_{00} output powers of 380 mW at 1.23 μm. Regenerative-initiation provides improvement in the output stability and ease of operation compared to fixed frequency AO modulators. By tuning the mode-locked laser in the range 1.21-1.26 μm, estimated values for forsterite dispersion constants have also been obtained for the first time. The demonstrated power and stability open the door to applications such as efficient second harmonic generation.

Generation of 48 fsec pulses and measurement of crystal dispersion by using a regeneratively-initiated self-mode-locked chromium-doped forsterite laser

Alphan Sennaroglu and Clifford R. Pollock

School of Electrical Engineering, Cornell University, Ithaca, NY 14853

Howard Nathel

Lawrence Livermore National Laboratory, Livermore, California 94550

Among the recently developed novel techniques of ultrashort pulse generation, self-mode-locking has become widely used and applied to several tunable solid-state lasers to produce femtosecond pulses. First demonstrated in the Ti:sapphire laser by Spence *et al.*[1], this scheme has been shown to work in other solid-state laser hosts including Nd:YLF [2], Cr^{3+}:$LiSrAlF_6$ [3], chromium-doped forsterite (Cr:forsterite) [4], Nd:YAG [5], and Cr^{3+}:$LiCaAlF_6$ [6]. Soliton-type pulse shaping mechanisms, where intensity dependent Kerr nonlinearities in the gain medium producing positively chirped pulses are balanced by prism pair negative group velocity dispersion, give rise to stable femtosecond pulse trains in these lasers. A variety of initiation techniques such as continuous-wave (cw) self-mode-locking [1], regenerative initiation [7-9], synchronous pumping [10], and acousto-optical modulation [11] have been used to set the initial intensity conditions necessary for the soliton-like pulse shaping to take place.

The broad gain bandwidth of the Cr:forsterite laser makes it a suitable candidate for the generation of ultrashort pulses. To date, acousto-optically mode-locked [12], synchronously pumped [12], acousto-optically initiated self-mode-locked [4], and additive-pulse mode-locked [13] modes of operation have been demonstrated. Seas *et al.* [4] reported the shortest pulses to date of 60 fsec (FWHM) duration using acousto-optically initiated self mode locking with intracavity group velocity dispersion (GVD) compensation. They reported that 90 fsec pulses were more routinely generated, suggesting to us that some pulsewidth instabilities were present. They reported only 85 mW of average output power.

In this paper, we describe the performance of a regeneratively initiated, self-sustainable, mode-locked Cr:forsterite laser operated at 3.5 ^{o}C that is pumped by a cw Nd:YAG laser. Regenerative mode-locking eliminates the need for synchronicity between the acousto-optic modulator rf drive signal and the cavity repetition frequency. In our experience with acousto-optic mode locking of a forsterite laser, maintaining this synchronicity was extremely critical for useful output. When cavity length drift occured, not only did the pulsewidth increase, but large fluctuations in the average power were observed. Regenerative initiation eliminated these problems. Regenerative modulation uses a portion of the cavity beat signal to drive the acousto-optic modulator electronics, thus obviating the need for stringent cavity length control. It also allows the in situ measurement of cavity dispersion. Once pulse shaping is initiated, a very stable train of femtosecond pulses develops due to the balance between intensity-dependent Kerr-induced nonlinearities and the intracavity dispersion of the cavity. As Seas *et al.* [4] demonstrated, Cr:forsterite is capable of operating in this self-sustained mode

once the pulses are initiated.

Unique to our work is the improvement in operating stability provided by regenerative initiation, the generation of significantly shorter nearly transform-limited pulses (48 fsec FWHM duration), and a significant increase in average TEM_{00} output power (380 mW at 1.23 μm). These represent to our knowledge the shortest and highest peak power pulses directly generated from this laser system. Furthermore, using the cavity dispersion measurement technique developed by Knox [14], the second and third order dispersion constants in the lasing range of the forsterite crystal have been measured for the first time. The combination of high power, reliable operation, and cavity dispersion measurements open the door to shorter pulse generation and applications such as efficient second harmonic generation of femtosecond pulses in the 615 nm region.

The experimental set-up of the regeneratively initiated self-mode-locked Cr:forsterite laser is shown in figure 1 and is similar to the laser described in reference 4 except for the cavity length, crystal length, output coupler, prism seperation, and method of acousto-optic initiation. The folded, astigmatically compensated laser resonator consisted of a flat wedged high reflector (M3) and a 3.5 % transmitting output coupler (O.C) of 157 cm radius of curvature with the gain medium positioned slightly off-center between a pair of high reflecting curved mirrors (M1 and M2) each of 5 cm focal length and separated by 10.8 cm. The laser mirrors were obtained from the optics division of Spectra Physics Lasers, Inc. and were broadband coated for operation between 1.15 and 1.35 μm. A regeneratively driven acousto-optic modulator (A.O.M) was placed near the output coupler. A pair of prisms (P1 and P2) placed on the high reflector side were used for dispersion compensation. The total cavity length was 185 cm corresponding to a longitudinal mode spacing of

81.265 MHz. A cw Nd:YAG laser (Quantronix model 416) operated at 1.064 μm was mode-matched and focussed into the forsterite crystal using an anti-reflection (AR) coated, bi-convex lens (L1) of 10 cm focal length through M1 having 93.3% transmission at 1.064 μm. A half-wave plate (W.P.) at 1.064 μm was used to adjust the pump polarization to obtain optimum power output from the laser.

The gain medium, a 4mm x 4mm x 12mm Brewster cut forsterite crystal with 0.3% chromium concentration, was oriented with the crystal a-axis (P_{nma} crystallographic notation) in the plane of incidence of a p-polarized electric field. The crystal was obtained from IFC, Inc.. The crystal was wrapped in indium foil and tightly clamped between copper plates to facilitate rapid heat exchange. A thermoelectric cooler with a feedback loop, maintained the crystal temperature at 3.5 ^{o}C with peak temperature fluctuations less than 0.2 ^{o}C. The careful temperature control of the crystal was crucial in obtaining a stable train of femtosecond pulses. Temperature fluctuations of a few degrees gave rise to as much as 50% power fluctuations over 100 μsec time scales when the laser was being pumped well above threshold. A plexiglass enclosure surrounding the crystal holder assembly was purged with dry nitrogen gas to minimize water condensation on the crystal surfaces. The gain medium had 70.9% absorption at 1.064 μm at the operating temperature of 3.5 ^{o}C.

With a 3.5% transmitting output coupler, 6.5W of pump absorbed, and a crystal temperature of 3.5 ^{o}C, the output power of the laser running in cw mode (no prisms, no A.O.M) was 420 mW. The output wavelength of the laser was centered at 1.23 μm. The absorbed pump power slope efficiency at low pump power levels was measured to be 10.4%, the threshold pump

power being 1.6 W. For absorbed pump powers beyond 5W, the slope efficiency started to level off due to increased thermal loading of the forsterite crystal. Alignment of the focussing mirrors was critical to quiet operation. Beyond pump power levels of 5W, the cw output power sometimes displayed chaotic power fluctuations. We believe this was due to thermal lensing induced by the pump beam. The fluctuations could be fully overcome by carefully translating the mirror M1.

The laser was first mode-locked without employing intracavity dispersion compensation. The regenerative mode-locking scheme is similar to that described in reference 8. The cavity loss was modulated using a regenerative acousto-optic mode-locker which had 0.4 % modulation depth and a 0.5 W RF amplifier. The acousto-optic modulator(A.O.M) (NEOS Technologies, Inc. model N12040-2-LIT-BR-IN)), used a 1 cm long Brewster angled quartz crystal operated off resonance. Approximately 4% of the laser output power was sent to an InGaAs photodiode to produce a signal for the regenerative mode-locker electronics. Inclusion of the A. O. modulator caused approximately 6% reduction in the total cw output power of the laser. A portion of the signal from the InGaAs detector was also sent to a Hewlett Packard model 5328A 500 MHz universal frequency counter to precisely register the pulse repetition rate. The mode-locked output of the laser was analyzed using a scanning spectrometer (Monolight model 8000) with approximately 2.5 nm wavelength resolution and an autocorrelator with a 2 mm thick $LiIO_3$ doubling crystal. The spectrum and autocorrelation signals were acquired using a Tektronix model 2230 500 MHz digital storage oscilloscope and recorded by an interfaced computer.

We observed three distinct modes of operation. Using no intracavity

dispersion compensation, and for cw output powers below 280 mW corresponding to 4.3 W of absorbed pump power, 41 psec FWHM pulses (assuming a Gaussian pulse shape) were obtained from the Cr:forsterite laser. The pulse width measured is in agreement with what was previously reported by Alfano's group [12], and is very close to that predicted from active mode-locking theory for chirp-free pulses [15], which was calculated to be 44 psec.

Increasing the absorbed pump power beyond 4.3 W, which increased the output power of the laser, resulted in pulses of 6.5 psec (FWHM) duration. Again a Gaussian pulse shape was assumed. As much as 380 mW cw TEM_{00} output power at 1.23 μm was obtained while the laser maintained this output pulse width. Due to the limited resolution of the scanning spectrometer, the bandwidth of the mode-locked pulses could not be fully resolved. We believe the shorter pulses at higher absorbed pump power are evidence of intracavity intensity induced nonlinear effects (i.e. self-phase-modulation) in the gain medium. Self-phase-modulation gives rise to increased bandwidth of the pulses which can support the shorter pulse widths. Because no intracavity dispersion compensation was employed, we believed that these 6.5 psec pulses had excess frequency chirp and hence were not transform-limited, as observed in [4].

To compensate for the positive second order dispersion in the cavity, a pair of SF-14 Brewster angled prisms (P1 and P2) were placed on the high reflector (M1) side of the cavity. The prism separation was 48 cm, slightly longer than that reported by Seas *et al.* [4], which is due to the longer Cr:forsterite crystal used in this work. Prior to observing femtosecond pulse generation, the laser resonator was first aligned at a low pump power level to

obtain optimum cw output power. Subsequently, the pump power was increased beyond the threshold level for self-phase-modulation (4.3 W) with the regenerative mode-locker operating to initiate the femtosecond pulse train. Once initiated, the laser produced a very stable uninterrupted train of femtosecond pulses. No apertures or other means of starting such as tapping on the table were necessary. The TEM_{00} output power of the laser was 380mW with the spectrum centered at 1.23 μm. Figure 2 and 3 show the noncollinear intensity autocorrelation and the spectral width of the femtosecond pulses respectively. Assuming a $sech^2$ [16] intensity profile the pulsewidth (FWHM) was measured to be 48 fsec. The overall dispersive broadening due to the output coupler and the autocorrelator optics was estimated to be less than 2 fsec for this 48 fsec pulse at 1.23 μm. A simultaneous measurement of 33.7 nm bandwidth gave a measured time-bandwidth product of 0.321 indicating that the pulses were nearly transform-limited and free of excess frequency chirp. We believe that higher intracavity power levels (28%higher, 2.77MW) were the predominant factor in obtaining pulses shorter than what was previously reported [4]. With the regenerative mode-locker off, self-sustained operation up to 2 minutes was observed. Cessation of the mode-locked operation was believed to be due to micromechanical perturbations of the system. The cavity repetition rate was stable to better than 40 Hz and could be varied by changing the cavity length in the range [81.2300-81.3200 MHz] without interrupting the mode locking process. The peak output power per pulse was determined to be 97 kW.

The mode-locked laser was tuned in the range 1.211-1.264 μm by translating a slit between the prism P2 and high reflector M3. Using the frequency counter, the pulse repetition rate was measured as a function of wavelength. By employing the cavity dispersion calculation technique

developed by Knox [14], and by accounting for the known dispersion of the AO cell and the prism pair, the second and third order dispersion constants of forsterite at 1.23 μm were determined to be $d^2n/d\lambda^2$=0.047 μm^{-2} and $d^3n/d\lambda^3$=-0.339 μm^{-3} respectively. The error in these measurements was estimated to be 10 %. Using these numbers, the calculated third order phase distortion $d^3\Phi/d\omega^3$ for one cavity round trip was found to be positive (~11,000 $fsec^3$) and not compensated. We have estimated [17] that the pulses have 10 fsec of cubic phase distortion and that with cubic dispersion minimization techniques [18] reduction of pulse widths by at least 20% is achievable.

In conclusion, we have demonstrated a regeneratively initiated self-mode-locked Cr:forsterite laser operated at 3.5 °C and pumped by a cw Nd:YAG laser at 1.064 μm. We have identified three regimes of operation for this laser. Without compensating for the cavity dispersion, 41 psec and 6.5 psec (FWHM) pulses with average TEM_{00} output powers of 280 and 380 mW respectively were produced at 1.23 μm. These modes of operation correspond to active mode-locking regimes, chirp-free and chirped, respectively. By employing intracavity GVD compensation, a very stable train of 48 fsec (FWHM) with average output power of 380 mW was generated. This regime is similar to the now common, self-mode-locked regime where soliton-like pulse shaping is important. Up to 2 minutes of self-sustained operation was observed. By tuning the mode-locked laser, second and third order crystal dispersion constants have also been measured for the first time. These represent, to our knowledge, the shortest, highest peak power light pulses directly generated from this laser system. These peak powers and operational stability open the door to applications such as second harmonic generation, and optical tomography of biological tissues.

Acknowledgments:

We would like to thank Timothy J. Carrig for helping with the experimental set-up and David Cohen with the data acquisition system. Thanks are also extended to Spectra Physics Lasers, Inc. for technical assistance. This work was supported by the National Science Foundation under grant ECS-9111838, the Joint Services Electronics Program, the Materials Science Center at Cornell University, and by the U.S. Department of Energy under the auspices of contract W-7405-Eng-48.

References:

1. D. E. Spence, P. N. Kean, and W. Sibbett, Opt. Lett. **16**, 42 (1991).

2. G. P. A. Malcolm and A. I. Ferguson, Opt. Lett. **16**, 1967 (1991).

3. A. Miller, P. LiKamWa, B. H. T. Chai, and E. W. Van Stryland, Opt. Lett. **17**, 195 (1992).

4. A. Seas, V. Petricevic, and R. R. Alfano, Opt. Lett. **17**, 937 (1992).

5. K. X. Liu, C. J. Flood, D. R. Walker, and H. M. van Driel, Opt. Lett. **17**, 1361 (1992).

6. P. LiKamWa, B. H. T. Chai, and A. Miller, Opt. Lett. **17**, 1438 (1992).

7. J. D. Kafka, M. L. Watts, and T. Baer, in *Digest of Conference on Lasers and Electro-optics* (Optical Society of America, Washington D.C., 1991), paper JMB3; J. D. Kafka, M. L. Watts, and T. Baer, in *Digest of Optical Society of America Annual Meeting*, (Optical Society of America, Washington D. C., 1991), paper TuI2.

8. J. D. Kafka, M. L. Watts, and J. J. Pieterse, IEEE J. Quantum Electron. **28**, 2151 (1992).

9. D. E. Spence, J. M. Evans, W. E. Sleat, and W. Sibbett, Opt. Lett. **16**, 1762 (1991).

10. F. Krausz, Ch. Spielmann, T. Brabec, E. Winter, and A. J. Schmidt, Opt. Lett. **17**, 204 (1992).

11. P. F. Curley and A. I. Ferguson, Opt. Lett. **16**, 1016 (1991).

12. A. Seas, V. Petricevic, and R. R. Alfano, Opt. Lett. **16**, 1668 (1991).

13. A. Sennaroglu, T. J. Carrig, and C. R. Pollock, Opt. Lett. **17**, 1216 (1992).

14. W. H. Knox, Opt. Lett. **17**, 514 (1992).

15. A. E. Siegman and D. J. Kuizenga, Opto-Electronics. **6**, 43 (1974).

16. A $sech^2$ pulse shape is expected when soliton-like pulse shaping is important.

17. R. L. Fork, C. H. B. Cruz, P. C. Becker, and C. V. Shank, Opt. Lett. **12**, 483(1987).

18. C. P. Huang, M. T. Asaki, S. Backus, M. M. Murnane, H. C. Kapteyn, and H. Nathel, Opt. Lett. **17**, 1289 (1992).

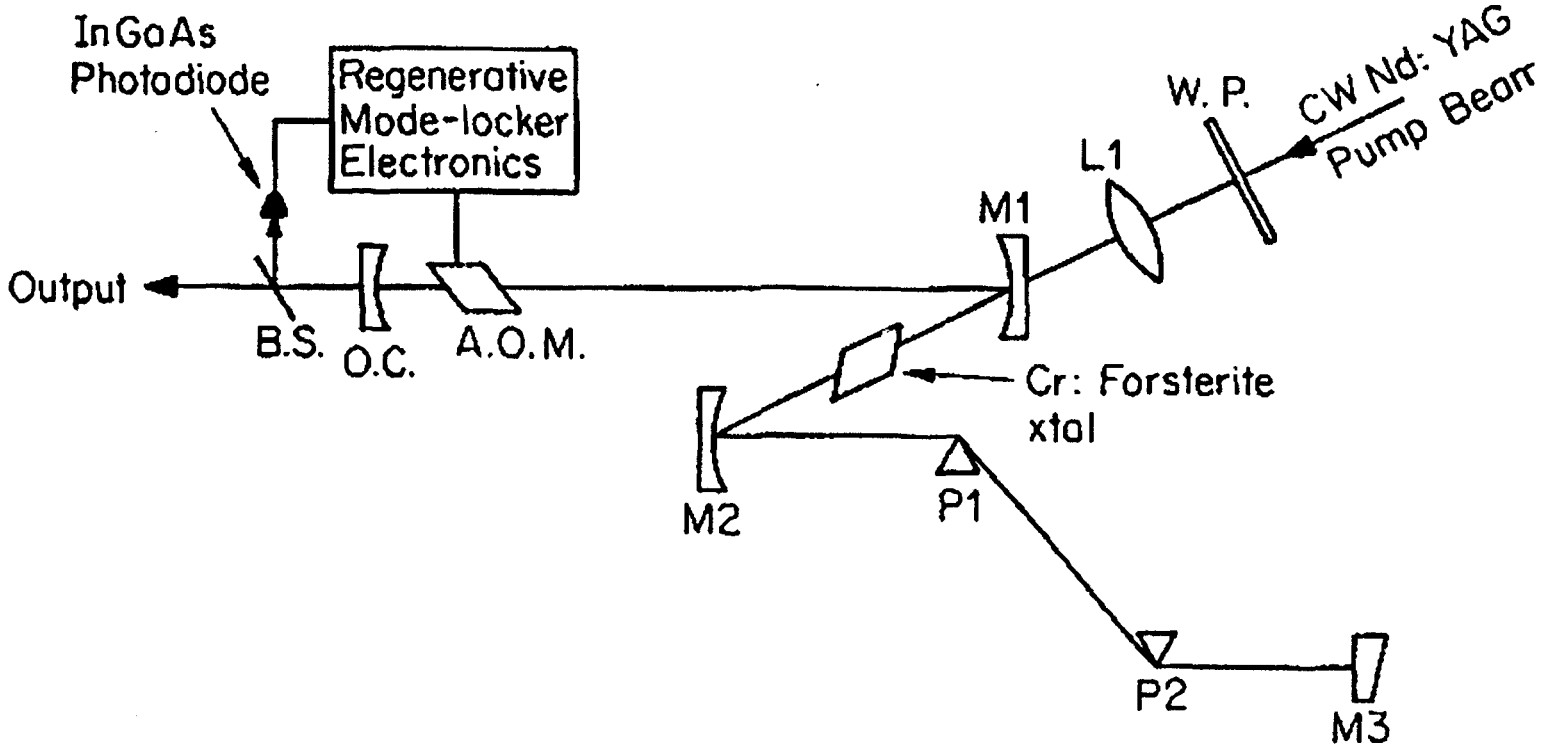
Figure Captions:

Figure 1:The schematic of the regeneratively initiated cw mode-locked Cr:forsterite laser.

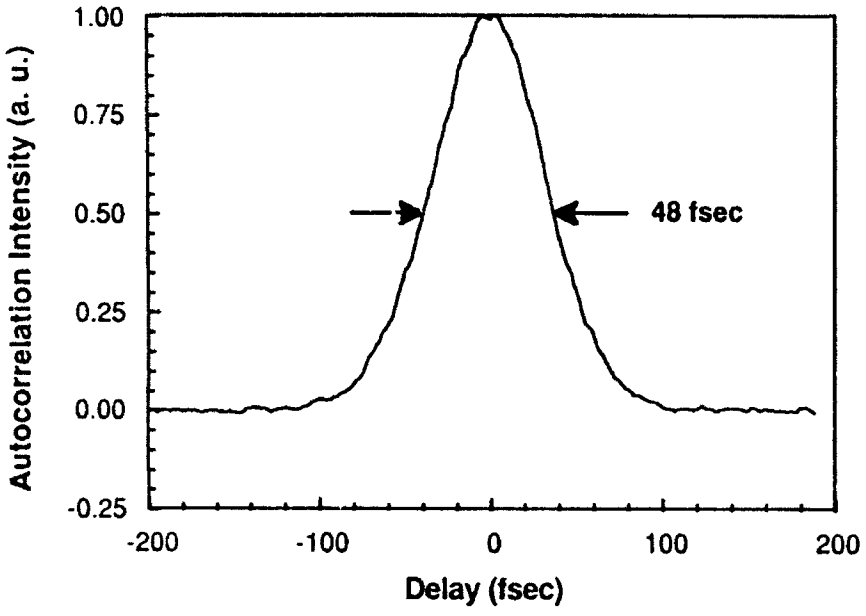
Figure 2:The noncollinear intensity autocorrelation of the regeneratively initiated cw mode-locked Cr:forsterite femtosecond pulses after dispersion compensation. The pulsewidth (FWHM) is 48 fsec.

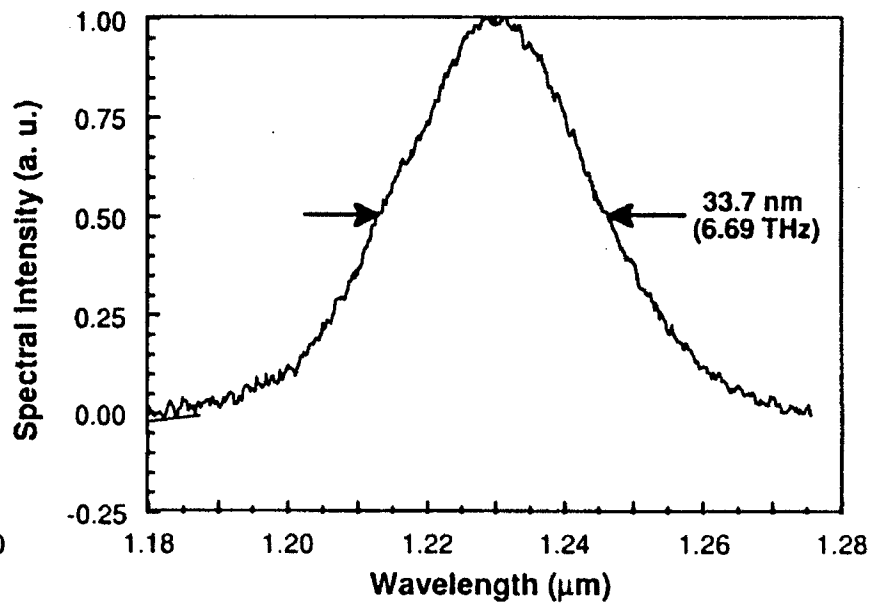
Figure 3:The spectrum of the regeneratively initiated cw mode-locked Cr:forsterite femtosecond pulses after dispersion compensation. The spectral width (FWHM) is 33.7 nm.

Modelocked Cr:forsterite oscillator



With dispersion compensation (SF-14 prisms) 48 fs near transform-limited pulses were generated





$\Delta\nu$ (FWHM) = 6.69 THz, τ (FWHM) = 48 fsec (sech^2)

$\Delta\nu\Delta\tau$ = 0.321 (sech^2)

λ = 1.23 μm

P_{abs} = 6.0 Watts

Output power = 380 mWatts

Peak output power per pulse = 97 kWatts

Self-sustaining

Generation of tunable femtosecond pulses in the red by frequency doubling a mode-locked Cr:forsterite laser

Alphan Sennaroglu and Clifford R. Pollock

School of Electrical Engineering, Cornell University, Ithaca NY 14853

Tel:(607)-255-5032; Fax:(607)-254-4565

Howard Nathel

Lawrence Livermore National Laboratories, Livermore CA 94551

Abstract

We report on the external second harmonic generation of a regeneratively-initiated self-mode-locked Cr:forsterite laser in $LiIO_3$ nonlinear crystal. Using 48 fsec pulses with average power of 246 mW at 1.23 μm, 75 fsec pulses with average power of 24 mW at 615 nm were obtained, giving conversion efficiencies approaching 10 %. The time-bandwidth product of the red pulses was measured to be 0.77. The second harmonic pulses were tunable from 605 nm to 635 nm, extending the operational wavelength range of the Cr:forsterite laser into the visible portion of the spectrum.

Generation of tunable femtosecond pulses in the red by frequency doubling a mode-locked Cr:forsterite laser

Alphan Sennaroglu and Clifford R. Pollock

School of Electrical Engineering, Cornell University, Ithaca NY 14853

Tel:(607)-255-5032; Fax:(607)-254-4565

Howard Nathel

Lawrence Livermore National Laboratory, Livermore CA 94551

External second harmonic generation (SHG) offers a simple scheme of extending the operational wavelength range of a tunable laser. Recently, there has been an unprecedented growth in the development of novel mode locking techniques using tunable solid-state lasers. Broadly tunable, high peak power subpicosecond pulses have been demonstrated over a large portion of the near IR region. Because such high peak powers are essential to achieving high conversion efficiencies in nonlinear processes such as SHG, these mode-locked tunable solid-state lasers open the way to efficient generation of tunable second harmonic pulses.

In this Letter, we describe the external doubling of a regeneratively-initiated, self-mode-locked Cr:forsterite laser using a $LiIO_3$ nonlinear crystal. Using 48 fsec(FWHM) input pulses at 1.23 μm with average output power of 246 mW, 75 fsec(FWHM) pulses at 615 nm with conversion efficiency of 10 % were obtained. By tuning the output of the pump laser from 1.21 to 1.27 μm, the second harmonic output wavelength could be tuned in the range 605 to 635 nm.

The regeneratively-initiated, self-mode-locked Cr:forsterite laser used in the SHG experiment has been described elsewhere[1]. Briefly, it consists of a folded, astigmatically compensated z-cavity with a 3.5 % transmitting output coupler. The gain medium is a 12 mm long Brewster-cut Cr:forsterite crystal having 0.3 % chromium concentration. The laser is collinearly pumped by a continuous-wave Nd:YAG laser operated at 1.06 μm. When maintained at an operating temperature of 3.5 ^{o}C through active cooling, the Cr:forsterite crystal absorbs 70.9 % of the incident 1.06 μm pump power. The absorbed pump power slope efficiency of the Cr:forsterite laser is 10.4%, the threshold pump power being 1.6 W. Compensating for the intracavity positive group velocity dispersion (GVD) by using a pair of Brewster-cut SF-14 prisms seperated by 48 cm, self mode locking is initiated with a regeneratively driven acousto-optic mode-locker operated off-resonance. The mode-locked Cr:forsterite laser, operating at a 81.27 MHz pulse repetition rate, is capable of delivering average powers as high as 380 mW. The output pulsewidth (FWHM) is 48 fsec at 1.23 μm. By translating a slit between the the second prism of the GVD compensation pair and the cavity high reflector, the output wavelength of this laser can be tuned in the wavelength region from 1.21 to 1.27 μm.

The SHG set-up used for externally doubling the mode-locked Cr:forsterite laser is shown in figure 1. As the nonlinear medium, a 2 mm thick $LiIO_3$ crystal(Cleveland Crystals, Inc.), type-I phase-matched at 1.23 μm was used. In order to prevent degradation of the surface quality of this hydroscopic crystal, a cover slip(of thickness 0.2 mm) with anti-reflection (AR) coating on one side was glued to each crystal surface using uv curing epoxy. The nonlinear crystal was mounted on a rotation-tilt stage to accurately optimize

the second harmonic efficiency while tuning the pump laser. The incident Cr:forsterite laser beam was focussed to a 25 μm diameter spot inside the $LiIO_3$ crystal using a telescope arrangement of two 5 cm focal length AR coated plano-convex lenses (L1 and L2) seperated by 1.5 cm. The emerging beam was recollimated with a broad band AR coated (450-700 nm) 5 cm focal length lens(L3). After separating the second harmonic signal from the fundamental with a dichroic filter (F1) having 99.9% reflectivity at 1.23 μm and 95% transmission in the red, the SHG power was measured with a Molectron model 5100 power meter. Temporal characteristics of the red pulses were studied by measuring the collinear intensity autocorrelation with a 0.6 mm thick BaB_2O_4 (BBO) crystal aligned for type I phase matching. The spectral width of the SHG pulses was measured with a 0.25 m monochromator and a silicon detector.

After careful alignment of the $LiIO_3$ crystal, 24 mW of average power at 615 nm was obtained with 246 mW of incident power at 1.23 μm, resulting in 9.7% conversion efficiency. Figure 2 shows the collinear intensity autocorrelation of the SHG pulses at 615 nm. Assuming a $sech^2$ intensity profile, the pulsewidth(FWHM) was measured to be 75 fsec. A simultaneous measurement of 13 nm spectral bandwidth gave a time-bandwidth product of 0.77. The red pulses could be tuned from 605 to 635 nm with the pulsewidth essentially remaining the same.

The expected efficiency of second harmonic generation from $LiIO_3$ was estimated by taking into account the walk-off effects between the fundamental and the second harmonic beams, the finite divergence of the fundamental beam and the finite spectral phase matching bandwidth of the crystal. Following the treatment of Boyd and Kleinman [2], the amount of second harmonic power $P_{2\omega}$ (in watts) generated from a monochromatic beam in a

nonlinear medium with the assumption of no absorption and pump depletion can be estimated using the equation

$$P_{2\omega} = \frac{16\pi^2\eta_0 d^2 L h_m(B,\xi)}{n^3} \frac{P^2(\lambda)F(\lambda)}{\lambda^3} \qquad (1)$$

In (1), where all the quantities are expressed in MKS units, η_0 is the vacuum impedance, d is the effective nonlinear coefficient of the medium, L is the crystal length, n is the crystal index of refraction and $P(\lambda)$ is the fundamental spectral power distribution. The dimensionless factor $h_m(B,\xi)$, which is a function of the normalized walk-off parameter B and the normalized focussing parameter ξ, accounts for the efficiency limitations due to walk-off effects arising from double refraction and the finite beam divergence (see reference 2 for definitions of B and ξ). One realizes that the fundamental beam in this experiment is no longer monochromatic for 48 fsec pulses and the effect of the finite spectral phase-matching bandwidth of the $LiIO_3$ crystal has to be taken into account through the efficiency factor $F(\lambda)$ appearing in (1) defined according to

$$F(\lambda) = \mathrm{sinc}^2\left[\frac{\Delta k(\lambda)L}{2}\right] \qquad (2)$$

In (2), $\Delta k(\lambda)$ is the wave vector mismatch between the fundamental and the second harmonic waves. By expressing $P^2(\lambda)$ as

$$P^2(\lambda) = P_0^2\,\rho_n(\lambda) \qquad (3)$$

where $\rho_n(\lambda)$ is the normalized spectral distribution function of the squared incident power, the effect of the finite spectral phase-matching bandwidth of the SHG crystal on the conversion efficiency can be estimated by integrating (1) over all wavelengths. This simply replaces the function $P^2(\lambda)F(\lambda)/\lambda^3$ appearing in (1) by the spectrally averaged value of $P^2(\lambda)/\lambda^3$ using $F(\lambda)$ as the

weighting factor.

Calculation of various quantities appearing in (1) was done using the Sellmeier equations for $LiIO_3$ given in reference 3. Using the fact that $LiIO_3$ is a negative uniaxial crystal, the type I phase matching angle θ_m and the walk-off angle ρ are calculated to be 25.94° and 3.67° respectively. Furthermore, since $LiIO_3$ belongs to the point group 6, the effective nonlinear coefficient d given by $d_{31}\sin(\theta_m+\rho)$ is calculated to be 2 pm/V using d_{31}=4.1 pm/V [4]. For a 2 mm thick crystal with index of refraction n=1.85218 and focussed beam diameter of 25 μm the parameters B and ξ discussed earlier evaluate to 4.4 and 1.35 giving $h_m(B,\xi)$~0.17 [2].

By using the Sellmeier equations and a fixed phase-matching angle of 25.94°, $F(\lambda)$ defined in (2) is plotted in figure 3 for a 2 mm long $LiIO_3$ crystal. Also plotted in figure 3 is the function $\rho(\lambda)$ (not normalized) for a sech^2 pulse of duration 48 fsec (FWHM). By averaging $\rho_n(\lambda)$ using $F(\lambda)$ as the weighting factor, we estimated that the finite spectral phase-matching bandwidth of the crystal would cause approximately 75 % reduction in the SHG conversion efficiency. With this consideration in mind and by substituting all the relevant parameters calculated above into (1), we came up with an expected conversion efficiency of approximately 11 % for 63 kW peak power pulses. This is in excellent agreement with the experimentally obtained value of 10%.

The finite phase matching bandwidth of the crystal is also expected to affect the temporal and spectral characteristics of the second harmonic pulses. One would ideally expect the second harmonic pulsewidth to be 0.707 times that of the fundamental pulses. However, as seen in figure 3, the limited phase matching bandwidth of the 2 mm thick $LiIO_3$ crystal will reduce the bandwidth available for doubling by at least 50 % resulting in second

harmonic pulses of about 70 fsec. In addition, GVD of $LiIO_3$ ($d^2n/d\lambda^2$=0.6866 μm^{-2} at 615 nm) together with the GVD of approximately 2.2 cm of fused silica glass between the SHG crystal and the autocorrelator is expected to further broaden these pulses to approximately 85 fsec. This is in good agreement with the 75 fsec (FWHM) pulses measured in our experiment. The measured time-bandwidth product of 0.77 also verifies that broadening and possible spectral distortion was experienced by the SHG pulses. Our estimations of the expected pulsewidth from the SHG process are only approximate. More accurate numerical analysis would be required to fully study the combined effects of finite phase-matching bandwidth of the nonlinear crystal and dispersive effects on the pulsewidth and time-bandwidth product.

In conclusion, we have demonstrated efficient external doubling of 48 fsec pulses from a mode-locked Cr:forsterite laser using $LiIO_3$ nonlinear crystal. With 246 mW of incident power at 1.23 μm, 75 fsec (FWHM) pulses with conversion efficiency of 10 % were obtained at 615 nm. The experimentally measured SHG conversion efficiency agreed well with the expected value which was calculated by taking into account the beam walk-off effects, finite beam divergence of the fundamental beam and the limited spectral phase-matching bandwidth of the $LiIO_3$ crystal. The red pulses which were tunable in the wavelength region from 605 nm to 635 nm now extend the operational wavelength range of the Cr:forsterite laser into the visible portion of the spectrum. With the available high peak powers from this laser system it should be possible to use more sophisticated nonlinear parametric amplification schemes to obtain broader wavelength tunability.

Acknowledgments

We thank Randy Ellingston, David Cohen, and Peter Powers for helpful discussions. This work was supported by the National Science Foundation under grant ECS-9111838, the Joint Services Electronics Program, the Materials Science Center at Cornell University, and by the U.S. Department of Energy under the auspices of contract W-7405-Eng-48.

References

1. A. Sennaroglu, C. R. Pollock, and H. Nathel, submitted for publication to Optics Letters, Jan 1993.

2. G. D. Boyd and D. A. Kleinman, J. Appl. Phys. **39** 3597 (1968).

3. M. M. Choy, R. L. Byer, Phys. Rev. B **14** 1693 (1976).

4. R. C. Eckardt, H. Masuda, Y. X. Fan, and R. L. Byer, IEEE J. Quantum Electron. **26** 922 (1990).

Figure Captions:

Figure 1: The experimental set-up of the externally doubled Cr:forsterite laser.

Figure 2: The collinear autocorrelation of the SHG pulses at 615 nm. The pulsewidth(FWHM) is 75 fsec.

Figure 3: The plot of $F(\lambda)$ and $\rho(\lambda)$ as a function of wavelength (μm) for a 2 mm thick $LiIO_3$ crystal phase-matched at 1.23 μm.

Fig 1

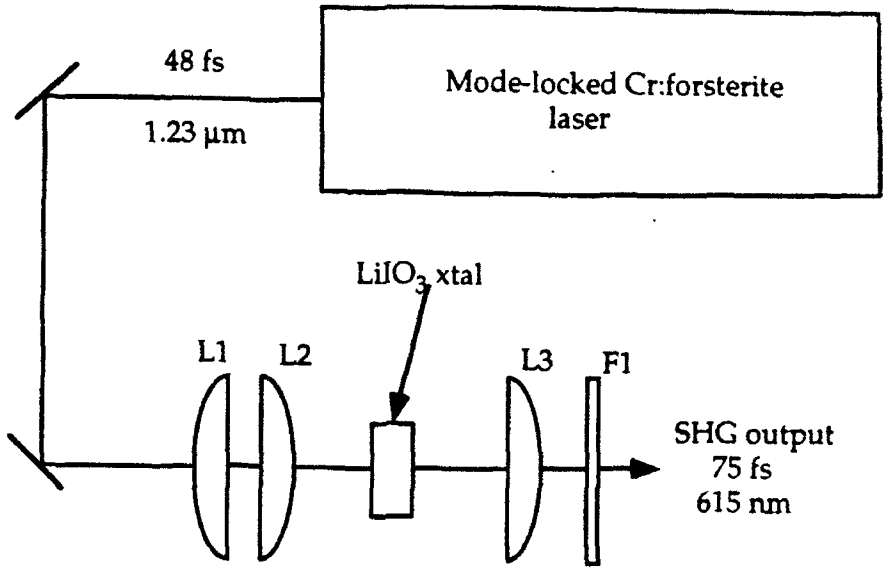

48 fs
1.23 μm
Mode-locked Cr:forsterite laser
$LiIO_3$ xtal
L1
L2
L3
F1
SHG output
75 fs
615 nm

Fig 2

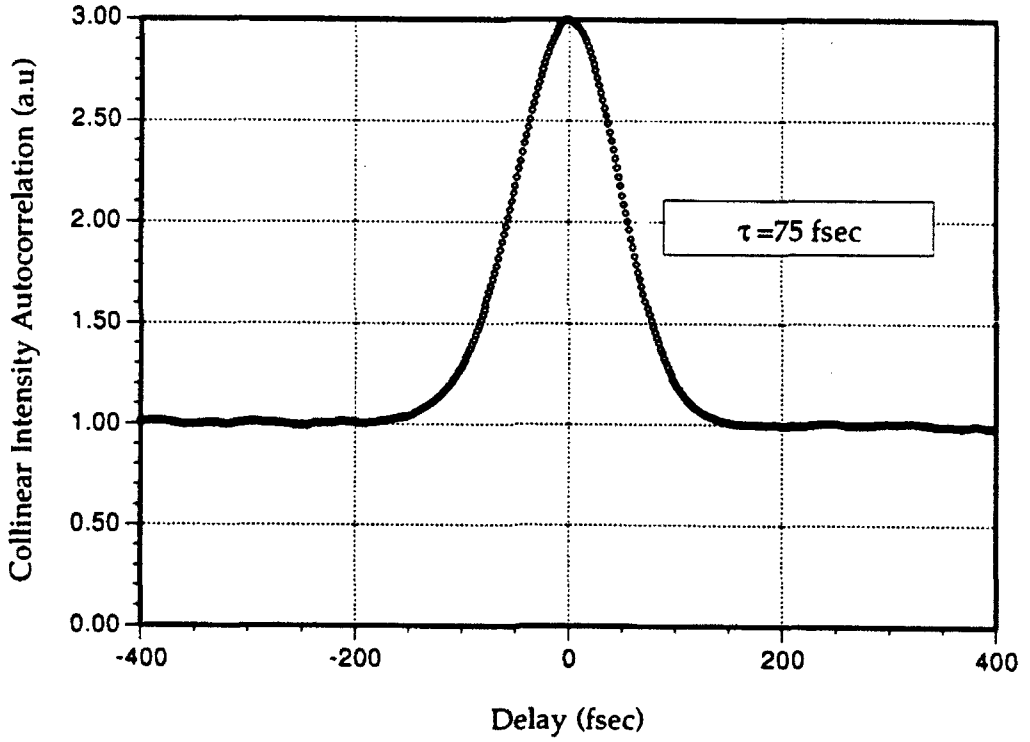
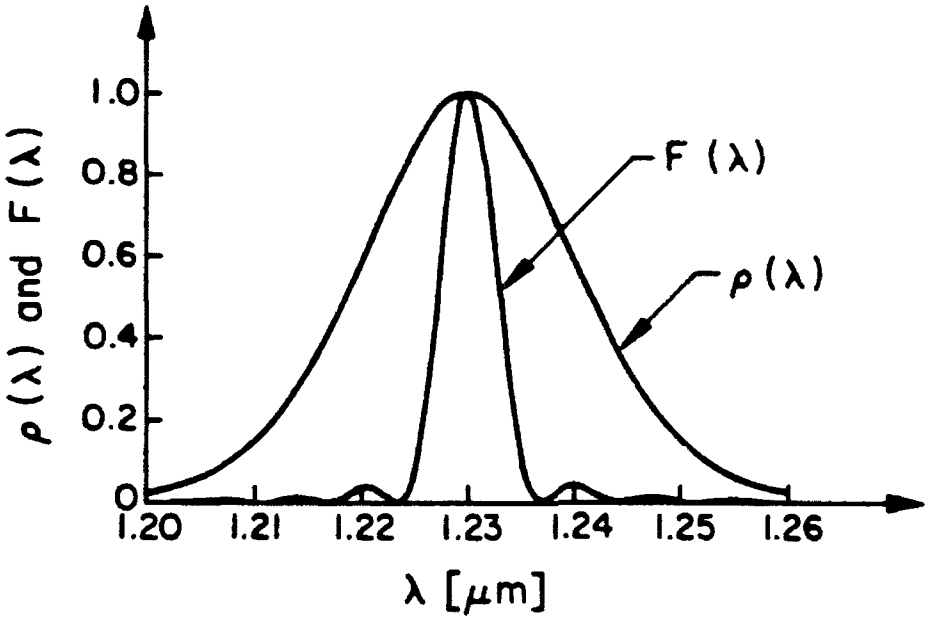

Collinear Intensity Autocorrelation (a.u)
3.00
2.50
2.00
1.50
1.00
0.50
0.00
-400
-200
0
200
400
Delay (fsec)
τ=75 fsec

Fig 3.



TASK 4 FEMTOSECOND DUAL CARRIER TRANSPORT AND OPTICAL INTERACTIONS IN COMPOUND SEMICONDUCTOR HETEROSTRUCTURES

J. P. Krusius

Heterojunction Vertical FET's Revisited: Potential for 225-GHz Large-Current Operation

Steven R. Weinzierl and J. Peter Krusius, *Senior Member, IEEE*

***Abstract*—High-speed operation of submicrometer $Al_xGa_{1-x}As$/GaAs unipolar heterojunction transistors is examined using two-dimensional time-dependent self-consistent ensemble Monte Carlo simulation. Careful device design can significantly increase ballistic injection over the heterojunction in steady state by eliminating retarding gate-induced space-charge reversal there. Design for optimal large-signal transient operation must also avoid gate-voltage-dependent ballistic injection. General design principles for optimizing high-speed operation are proposed. The resulting VFET's show cutoff frequencies of 225 GHz at large drain currents at 300 K, with frequency-independent two-port *y* parameters.**

I. Introduction

BANDGAP engineered unipolar heterojunction transistors have long held great promise for ultra-high-speed operation [1]. Although today's lateral heterostructure devices are well developed, unipolar heterostructure devices with transport across the heterolayers (vertical FET, VFET) have not lived up to their expected performance. Early preliminary Monte Carlo simulations predicted idealized intrinsic transconductances of 1250 mS/mm and unity gain cutoff frequencies of 250 GHz at 77 K [2], while fabricated devices have never surpassed transconductances of 100 mS/mm [3], [4]. Three reasons have motivated this study of heterostructure VFET devices: 1) to explain the wide performance gap between predicted and measured characteristics of VFET-type devices, 2) to establish guidelines for the optimum VFET device designs, and 3) to study carrier launching across a heterojunction (HJ) in the presence of lateral space charges for the first time using a realistic nonequilibrium carrier transport formulation. While specifically focusing on the HJ-VFET the principles found in this study are applicable to a number of other devices, including the vertical MESFET, the permeable base transistor (PBT), and VFET's with a planar doped barrier launcher.

II. Simulation Method

A two-dimensional self-consistent time-dependent ensemble Monte Carlo particle formulation is used here to explore the nonequilibrium transport processes described above. This method is a straightforward extension [5] of our equivalent one-dimensional formulation [6]. The full knowledge of the microscopic processes provided by the Monte Carlo method allows computation of all figures of merit for the intrinsic device; no extrinsic device parasitics are considered here. Both transconductance (g_m) and gate capacitance (C_G) are determined from their definitions using simulated steady-state terminal current or integrated charge data. The unity-gain cutoff frequency (f_t) is then computed via $f_t = g_m/(2\pi C_G)$. The complex frequency-dependent small-signal *y* parameters are determined directly from the Monte Carlo result via the Fourier decomposition method [7], [8], i.e., $y_{ij}(\omega) = F[\Delta I_i(t)]/F[\Delta V_j(t)]$, where F denotes the Fourier transform, $\Delta I_i(t)$ the current change at port i in response to the voltage change $\Delta V_j(t)$ at port j.

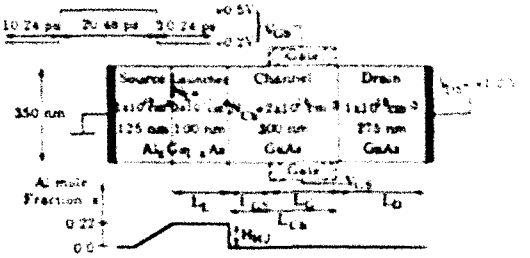


Fig. 1. Cross section of heterojunction VFET with parameters for baseline device (1 in Table I). Inserts show the $Al_xGa_{1-x}As$ grading profile and the gate pulse for transient analysis.

III. Definition of Device Structures

All VFET devices examined have the same structure derived from fabricated devices, in which current flows in parallel fingers from the top electrode (source) down through the channel into the bottom electrode (drain). The channel current is modulated by lateral gate electrodes placed symmetrically on both sides. Only one of these fingers needs to be simulated and its cross section is shown in Fig. 1. Source and drain contacts are assumed ohmic, while gates are Schottky contacts. The heterostructure launcher is embedded into the source and has a graded $Al_xGa_{1-x}As$ ramp and an abrupt heterojunction toward the channel. Table I shows the parameter sets for three different devices: a fabricated device [4], a baseline device (starting point for optimization) similar to the fabricated one, and the fully optimized device designed for 300 K

Manuscript received March 29, 1991; revised August 26, 1991. This work has been supported by the Joint Services Electronics Program under Contract F49620-90-C-0039, monitored by AFOSR (Dr. G. Witt). The review of this paper was arranged by Associate Editor S. E. Laux.

The authors are with the School of Electrical Engineering, Cornell University, Ithaca, NY 14853.

IEEE Log Number 9106896.

TABLE I
LAYER SEQUENCES FOR FABRICATED, BASELINE, AND OPTIMIZED HJ VFET DEVICES

Layer Name	Material	Parameter	Fig. 1 Symbol	Fabricated	Baseline	Optimized
Source	n^+ GaAs	length, nm		50	50	50
		doping, cm^{-3}		4×10^{17}	1×10^{17}	1×10^{17}
Grading	n^+-$Al_xGa_{1-x}As$	length, nm		75	75	50
		doping, cm^{-3}		4×10^{17}	1×10^{17}	1×10^{17}
Launcher	n-$Al_xGa_{1-x}As$	length, nm	L_L	90	100	50
		doping, cm^{-3}	N_L	3×10^{17}	3×10^{17}	1×10^{17}
		Al mole fraction	H_{Al}	22%	22%	12%
Spacer	i-$Al_xGa_{1-x}As$	length		10 nm	not included	
Channel	GaAs	length, nm	L_{Ch}	500	300	200
		doping, cm^{-3}	N_{Ch}	2×10^{15}	2×10^{15}	2×10^{15}
Drain	GaAs	length, nm	L_D	1200	300	150
		doping, cm^{-3}		4×10^{18}	1×10^{18}	1×10^{18}
Additional parameters		gate length, nm	L_g	200	200	130
		gate-source spacing, nm	L_{GS}	+100	+100	+35
		device width, nm		350	350	350

operation using the general guidelines given in Section VII. The baseline device has a 350-nm lateral width, and two symmetric 200-nm-long gate electrodes placed 100 nm downstream from the HJ. Source and drain doping in the baseline device is smaller than in the fabricated devices in order to avoid degeneracy and carrier-carrier scattering in these regions. The thin undoped spacer region at the heterojunction in the fabricated device was dropped as it is likely to be washed out during materials growth. Channel and drain lengths are shorter than in the fabricated device, as the fabricated channel length of 500 nm far exceeds the quasi-ballistic mean free path even at 77 K and the long drain length increases the series resistance. Fourteen design variations, covering all significant characteristics, have been defined in Table II. The optimization occurs in two steps. First, the operation of the fabricated device is analyzed. Next a new more suitable baseline device is defined for optimization. Finally single parameter variations are performed successively until the optimum is reached. A fully statistical response study is not necessary because of the microscopic insight provided by the Monte Carlo method.

IV. Correlation with Measured Data

The accuracy of the method was verified by simulating a two-dimensional cross section of the fabricated device and then comparing simulated steady-state current–voltage (I-V) characteristics with that measured in the fabricated device [4], whose layer sequence is given in Table I. It has 10 parallel fingers which are each 132 μm long and 350 nm wide, with 200-nm-long gates. The gate-to-source spacing is 100 nm. The simulated steady-state drain current differed from measured data at 300 K by less than 15% (maximum global error), a result obtained without any adjustable parameters.

V. Steady-State Operation

The key to understanding steady-state operation of this class of devices is the dipole layer at the heterojunction. It was recently shown that two-dimensional macroscopic current continuity in conjunction with the lateral space charge induced by the gate electrodes controls the electron injection conditions over the heterojunction [9]. Specifically, a dipole moment forms at the heterojunction. Its magnitude and direction is dependent on the externally applied gate voltage. Usually, the dipole moment is directed so as to retard ballistic injection, which then becomes gate-voltage dependent. The baseline VFET design (device 1) demonstrates this effect very distinctly. Its electron density, average electron drift velocity, and self-consistent conduction band edge in the center of the channel along the direction of carrier flow are given in Figs. 2–4. This effect limits the performance of the baseline device to $g_m = 312$ mS/mm and $f_t = 64$ GHz at a drain current density of $I_D = 5 \times 10^4$ A/cm^2. This constitutes a negligible improvement over the GaAs device with no embedded heterojunction (device 2).

Channel-limited transport in an FET is forced by the applied gate-to-source voltage V_{GS} via the depletion regions at a location in the channel where carrier densities are low and where carrier velocities reach approximately the saturation velocity. Therefore, three different methods for controlling channel-limited transport were investigated by adjusting device parameters from their baseline

TABLE II
DEFINITION OF DEVICE PARAMETER CHANGES FOR OPTIMIZATION

Device No.	Parameter(s) Changed	Fig. 1 Symbol	Modified Value From Current Baseline
1	**Baseline**	baseline	baseline
2	launcher height	H_{HJ}	$x = 0\%$
3	channel length	L_{Ch}	220 nm
	gate-source spacing	L_{GS}	+10 nm†
4	gate-source spacing	L_{GS}	−100 nm
5	channel doping	N_{Ch}	7×10^{16} cm^{-3}
6	**New Baseline**		
	channel length	L_{Ch}	200 nm
	gate length	L_G	150 nm
	gate-source spacing	L_{GS}	+25 nm
	channel doping	N_{Ch}	7×10^{16} cm^{-3}
	launcher height	H_{HJ}	133 meV, $x = 22\%$
	launcher length	L_l	100 nm
	launcher doping	N_l	3×10^{17} cm^{-3}
	drain length	L_D	275 nm
7	launcher height	H_{HJ}	266 meV, $x = 44\%$
8	channel doping	N_{Ch}	2×10^{17} cm^{-3}
9	launcher height	H_{HJ}	67 meV, $x = 11\%$
10	launcher length	L_l	0 nm
11	launcher doping	N_l	1×10^{18} cm^{-3}
12	drain length	L_D	150 nm
13	gate length	L_G	75 nm
14	gate length	L_G	75 nm
	gate-source spacing	L_{GS}	+50 nm
15	**Optimized**		
	channel length	L_{Ch}	200 nm
	gate length	L_G	130 nm
	gate-source spacing	L_{GS}	+35 nm
	channel doping	N_{Ch}	7×10^{16} cm^{-3}
	launcher height	H_{HJ}	73 meV, $x = 12\%$
	launcher length	L_l	50 nm
	launcher doping	N_l	1×10^{18} cm^{-3}
	drain length	L_D	150 nm

†Changed only to maintain gate length L_G of 200 nm.

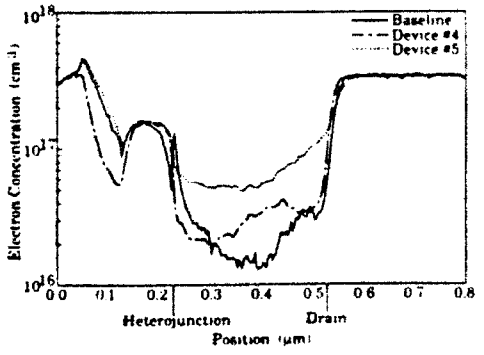


Fig. 2. Steady-state electron concentration along the center of the channel in the direction of current flow for devices 1, 4, and 5 at 300 K. See Table II for device parameters. $V_{GS} = +0.2$ V and $V_{DS} = +1.0$ V.

values: a) reduction of channel length to 220 nm (device 3), b) gate electrode placement symmetrically around the HJ ($L_{GS} = -100$ nm, device 4), and c) enhanced channel doping ($N_{Ch} = 7 \times 10^{16}$ cm^{-3}, device 5). In each of these

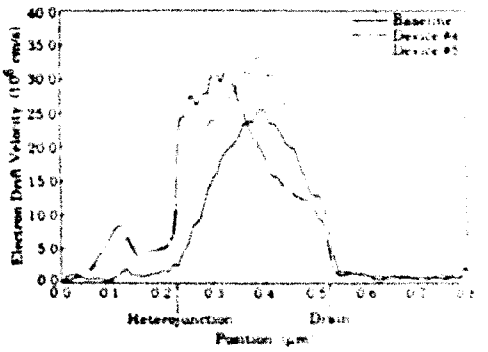


Fig. 3. Steady-state average electron drift velocity along the center of the channel in the direction of current flow for devices 1, 4, and 5 at 300 K. $V_{GS} = +0.2$ V and $V_{DS} = +1.0$ V.

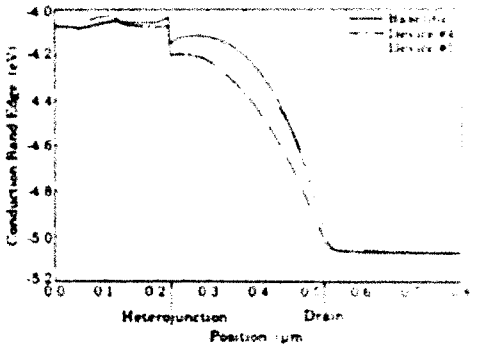


Fig. 4. Steady-state self-consistent Γ valley band edge along the center of the channel in the direction of current flow for devices 1, 4, and 5 at 300 K. See Table II for device parameters. $V_{GS} = +0.2$ V and $V_{DS} = +1.0$ V.

cases the reversal of the sign of the dipole layer moment at the HJ will be prevented, which is reflected in an enhanced g_m (Fig. 5). Because the gate capacitance C_G is also affected, the cutoff frequency f_t may or may not improve (Fig. 5). The device with the gate overlapping the source (device 4) exhibits a substantially reduced f_t due to increased C_G, and performs worse than the device with no launcher (device 2). Devices 3 and 5 both showed improved performance, with g_m's of 333 and 418 mS/mm, respectively, and f_t's improved 30% and 17% over the baseline device. Although both exhibit the desired flat-band condition at the HJ even in saturation, device 3 with the short channel still suffers from channel-limited transport due to insufficient channel doping, and device 5 with the enhanced channel doping still has a channel longer than the quasi-ballistic mean free path. Thus the best method for preventing space-charge reversal at the heterojunction is to both decrease the channel length and increase the channel doping (new baseline device, device 6, $g_m = 438$ mS/mm, $f_t = 81$ GHz). This device is taken as the new baseline device.

The structure of the HJ launcher itself is obviously the other important factor controlling steady-state operation. One expects that a large conduction band offset at the HJ launcher results in enhanced immediate electron transfer into the heavier mass L valleys. This mechanism will

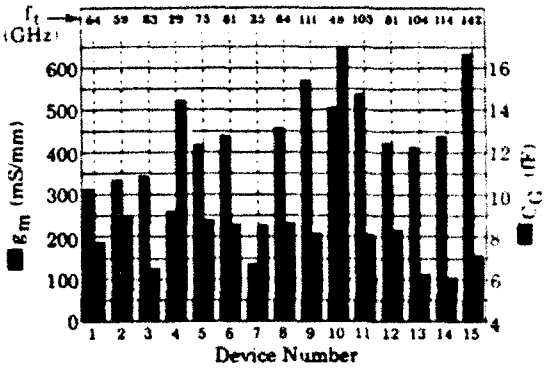


Fig. 5. Transconductance g_m, gate capacitance C_{GS}, and cutoff frequency f_t for devices 1-15 at 300 K. See Table II for device parameters. V_{GS} was stepped from +0.2 to +0.5 V while V_{DS} was held at +1.0 V.

make quasi-ballistic channel transport impossible because transfer from the Γ to L valley occurs between orthogonal quantum states, randomizing all components of the momentum wave vector. In addition, quantum-mechanical reflection at the HJ is increased. Simulation results for device 7 with a doubled band offset (H_{HJ} = 266 meV) confirm this expectation with an f_t of less than half that for the device with no launcher (device 2).

Although ballistic injection occurs in device 8 with an enhanced channel doping ($N_{Ch} = 2 \times 10^{17}$ cm^{-3}), ballistic transport in the channel is prevented by the dominant ionized impurity scattering mechanism, which has no strong preference for small-angle scattering for electron energies in the channel; a widened and drifted carrier distribution function downstream from the HJ is produced. Current continuity together with the large channel doping actually forces launcher-limited transport in this device, as demonstrated by a pulled-down band edge. This results in no improvement over the new baseline, device 6. If the constant mole fraction section in the HJ launcher of the new baseline device 6 is left out with everything else being constant, the resulting device 10 suffers from the largest gate capacitance. In this case the conditions for thermionic emission are no longer satisfied, and Γ to L valley transfer in the channel is increased. This results in no improvement over the new baseline, device 6. Decreasing the drain length by nearly half to 150 nm (device 12), also gives no improvement over the new baseline, device 6.

Substantial improvement is realized by decreasing the launcher height (H_{HJ} = 66 meV, device 9), increasing launcher doping ($N_L = 1 \times 10^{18}$ cm^{-3}, device 11), and decreasing the gate length (L_G = 75 nm, device 13). Device 9 with the shallower launcher gave a notable improvement over the baseline device 6, primarily because less Γ to L valley transfer downstream from the heterojunction enhances the transconductance, while still providing sufficient ballistic injection at the HJ. Increasing launcher doping prevents launcher-limited transport as evident from device 6. Decreasing the gate length primarily reduces the gate capacitance, while still providing a channel pinch-off capability. Moving this short gate farther downstream (L_G = 100 nm, L_{GS} = +50 nm, device 14), shows nearly no difference, indicating that L_{GS} = +25 nm is sufficient to prevent gate-source interaction.

VI. Transient Operation

The large signal switching characteristics are quantified here via the response of the device to a voltage pulse applied to the gate terminals while keeping the drain voltage fixed during the transient. A gate step voltage of ΔV_{GS} = +0.3 V (less depletion) with zero rise time for a period of 20.48 ps was used with the drain biased into saturation (V_{DS} = +1.0 V). This corresponds to an increase in the drain current of 74% for the baseline device. At this bias point g_m has half its maximum value. From the steady-state operation principles discussed above, one expects that devices with a voltage-dependent, and hence current-dependent, dipole layer at the HJ (with polarity reversal) will have poor switching characteristics with a long period of damped charge and terminal current oscillations. This is confirmed by our simulations. The oscillations are driven by the following two mechanisms. First, the large-signal transient settling time is largely determined by the current density, which is substantially reduced during the transient due to the current-dependent ballistic injection. Second, the nonlinear injection process at the HJ, and the linear injection processes at the ohmic source and drain contacts, are coupled. This coupling occurs on a time scale on the order of the dielectric relaxation time $\tau_{D\#}$, which is about 15 fs in the heavily doped source/drain regions. Contrary to this, the overall current density through the device is at best modulated on a time scale related to the plasma frequency ω_p (about 100 fs), and at worst on the time scale of the source-heterojunction transit time $\tau_{S\,HJ}$, which is about 1 ps for this device size. Combined with the current-dependent injection, the presence of these two different natural time scales leads to an out-of-phase heterojunction-to-ohmic contact feedback, which drives the current oscillations during the transient (Fig. 6). Damping is provided by the scattering mechanisms. This is confirmed by the fact that the period of oscillations in the drain current transient in Fig. 6, about 320 fs, exactly matches the time dependence of the ballistic fraction at the HJ (Fig. 7). The presence of the two coupled processes is manifested in the strong frequency dependence of the transconductance g_m (real part of y_{21} in Fig. 8). Also seen in that figure is the excessive gate capacitance of some of the devices arising from the gate-source interaction. For example, device 4 has a positive susceptance y_{21}. All devices showed a similar gate self-admittance y_{11}: the susceptance was capacitive and the conductance small because the Schottky gates allowed only displacement current to flow.

VII. Device Design Criteria

Design criteria have been derived from the steady-state and transient operation principles discussed above. The key to balanced high-speed and high-current operation is held by the dipole layer at the HJ: it should not be a re-

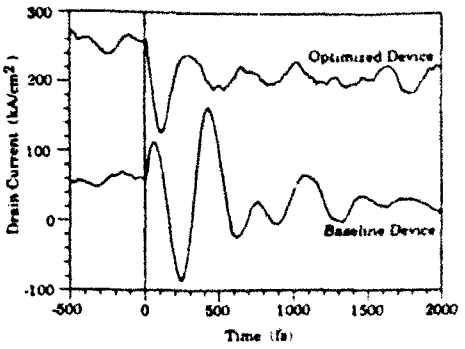


Fig. 6. Transient drain current I_D as a function of time for baseline and optimized devices 1 and 15. The applied gate step voltage was 0.3 V into less channel depletion: V_{GS} decreased from +0.5 to +0.2 V, and V_{DS} = +1.0 V.

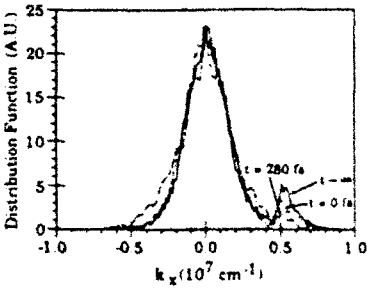


Fig. 7. Distribution function for Γ-valley electrons 10 nm downstream from the HJ launcher in the center of the channel for the baseline device 1 at 300 K, given at three specified times after the application of the gate step voltage. V_{GS} increased from +0.2 to +0.5 V, and V_{DS} = 1.0 V.

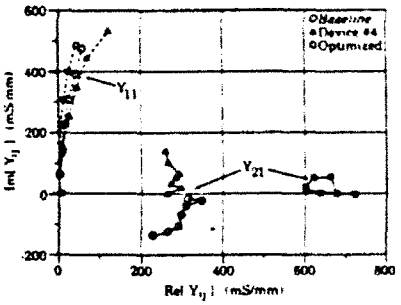


Fig. 8. Small-signal y parameters y_{11} and y_{21} as a function of frequency for devices 1, 4, and 15. Device parameters are defined in Table II. Simulated results are given in 24.4-GHz increments between 0 and 147 GHz. V_{GS} was increased from +0.2 to +0.5 V for 20.48 ps while V_{DS} = +1.0 V.

tarding one for the desired operating conditions and it's bias dependence should be as small as possible. This can be accomplished by following the guidelines below:

1) The channel length L_{Ch} should be comparable to the quasi-ballistic mean free path in the channel (~200 nm at 300 K lattice temperature for GaAs). Then the formation of a voltage-dependent retarding dipole layer at the heterojunction is prevented.

2) The gate length L_G should be reduced until the edge and area gate capacitance contributions become comparable. The gate edge should not be close to the heavily doped source, or drain, areas to minimize capacitive feedback. 130-nm gate lengths with +35-nm gate-to-source

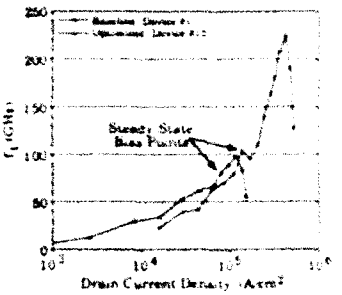


Fig. 9. Unity gain cutoff frequency f_t as a function of drain current density for baseline and optimized devices 1 and 15. V_{DS} = 1.0 V for both devices, −1.0 V < V_{GS} < +1.1 V for the baseline device, and −2.8 V < V_{GS} < +1.1 V for the optimized device. A gate Schottky barrier height of 0.8 V is assumed.

spacings for channel lengths of 200 nm can be achieved [10].

3) The channel doping density N_{Ch} is set by the tradeoff between competing demands: its reduction is required to minimize ionized impurity scattering and its enhancement is required to prevent channel-limited transport.

4) The launcher height H_{HJ} should be set by trading off the kinetic energy increase at the HJ to electron transfer into upper conduction band valleys. It has already previously been shown by Tang and Hess [11] that a 70-meV launcher height, which corresponds to x = 0.11, will result in average overshoot velocities as large as 5 × 10^7 cm/s for 300 K.

5) The length of the launcher L_L must be long enough to allow for a symmetric quasi-equilibrium momentum distribution immediately upstream from the HJ, to satisfy conditions for thermionic emission.

6) The launcher doping N_L must be large enough to support overshoot velocities in the channel in order to prevent launcher-limited transport. Doping it the same as the rest of the source is acceptable for GaAs.

7) The length of the heavily doped drain region should be minimized to keep the transit time short. About 150 nm is needed in order to thermalize hot carriers from the L valley into the Γ valley before reaching the ohmic drain contact.

8) The length of the graded launcher region should not be reduced to below 50 nm to avoid quantum-mechanical reflection. The same requirement justifies the use of the semi-classical Monte Carlo transport formulation [5].

VIII. Optimized Device Characteristics

The optimized device 15 simultaneously displays excellent steady-state and transient characteristics for 300 K operation. It reaches the computed intrinsic f_t of 144 GHz at a drain current of 150 kA/cm², which is a 120% improvement compared to the baseline design. The optimized device exhibits a peak f_t of 225 GHz at a four times larger current density of 4 × 10^5 A/cm² compared to a maximum f_t of 100 GHz at 1 × 10^5 A/cm² for the baseline device (Fig. 9). The large g_m is nearly independent of frequency and the transsusceptance remains near zero

(Fig. 8). The gate step transient is very short and critically damped, so that no oscillatory drain current behavior is seen (Fig. 6). The gate self-conductance (Re y_{11}) is the smallest of all devices in Table II, due to its short gate length, and the larger dielectric relaxation and plasma frequencies resulting from the higher channel carrier concentration. The characteristics of the optimized device 15 have also been examined for 77 K operation. Simulations show that increased channel resistance due to dopant freeze-out is compensated by the enhanced overshoot velocities, and bandgap narrowing. Effective mass changes are minor effects and mutually compensating. The HJ-VFET optimized at 300 K but operated at 77 K shows excellent temperature-independent operation, but suffers from launcher-limited transport (f_t = 167 GHz at I_D = 95 kA/cm^2). For best performance at 77 K the device would require reoptimization following the guidelines in Section VII.

IX. Conclusions

The insight provided by the analysis facilitated the establishment of device design principles for highest speed operation. Ballistic electron injection and the multidimensional dipole layer are the key issues in heterojunction VFET's compared to conventional FET's. Optimized AlGaAs/GaAS VFET's were shown to reach cutoff frequencies up to 100 GHz for "normal" current densities below 1×10^5 A/cm^2, while peak cutoff frequencies up to 225 GHz are possible for current densities as large as 4×10^5 A/cm^2. Fabricated devices never reached such performance levels because their channel doping was deliberately set low in order to reduce scattering and effect quasi-ballistic transport [12]. As shown here, transport became channel-limited and a retarding dipole layer was formed at the heterojunction. Maximum measured transconductances of 60 mS/mm did not stimulate any high-frequency characterization [4]. Our work shows that proper control of ballistic injection under multidimensional hot-electron conditions requires careful device optimization, which would be difficult to achieve without the microscopic insight provided by accurate device simulation. Finally, this study shows that HJ-VFET's should be reconsidered for applications for which both high speeds and the largest current densities are required.

Acknowledgment

The authors acknowledge fruitful discussions with Dr. J. Wendt of Sandia National Laboratories, Prof. L. Eastman of Cornell University. The optimization runs have been performed partly on the IBM 3090-600J supercomputer at the Cornell Center for Theory and Simulation and Hewlett Packard 9000/300 and 400 series engineering workstations.

References

[1] W. G. Oldham and A. G. Milnes, "n-n semiconductor heterojunctions," *Solid-State Electron.*, vol. 6, pp. 121-312, 1963.
[2] K. Tomizawa, Y. Awano, N. Hashizume, and M. Kawashima, "Simulation of GaAs submicron FET with hot electron injection structure," *Electron. Lett.*, vol. 19, no. 17, pp. 697-698, Aug. 18, 1983.
[3] U. Mishra, P. A. Maki, J. R. Wendt, W. Schaff, E. Kohn, and L. F. Eastman, "Vertical electron transistor (VET) in GaAS with a heterojunction (AlGaAs-GaAs) cathode," *Electron. Lett.*, vol. 20, no. 3, pp. 145-146, Feb. 2, 1984.
[4] J. R. Wendt, "The effects of hot-electron injection cathodes on the performance of gallium arsenide vertical field-effect transistors," Ph.D. dissertation, Cornell University, Ithaca, NY, 1988.
[5] S. R. Weinzierl, "Two-dimensional Monte Carlo simulation of submicron unipolar and bipolar compound semiconductor devices with ballistic injection cathodes," Ph.D. dissertation, Cornell University, Ithaca, NY, 1992.
[6] A. Al-Omar and J. P. Krusius, "Self-consistent Monte Carlo Study of high field carrier transport in graded heterostructures," *J. Appl. Phys.*, vol. 62, no. 9, pp. 3825-3835, Nov. 1, 1987.
[7] S. E. Laux, "Techniques for small signal analysis of semiconductor devices," *IEEE Trans. Electron Devices*, vol. ED-32, no. 10, pp. 2028-2037, 1985.
[8] C. Moglestue, "A Monte Carlo particle study of the intrinsic noise figure in GaAs MESFET's," *IEEE Trans. Computer-Aided Des.*, vol. CAD-4, no. 4, pp. 536-540, 1985.
[9] S. R. Weinzierl and J. P. Krusius, "Conditions for ballistic injection across AlGaAs/GaAs heterojunctions in the presence of lateral space charges," submitted for publication.
[10] Y. H. Won, K. Yamasaki, T. Daniels-Race, P. J. Tasker, W. J. Schaff, and L. F. Eastman, "A high voltage-gain GaAs vertical field effect transistor with an InGaAs/GaAs planar-doped barrier launcher," *IEEE Electron Device Lett.*, vol. 11, no. 9, pp. 376-378, Sept. 1990.
[11] J. Tang and K. Hess, "Investigation of transient electronic transport in GaAs following high energy injection," *IEEE Trans. Electron Devices*, vol. ED-29, no. 12, pp. 1906-1911, Dec. 1982.
[12] *J. R. Wendt, personal communication.*

Steven R. Weinzierl received the A.B. degree in physics and mathematics/computer science from Vassar College in 1986, and the Ph.D. degree in electrical engineering from Cornell University, Ithaca, NY, in 1992.

Currently he is a research staff member at Solid State Measurements, Inc., Pittsburgh, PA, where he is developing software for semiconductor materials and device characterization based on spreading resistance and capacitance-voltage measurements.

Dr. Weinzierl is a member of Phi Beta Kappa.

J. Peter Krusius (M'79-SM'84) received the Ph.D. degree in electron physics from the Helsinki University of Technology, Helsinki, Finland.

He did research on semiconductor physics at the Institute of Physics of the University of Dortmund (Dortmund, Germany), the Electron Physics Laboratory at Helsinki University of Technology, and the Semiconductor Laboratory at the Technical Research Center of Finland before coming to Cornell University, Ithaca, NY, on a Fulbright Fellowship. He is currently Full Professor of Electrical Engineering at the School of Electrical Engineering at Cornell. He teaches in the area of solid state electronics. He also pursues vigorous research programs in the areas of nanoelectronics, advanced electronic packaging, and device physics. He leads the Joint Services Electronics Program at Cornell. During the academic year 1988-1989 he was on sabbatical leave at IBM T. J. Watson Research Center, Yorktown Heights, NY.

Dr. Krusius is a member of ECS, MRS, and APS.

the carriers in the drift-diffusion device analysis program. This study suggests that, for shorter gate length MESFET's, it is possible to obtain a reasonably accurate simulation with a modified drift diffusion simulator such as PISCES. This results in a much faster computation than with a Monte Carlo simulator, and makes it possible to use the device parameter extraction capabilities of PISCES.

References

[1] C. K. Williams, T. H. Glisson, J. R. Hauser, M. A. Littlejohn, and M. F. Abusaid, "Two-dimensional Monte Carlo simulation of a submicron GaAs MESFET with a nonuniformly doped channel," *Solid-State Electron.*, vol. 28, pp. 1105–1109, 1985.

[2] Y.-K. Feng and A. Hintz, "Simulation of submicrometer GaAs MESFET's using a full dynamic transport model," *IEEE Trans. Electron Devices*, vol. 35, pp. 1419–1431, 1988.

[3] F. A. Buot, "Two-dimensional numerical modeling of HEMT using an energy transport model," *COMPEL*, vol. 6, pp. 45–52, 1987.

[4] C. M. Snowden and A. Loret, "Two-dimensional hot-electron models for short-gate-length GaAs MESFET's," *IEEE Trans. Electron Devices*, vol. ED-34, pp. 212–223, 1987.

[5] M. R. Pinto, C. S. Rafferty, and R. W. Dutton, *PISCES-II User's Manual*, Stanford Univ., Stanford, CA, 1984.

[6] R. W. McColl, R. L. Carter, J. M. Owens, and T.-J. Shieh, "GaAs MESFET simulation using PISCES with field-dependent mobility-diffusivity relation," *IEEE Trans. Electron Devices*, vol. ED-34, pp. 2034–2039, 1987.

[7] C.-S. Chang and D.-Y. S. Day, "Analytic theory for current-voltage characteristics and field distribution of GaAs MESFET's," *IEEE Trans. Electron Devices*, vol. 36, pp. 269–280, 1989.

[8] M. A. Khatibzadeh and R. J. Trew, "A large-signal, analytical model for the GaAs MESFET," *IEEE Trans. Electron Devices*, vol. 36, pp. 231–238, 1988.

[9] K. Yamasaki and M. Hirayama, "Determination of effective saturation velocity in n^+ self-aligned GaAs MESFET's with submicrometer gate lengths," *IEEE Trans. Electron Devices*, vol. ED-33, pp. 1652–1658, 1986.

[10] Y.-K. Feng, "New v(E) relationship for GaAs," *Electron. Lett.*, vol. 21, pp. 453–454, 1985.

[11] Y.-C. Wang and Y.-T. Hsieh, "Velocity overshoot effect on a short-gate microwave MESFET," *Int. J. Electron.*, vol. 47, pp. 49–66, 1979.

Space-Charge Effects in Ballistic Injection Across Heterojunctions

S. R. Weinzierl and J. P. Krusius

***Abstract*—Conditions under which ballistic injection across heterojunctions is suppressed in unipolar FET devices has been examined using two-dimensional Monte Carlo simulation. Gate-induced lateral space charges influence via macroscopic current continuity the dipole layer at the heterojunction. A retarding dipole layer is shown to result in ballistic electron fractions and transit times comparable to those found in homojunction devices. Guidelines for avoiding the formation of a retarding dipole layer are given.**

Semiconductor field-effect devices utilizing hot-electron anodes, in particular ballistic injection across a heterojunction, have long held promise as high-speed switches [1]. Overshoot velocities are thought to arise downstream from the heterojunction as injected electrons convert potential into kinetic energy. By maintaining overshoot velocities for hundreds of nanometers into the channel, these near-ballistic electrons should provide a substantially reduced transit time. However, this description of electron injection is overly naive considering recent results from one-dimensional self-consistent Monte Carlo simulations [2]. Temperature, applied voltage, and launcher height can change the magnitude and direction of the dipole field at the heterojunction and thus profoundly influence the injection process in laterally uniform one-dimensional structures. The largest injected ballistic fraction is achieved, when near-flatband conditions exist at the heterojunction. We consider here for the first time how two-dimensional phenomena, always present in real devices, affect the injection process. Two-dimensional self-consistent ensemble Monte Carlo simulation is used to show that lateral space charges induced by gate electrodes downstream from the heterojunction can also cause dipole moment reversal at the heterojunction and thus suppress ballistic injection.

The self-consistent fully two-dimensional ensemble Monte Carlo method used here has been described elsewhere [3]. While the heterojunction space-charge effect is a generic one, a specific device has to be selected in order to study it quantitatively. A vertical FET (VFET) with a cross section of 800 nm × 350 nm (Fig. 1), identical to a device recently examined in a full optimization study, has been chosen here [4]. The $Al_xGa_{1-x}As$ heterojunction launcher consists of a 75-nm region, in which the Al mole fraction x increases linearly from 0 to 22%, followed by a 100-nm region with a constant mole fraction of 22%. Current in the 300-nm-long channel is controlled by two ideal 200-nm-long Schottky-barrier gate electrodes placed symmetrically on both sides 100 nm downstream from the heterojunction.

The simulated electron concentration for negative applied gate voltages (more depletion) shows that a retarding dipole layer is formed at the heterojunction via sign reversal at higher drain voltages (Fig. 2). Note that this result has been obtained by fulfilling nonequilibrium transport equations and Poisson's equation without simplifying approximations. The observed phenomenon can be explained by extending the one-dimensional theory for flatband conditions [2] into two dimensions. For laterally uniform one-dimensional injection, the flat band at the heterojunction will prevail for all applied voltages for which the following macroscopic current continuity relation is satisfied:

$$n_{AlGaAs} v_{inj} \leq n_{Ch} v_{Ch}. \tag{1}$$

Here n_{AlGaAs} and n_{Ch} denote the actual carrier densities in the launcher and the channel, and v_{inj} and v_{Ch} the average drift velocities for injected electrons at the heterojunction and downstream from it, respectively. Note that the two carrier concentrations are not solely determined by the local doping densities, but also influenced by carrier spillover and transport effects. Local doping densities provide, however, a good starting point for estimating n_{AlGaAs} and n_{Ch}. Equation (1) is a direct consequence from the local enforcement of current continuity across the heterojunction. As previously shown, the average electron velocities in (1) will necessarily include the effects of ballistic electrons and quantum-mechanical reflection processes at the heterojunction, because they are determined by an integral over the entire local electron distribution function.

In two dimensions, current continuity is no longer fulfilled locally, but over each cross section of the device. The throughput in

Manuscript received March 28, 1991; revised February 14, 1992. This work was supported by the Joint Services Electronics Program under Contract F49620-90-C-0039, monitored by AFOSR (Dr. G. Witt). The review of this brief was arranged by Associate Editor M. Shur.

The authors are with School of Electrical Engineering, Cornell University, Ithaca, NY 14853.

IEEE Log Number 9200398.

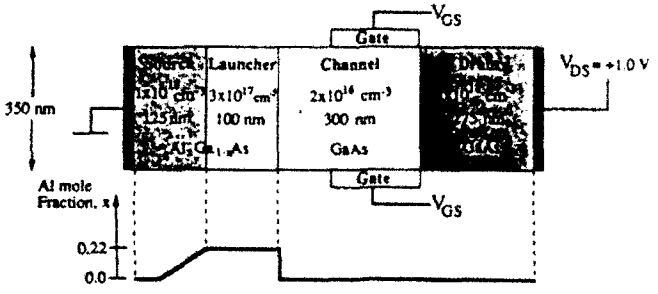


Fig. 1. Cross section of heterojunction VFET. The $Al_xGa_{1-x}As$ grading profile (mole fraction x as function of position) and applied voltages are also shown.

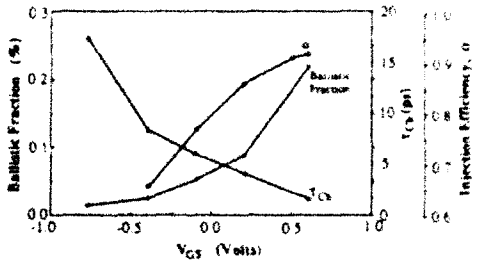


Fig. 3. Ballistic fraction (percent) in center of channel 10 nm downstream from heterojunction, average channel transit time τ_{Ch}, and injection efficiency α as a function of gate voltage V_{GS} with V_{DS} = +1.0 V and T = 300 K. All results are from Monte Carlo method without approximations.

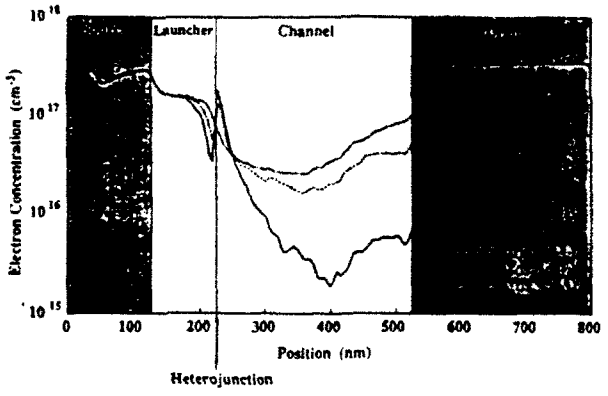


Fig. 2. Electron concentration, from Monte Carlo method, as a function of position along the center of the channel for the gate voltages: V_{GS} = −0.8 V, −0.1 V, and +0.6 V. V_{DS} = +1.0 V. A Schottky-barrier height of 0.8 V was assumed. Note that carrier concentrations in heavily doped regions are lower than the doping densities because of fully included donor statistics.

the channel downstream from the heterojunction will also be limited by the depletion regions modulated by the lateral gate electrodes. One-dimensional depletion theory can be used to obtain an expression for the widths of the lateral depletion regions ΔW, which combined with the channel width W determine the current throughput. Therefore, the current continuity equation for two-dimensions reads

$$\frac{n_{AlGaAs}}{n_{Ch}} \le \frac{v_{Ch}}{v_{inj}}\left(1 - \frac{2\,\Delta W}{W}\right) \le \frac{v_{Ch}}{v_{inj}}\left(1 - \frac{2}{W}\sqrt{\frac{2\epsilon}{qN_{Ch}}\left(V_{bi} - V_{GS} - \frac{k_B T}{q}\right)}\right) = \frac{v_{Ch}}{v_{inj}} F_{Ch}. \quad (2)$$

Here W is the full lateral width of the device, ΔW the depletion width, ϵ the dielectric constant, V_{bi} the built-in potential of the metal–semiconductor junction, V_{GS} the applied gate–source voltage, k_B Boltzmann's contant, and T the lattice temperature. The channel width factor F_{Ch} in (2) is always less than unity.

The two average velocities in (2) always satisfy $v_{Ch} \le v_{inj}$, since the upper limit corresponds to quasi-ballistic injection and transport for all electrons. In addition, n_{Ch} should be smaller than n_{AlGaAs} because of heavy doping in the source region. Consequently, the inequality in (2) does not usually hold, and therefore a retarding dipole layer will be formed. Fig. 2 illustrates these conditions. The effect of the dipole layer reversal on ballistic injection is clearly demonstrated by the Monte Carlo results in Fig. 3, which show the fraction of ballistic electrons 10 nm downstream from the heterojunction as a function of V_{GS}. The ballistic fraction has been calculated by integrating numerically over the ballistic peak in distribution function from $k_z = 4.0 \times 10^6$ cm^{-1} to infinity. The ballistic fraction is reduced from 25% to zero as the gate bias is dropped from V_{GS} = +0.6 V to V_{GS} = −0.6 V (more depletion). This rapid drop-off of the ballistic fraction as a function of the gate voltage exactly replicates the behavior of the channel width factor F_{Ch} for the present case with W = 350 nm. Thus we have shown that lateral space charges control electron injection across the heterojunction via the dipole layer reversal mechanism.

The average transit time τ_{Ch} of electrons across the channel, computed directly as an estimator from the self-consistent Monte Carlo results (Fig. 3), is significantly reduced as the ballistic fraction increases. This transit time varies by a factor of 10 for a 1.5 V change in V_{GS}. This strong reduction can be explained by the presence of a larger number of quasi-ballistic electrons and an increased heterojunction injection efficiency α defined as

$$\alpha = 1 - \frac{N_{Back}}{N_{Fwd}}. \quad (3)$$

Here N_{Back} and N_{Fwd} denote the number of electrons injected upstream and downstream from the heterojunction, respectively. α computed directly from the Monte Carlo results without approximations is also given in Fig. 3. One observes a tradeoff between the maximum ballistic injection efficiency and the acceptable gate voltage swing between the open and pinched off states of the channel.

From the above it is clear how to avoid the reversal of the dipole layer at the heterojunction with all its adverse consequences for steady-state and transient device operation. Two primary means are suggested here to help satisfy (2): a) Place the gates higher upstream in the channel, but not too close in order not to increase the gate–source capacitance excessively. It may also be helpful to shorten the gate length. A gate placement closer to the source would shift the point of minimum lateral width (bottleneck) into an area, where either n or v is higher. b) Increase the channel doping to boost the electron concentration in the channel, but not too high in order not to increase the ionized impurity scattering. Other means include a nonuniform channel cross section or nonuniform channel doping, but these are rather difficult to achieve in practice. Equation (2) also clearly explains why fabricated VFET devices have never reached expected performance levels as measured by transconductance and cutoff frequency [5], [6]. The full multiparameter optimization of heterojunction VFET devices for high-speed and

high-current operation discussed in detail elsewhere supports the above conclusions [4].

References

[1] M. Heiblum, M. Nathan, D. C. Thomas, and C. M. Knoedler, "Direct observation of ballistic transport in GaAs," *Phys. Rev. Lett.*, vol. 55, no. 20, pp. 2200–2203, Nov. 11, 1985.

[2] A. Al-Omar and J. P. Krusius, "Conditions for space-charge reversal at thermionic heterojunctions designed for ballistic electron injection," *IEEE Electron Device Lett.*, vol. 9, no. 2, pp. 81–83, Feb. 1988.

[3] S. Weinzierl and J. P. Krusius, "Lateral space charge effects on ballistic electron transport across graded heterojunctions," *Solid-State Electron.*, vol. 32, no. 12, pp. 1557–1561, Dec. 1989.

[4] ——, "Heterojunction vertical FET's revisited: Potential for 200 GHz large current operation," *IEEE Trans. Electron Devices*, vol. 39, no. 5, pp. 1050–1055, May 1992.

[5] U. Mishra, and P. A. Maki, J. R. Wendt, W. Schaff, E. Kohn, and L. F. Eastman, "Vertical electron transistor (VET) in GaAs with a heterojunction (AlGaAs-GaAs) cathode," *Electron Lett.*, vol. 20, no. 3, pp. 145–146, Feb. 2, 1984.

[6] K. Tomizawa, Y. Awano, N. Hashizume, and M. Kawashima, "Simulation of GaAs submicron FET with hot electron injection structure," *Electron. Lett.*, vol. 19, no. 17, pp. 697–698, Aug. 18, 1983.

PROCEEDINGS REPRINT

SPIE—The International Society for Optical Engineering

Reprinted from

Ultrafast Lasers Probe Phenomena in Semiconductors and Superconductors

24–25 March 1992
Somerset, New Jersey

Volume 1677

Investigation of the Role f Free Carrier Screening During the Relaxation of Carriers Excited by Femtosecond Optical Pulses

J. E. Bair and J. P. Krusius

Cornell University, Schools of Applied Physics and Electrical Engineering,
Ithaca, NY 14850

ABSTRACT

The role of free carrier screening, in the ultrafast relaxation of optically excited carriers, is reassessed using the ensemble Monte Carlo technique. The conventional static screening approximation is compared to a new dynamic screening model. Evolution of the nonequilibrium dynamic dielectric function and its consequences for the carrier scattering are examined. It is shown that dynamic screening results in significant enhancement of both the carrier-carrier and polar optic phonon scattering rates. Relaxation times for the dynamic screenin, model are found to be dramatically shorter than those for the static screening model. Methods of experimentally differentiating between the two models are proposed.

1. INTRODUCTION

In the last few years the femtosecond relaxation of optically excited electron-hole plasmas has received considerable interest. A number of investigations, both experimental[1-3] and simulation,[4-7] have drawn attention to carrier-carrier scattering as an important mechanism through which the relaxation occurs. Until recently, carrier-carrier scattering has been exclusively modeled using a static screening approach. Evidence has been accumulating that this approach may be inadequate. Calculations show that static screening seriously underestimates the carrier-carrier scattering rates.[8,9] Further, recent experiments have reported carrier-carrier scattering rates significantly larger than are possible within the static screening approximation[3]. Recently, a molecular dynamics approach combining free carrier screening and carrier-carrier scattering has succeeded in improving correlation with experiment.[2,5,7] In this work the effect of free carrier screening is examined using an ensemble Monte Carlo simulation. A new model of free carrier screening has been developed that fully includes both the frequency and wavelength dependence of the free carrier dielectric function. In contrast to the molecular dynamics approach, this new model operates within the traditional ensemble Monte Carlo method and can be generalized to other situations. In order to investigate the role of free carrier screening in the relaxation of these optically excited electron-hole plasmas, simulations of femtosecond optical pulse-probe experiments were performed incorporating both this new model and a standard long wavelength static approximation on $In_{0.53}Ga_{0.47}As$ thin films. The number of physical processes to be considered has been minimized, and the role of free carrier screening emphasized, by limiting the energy of the exciting photons to within 100 meV of the band gap. This allows the conduction band upper valleys and split off hole band to be neglected. It also maximizes the amount of free carrier screening for a given carrier density.

2. FREE CARRIER SCREENING

The models of free carrier screening used in this investigation are based on the Lindhard dielectric function. This formula has the advantages that it fully accounts for both the energy and wavelength dependence of the linear dielectric functions, and can be calculated for an arbitrary distribution of free carriers so that no assumptions about the form of the nonequilibrium distribution function need to be made. The dielectric function for a system of free carriers is given by the Lindhard formula as:

0-8194-0838-7/92/$4.00

$$\varepsilon(\vec{q},\omega) = \varepsilon_0 + \frac{4\pi e^2}{q^2}\sum_{k}\frac{f(\vec{k}) - f(\vec{k}+\vec{q})}{E(\vec{k}+\vec{q}) - E(\vec{k}) - \frac{h\omega}{2\pi} + \frac{ih\alpha}{2\pi}} \tag{1}$$

Where ε_0 is the dielectric constant of the semiconductor in the ground state. If the static long wavelength limit is taken, this is simplifies to:

$$\varepsilon(\vec{q}) = \varepsilon_0 - \frac{4\pi e^2}{q^2}\sum_{k}\frac{\vec{q}\cdot\nabla_k f(\vec{k})}{\vec{q}\cdot\nabla_k E(\vec{k})} \tag{2}$$

This is the starting point for most current models of free carrier screening. Clearly this approximate expression can not express the full complexity of the more accurate expression Eqn. (1). For purposes of this investigation we define the free carrier dielectric function as:

$$\varepsilon_{fc}(\vec{q},\omega) = \frac{\varepsilon(\vec{q},\omega)}{\varepsilon_0} \tag{3}$$

The relationship between the free carrier dielectric function and the carrier scattering rates is:

$$\lambda(\vec{k}_1,\vec{k}_2) = \frac{\lambda_0(\vec{k}_1,\vec{k}_2)}{|\varepsilon_{fc}|^2} \tag{4}$$

where $\lambda_0(k_1,k_2)$ is the scattering rate neglecting screening by free carriers and $\lambda(k_1,k_2)$ is the scattering rate including free carrier screening. Thus, it is really the inverse of the free carrier dielectric function that is of interest in this work.

The screening models used in this work are derived using Eqn. (1) and Eqn. (2), with the simplifications that anisotropy in the carrier distribution functions and band structure are ignored, and an approximate parabolic band structure is used in calculating the dielectric function. The resulting dielectric function is isotropic in momentum (**k**) space. In the Monte Carlo simulation the dielectric function is recalculated self-consistently from the carrier distribution functions after each 5 fs time step. The contributions of all three carrier types (electrons, heavy holes, and light holes) are included. The static long wavelength model used here is similar to that proposed by Osman and Ferry[6].

3. FORMULATION AND IMPLEMENTATION

The relaxation dynamics of the optically excited carriers is studied using the ensemble Monte Carlo approach, including electrons and holes, to simulate the evolving distribution function, and its interaction with the optical field. The distribution function includes all three momentum (**k**) space dimensions and one spatial dimension normal to the surface of the thin film and parallel to the photon beam. The distribution function is assumed homogeneous in the two lateral directions in the plane of the film. Inhomogeneities arising from the optical excitations are fully included with carrier motion governed by a self-consistent electric field (solution to Poisson's equation).

The model of the band structure includes the conduction band, and the heavy and light hole bands valence bands around the fundamental optical gap in the center of the zone. The bands are described by a four band **k.p** method, with perturbative corrections from higher bands, as given by Kane[10]. The perturbative terms are necessary to get the correct sign for the heavy hole mass and include band

warping. The resulting heavy hole band is parabolic but warped, while the other two bands are both nonparabolic and warped. The split off band and upper conduction band valleys are neglected because of the small photon energies used in this investigation.

All important carrier scattering mechanisms are included: carrier-carrier, phonon, ionized impurity and alloy scattering. Nonpolar optical and deformation potential phonon scattering were derived using the method of deformation potential operators of Pikus and Bir[11] as applied to the four band **k.p**[12]. To further simplify these results the expression for nonpolar optic phonon scattering was evaluated at the band edge and the effective deformation potential of Lawaetz[13] was used for the valence bands. Alloy scattering is included through the elementary approach of Harrison and Hauser[14]. Polar optic phonon, piezoelectric, and ionized impurity are handled using the well known formulas including proper overlap integrals derived from the **k.p** structure. Both inter- and intra-band scattering are included in the valence bands for all single particle scattering processes. Carrier-carrier scattering is treated following the method of Brunetti et al[15], with the improvements suggested by Mosko et al[16],. and with the simplification that particles do not change bands. The polar optic phonon, ionized impurity, piezoelectric, and carrier-carrier scattering rates are each self-consistently screened using the screening models described above. Degenerate statistics are used for all scattering processes through the rejection method.

The optical excitation of electron-hole pairs is handled self-consistently. The number of carriers generated at a given time is calculated using the instantaneous value of the carrier distribution functions with the pulse altered to reflect the absorbed energy. Excitation rates are calculated from Fermi's golden rule with the momentum matrix elements calculated from **k.p** theory. Both the anisotropy of the optical matrix elements and their energy dependence are included and reflected in the excited carrier distributions.

4. RESULTS

Simulated pulse-probe experiments were performed for a 0.25 μm $In_{0.53}Ga_{0.47}As$ thin film using both the dynamic and static screening models discussed above. Both the excitation and probe pulses have a secant squared intensity profile with 100 fs FWHM and a photon energy of 810 meV. This corresponds to a combined carrier energy of 60 meV above the optical gap of 0.75 eV. The intensity of the excitation pulse was 5.0×10^{13} eV/cm^2 and the probe pulse was assumed to have negligible intensity.

4.1 Free Carrier Dielectric Function

Figs. 1-4 show the reciprocal of the dynamic free carrier dielectric function squared as extracted from the simulation at 0, 100, 200, and 1000 fs after the initial excitation. In all four cases the expected spectrum of plasma modes is evident to the left of each plot at high energies. The plasma frequency can be seen to increase in energy from 0 to 100 fs due to the increase in carrier density as is expected. The most interesting feature is the ridge extending diagonally across Figs. 1-3. The ridge is found to correspond to the plasma spectrum of the heavy holes taken alone. The size of the ridge decreases with increasing delay until at 1000 fs the dielectric function takes the form expected of an equilibrium carrier distribution. This feature is a consequence of the highly nonequilibrium heavy hole distribution at early times. It results from the heavy holes being excited initially into an extremely narrow region of momentum (**k**) space. This results in an unusually sharp resonance with the heavy holes for potentials in this region of frequency and wavelength. As the heavy hole distribution relaxes toward equilibrium, the ridge shrinks in size and eventually disappears due to the dispersal of heavy holes in **k** space and the resulting broadening of the heavy hole resonance. The smaller bump to the left of the main ridge appearing at delays of 100 and 200 fs has a similar origin. This results from a phonon replica of the initial heavy hole distribution due to absorption of optical phonons.

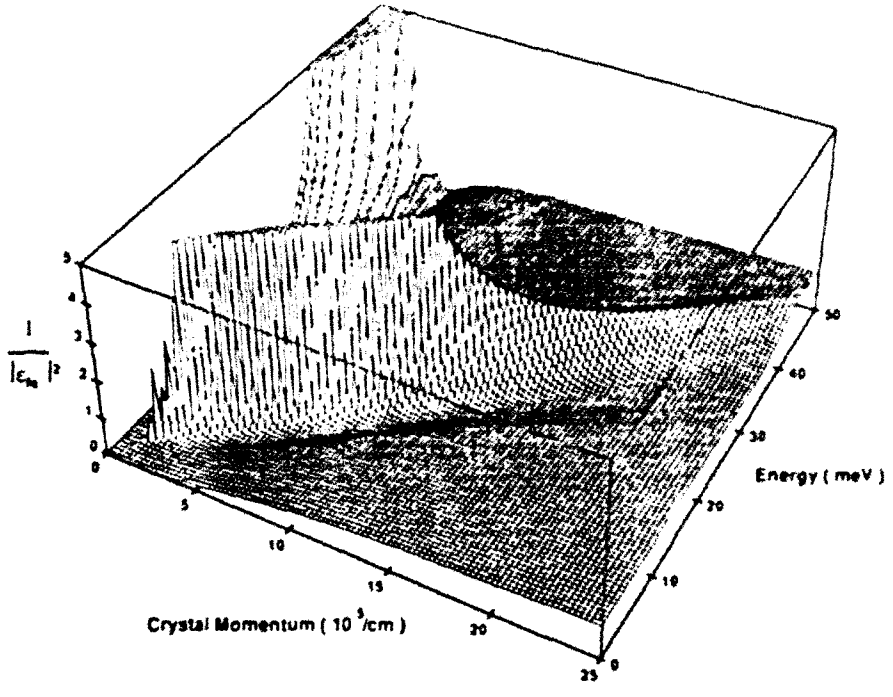


Fig. 1. Simulated nonequilibrium free carrier screening ($|\varepsilon_{fc}|^{-2}$) 0 fs after the initial excitation.

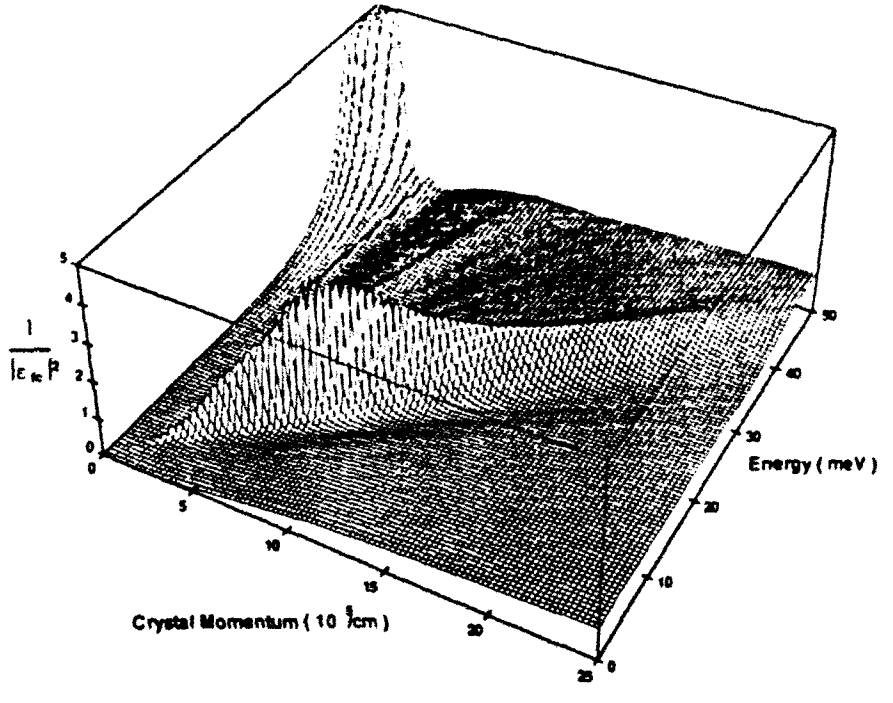


Fig. 2. Simulated nonequilibrium free carrier screening ($|\varepsilon_{fc}|^{-2}$) 100 fs after the initial excitation.

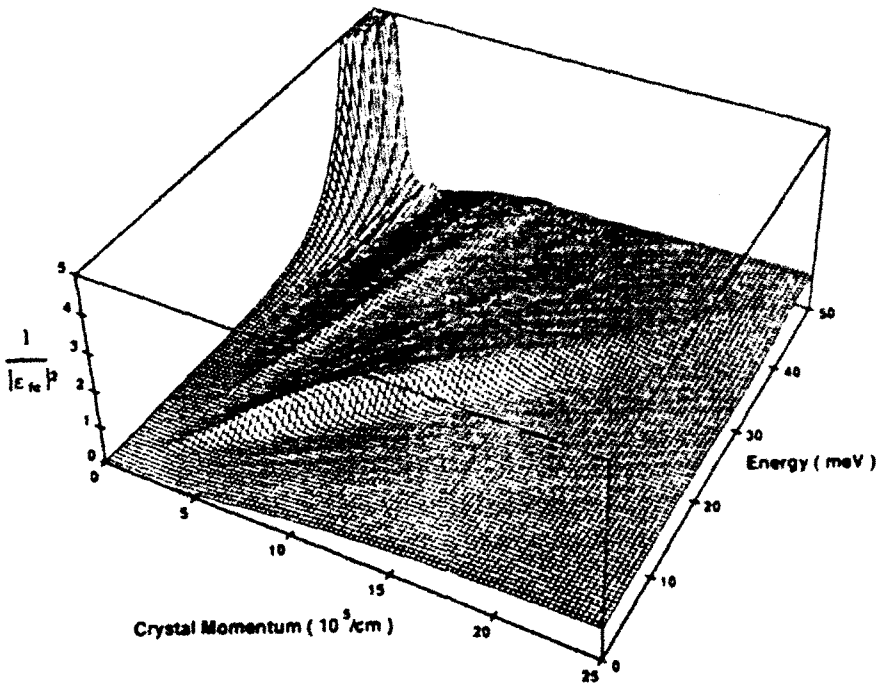


Fig. 3. Simulated nonequilibrium free carrier screening ($|\varepsilon_{fc}|^{-2}$) 200 fs after the initial excitation.

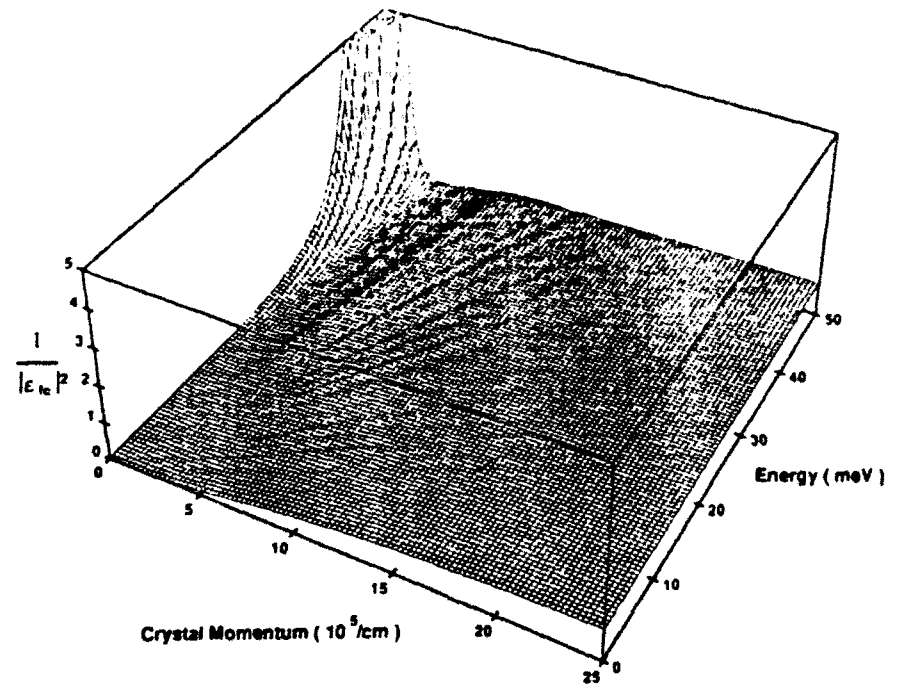


Fig. 4. Simulated nonequilibrium free carrier screening ($|\varepsilon_{fc}|^{-2}$) 1000 fs after the initial excitation.

The large heavy hole resonance crosses the region of energy-momentum space corresponding to carrier-carrier scattering. Thus indicating carrier-carrier scattering should be significantly enhanced at early times. However, in general screening decreases with increasing energy. Thus, even if the resonance is ignored, inelastic scattering processes are over screened by a static screening model, suppressing the scattering rates. This applies to both carrier-carrier and polar optic phonon scattering. For the examples discussed here, the optical phonon energy of 34 meV is too large for polar optic phonon scattering to interact significantly with the heavy hole resonance and the plasma frequency is too large for either polar optic phonon or carrier-carrier scattering do interact with the ordinary plasma modes. Thus, inelastic scattering rates are generally greater with dynamic screening than with static screening, and at early times scattering processes that involve the transfer of small amounts of energy such as carrier-carrier should be greatly enhanced due to the resonance with the heavy holes.

4.2 Carrier Scattering Rates

In order to verify these conclusions the scattering rates for the electrons and heavy holes are shown in Figs. 5 and 6 for both static and dynamic screening. In the case of electron-electron, electron-heavy hole and electron light hole the anticipated enhancement of the carrier-carrier scattering rates at early times is evident. The decay of these scattering rates with increasing delay is also closely correlated with the decline in the heavy hole resonance. In contrast, the heavy hole-heavy hole scattering rate is actually suppressed at early times. This is because most heavy hole-heavy hole events fall slightly forward of the ridge (towards larger k) in energy-momentum space where the screening is slightly greater. At later times, all the carrier-carrier scattering rates except electron-heavy hole scattering are significantly larger that their static counterparts in agreement with the conclusion that static screening generally over screens inelastic scattering events. The magnitude of the difference is smallest for the heavy hole-heavy hole case reflecting the large momentum transfers involved due the flatness of the heavy hole band. Thus this process is only weakly screened in both cases. Electron-heavy hole scattering is an exception to the general enhancement of carrier-carrier scattering since the large differences in carrier mass make this process approximately elastic.

For the polar optic phonon-heavy hole scattering rate the difference between the static and dynamic screening models is small. As in the case of heavy hole-heavy hole scattering, this is due to the flatness of the heavy hole band and the resulting large momentum transfers. In the case of electron-polar optic phonon scattering the scattering rate with dynamic screening is much greater for times after 0 fs. The statically screened scattering rate is initially as large as the dynamic one but is suppressed by carrier screening as carriers are excited around 0 fs. Electron-optical phonon scattering is only lightly screened by the dynamic dielectric function because of the optical phonons large energies.

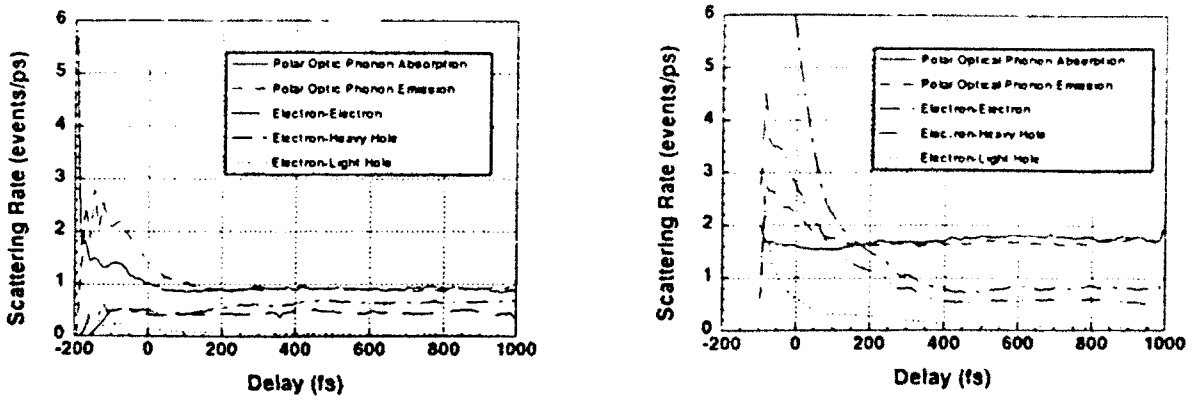


Fig. 5. Ensemble averaged electron scattering rates. Left static screening, right dynamic screening.

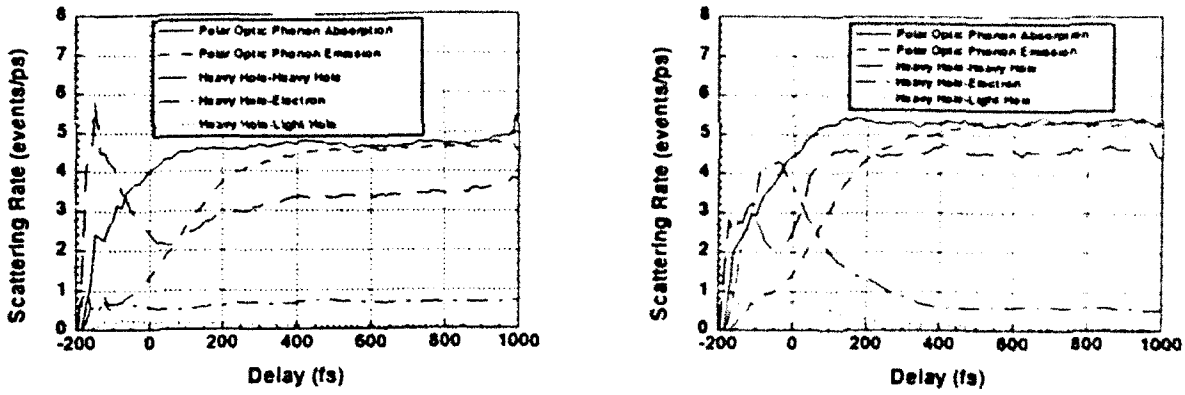


Fig. 6. Ensemble averaged heavy hole scattering rates. Left static screening, right dynamic screening.

4.3 Distribution Function

The difference in scattering rates between the two screening models has significant consequences for the evolution of the respective carrier distribution functions. This is most obvious for the electron distribution functions which are shown in Figs 7 and 8. The effects of the increased carrier-carrier scattering are evident in the rapid washing out of the initial excitation peaks with dynamic screening. Also the rate at which carriers transfer to the bottom of the band is much more rapid. Clearly the electrons relax toward equilibrium much more rapidly with dynamic screening, and carrier-carrier scattering appears to play a larger role even though the dominant scattering mechanism is polar optic phonon in both cases.

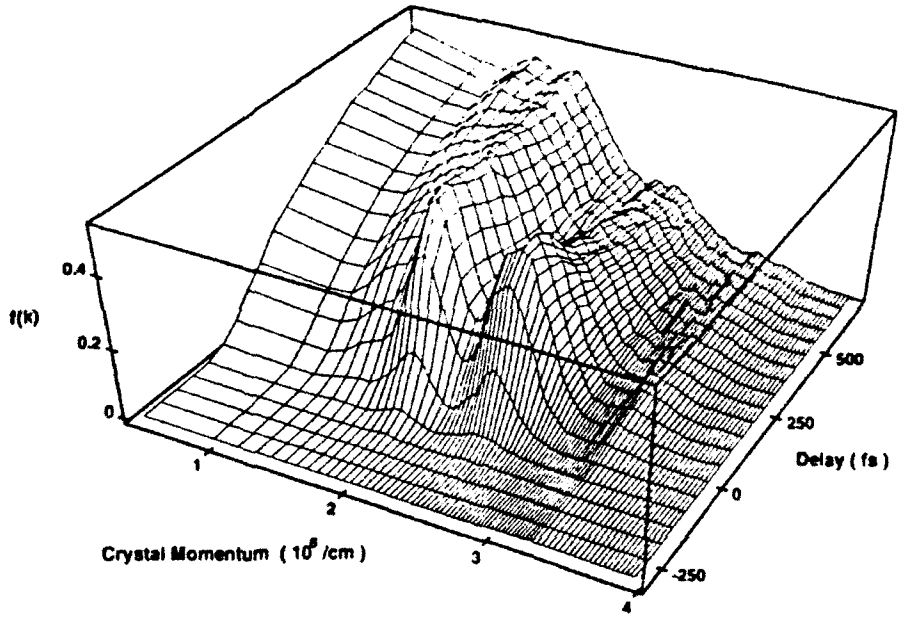


Fig. 7. Evolution of the electron distribution function simulated using static screening.

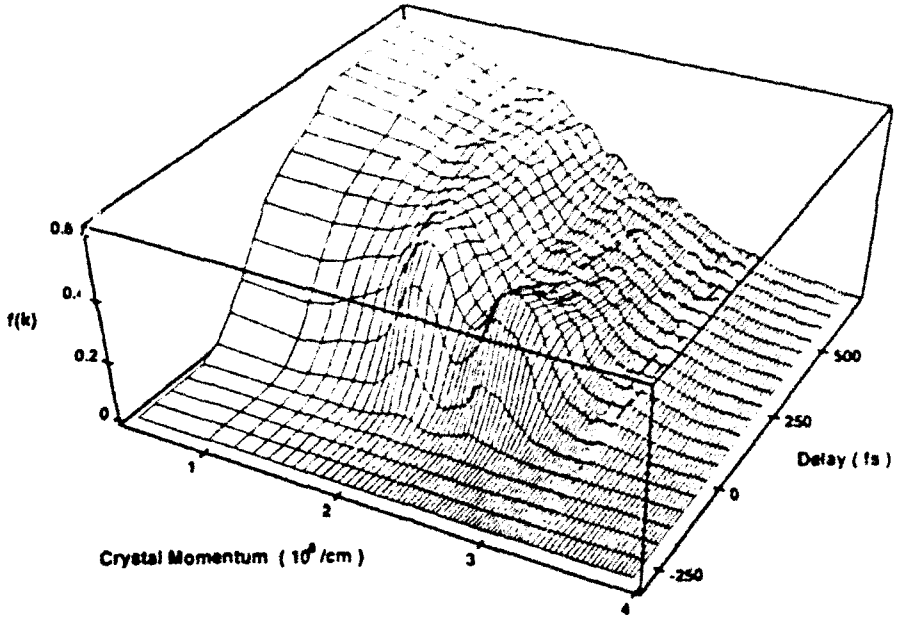


Fig. 8. Evolution of the electron distribution function simulated using dynamic screening.

4.4 Pulse Probe Results

To examine the effect of the screening model on experimentally measurable parameters the probe transmission was calculated in each case for a simulated pulse-probe experiment. These results are shown in Fig. 9. The results are precisely what would be expected. The dynamically screened curve approaches equilibrium much faster than the statically screened one. The peak transmission is also lower for the dynamic case because its larger scattering rates do not allow as many carriers to accumulate in the optically coupled regions.

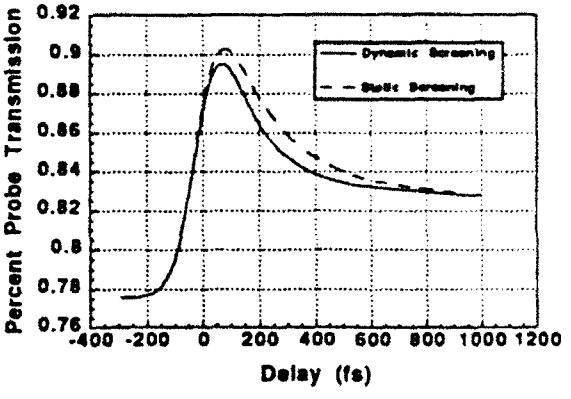


Fig. 9. Simulated probe transmission for both the static and dynamic screening models.

In order to obtain a numerical measure of the difference in relaxation rate between the two cases exponential fits was performed on the tail of each curve for times between 200 and 700 fs after the excitation. The relaxation times obtained were 145 fs for the dynamic screening and 205 fs for static screening, confirming that the dynamically screened carriers relax substantially faster than those that are statically screened. Such a large difference in relaxation times should be obvious in experiments and enable the dynamic screening model to be verified. In Fig. 10 the relationship between

relaxation time and excitation pulse intensity is shown, while in Fig. 11 the relaxation time is plotted versus the photon energy. In both cases only the indicated parameter is varied, all others remain unchanged. It is clear that dynamic screening produces faster relaxation times for all photon energies and pulse intensities. As a function of pulse intensity both curves have the same qualitative behavior with relaxation times gradually increasing with increasing pulse intensity. In contrast the two curves differ qualitatively as a function of photon energy. The static screening curve shows a definite step for photon energies about 35 meV above the band edge. This is absent for dynamic screening. The location of the step in energy corresponds to the first phonon threshold for electrons excited from the heavy hole band. For photon energies greater than this these electrons have sufficient energy to emit optical phonons. The existence of this step is a clear indication relaxation is phonon dominated with static screening while its absence with dynamic screening points to the increased importance of carrier-carrier scattering. The large difference in relaxation times and the presence or absence of this step together provide a means to discriminate experimentally between the two screening models.

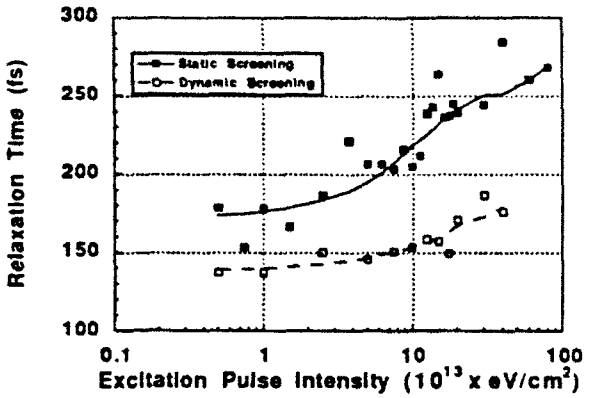


Fig. 10 Comparison of the extracted relaxation rates as a function of excitation pulse intensity.

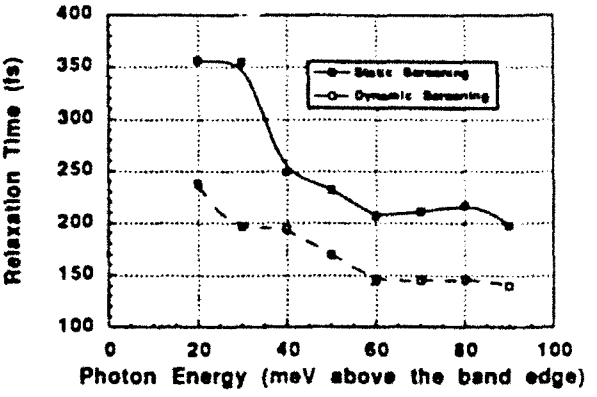


Fig. 11. Comparison of the extracted relaxation rate as a function of the energy of the exciting photons.

5. CONCLUSIONS

It seems clear from the present results that free carrier screening plays a crucial role in the relaxation of carriers excited near the band gap in femtosecond optical probing. Static screening has been found to seriously underestimate both the carrier-carrier scattering rates and the polar optic phonon scattering rates, and lead to significantly longer relaxation times when compared to dynamic screening. It was found that the highly nonequilibrium state of the distribution function results in an unexpectedly sharp resonance with the heavy holes in the dynamic dielectric function that greatly enhances certain carrier-carrier scattering events for times less than 400 fs after the initial excitation. From this we conclude that static screening is inadequate for developing an accurate understanding of these types of experiments. Work is currently underway, in collaberation with the experimental effort of C. Pollock's research group at Cornell, to quantitatively verify these results. It seems clear that future work in modeling these processes must include dynamic screening for significant progress to be made.

6. ACKNOWLEDGMENTS

This work has been supported by the Joint Services Electronics Program at Cornell University (Contract Number F494620-87-C-0044)

REFERENCES

1. P. Becker, H. Fragnito, C. Brito Cruz, R. Fork, J. Cunningham, J. Henry, and C. Shank, "Femtosecond Echoes from Band-to-Band Transitions in GaAs," Phys. Rev. Lett., Vol. 61, pp. 1647-1649, 1988.

2. Thomas Elsaesser, Jagdeep Shah, Lucio Rota, and Paolo Lugli,"Initial Thermalization of Photoexcited Carriers in GaAs Studied by Femtosecond Luminescence Spectroscopy", Phys. Rev. Lett., Vol. 66 No. 13, pp. 1757-1760, 1991.

3. J. A. Kash, "Carrier-carrier scattering in GaAs: Quantitative measurements from hot (e,A^o) luminescence", Phys. Rev. B, Vol. 40 No. 5, pp. 3455-3458, 1989.

4. C. J. Stanton, D. W. Bailey, and K. Hess, "Femtosecond-Pump, Continuum-Probe Nonlinear Absorption in GaAs", Phys. Rev. Letters, Vol. 65 No. 2, pp. 231-234, 1990.

5. R. P. Joshi, R. O. Grondin, and D. K. Ferry, "Monte Carlo simulation of electron-hole thermalization in photoexcited bulk semiconductors," Phys. Rev. B, Vol. 42 No. 9, pp. 5685-5692 , 1990

6. M. A, Osman and D. K. Ferry, "Monte Carlo Investigation of the Electron-Hole Interaction Effects on the Ultrafast Relaxation of Hot Photoexcited Carriers in GaAs", Phys. Rev. B, Vol. 36 No. 11, pp. 6018-6032, 15 October 1987.

7. M. J. Kann, A. M. Kriman, and D. K. Ferry,"Role of electron-electron scattering on ultrafast probe phenomena of photoexcited carriers in GaAs", ,Ultrafast Laser Probe Phenomena in Bulk and Microstructure Semiconductors III, Robert R. Alfano, Editor, Proc. SPIE 1282, pp. 98-108, 1990.

8. Jeff F. Young, Norm L. Henry, and Paul J. Kelly,"Full Dynamic Screening Calculation of Hot Electron Scattering Rates in Multicomponent Semiconductor Plasmas",Solid State Elec., Vol. 32 No. 12, pp. 1567-72, 1989 .

.9 R. Binder, D. Scott, A. E. Paul, M. Lindberg, K. Hennebergerger, and S. W. Koch, "Carrier-carrier scattering and optical dephasing in highly excited semiconductors", Phys. Rev. B, Vol. 45 No. 3, pp. 1107-1115, 1992.

10. E. O. Kane,"The k.p Method",Semiconductors and Semimetals, Eds. R. K. Willardson and A. C. Beer, Vol. 1, pp. 75-100, Academic Press, New York, 1966.

11. G. L. Bir and G. E. Pikus, "Theory Of The Deformation Potential For Semiconductors With A Complex Band Structure", Fiz. Tverd. Tela, Vol. 2 No. 9,p p. 2287-2300, 1960.

12. W. Zawadzki, "Mechamisms of Electron Scattering in Semiconductors", Handbook on Semiconductors, Eds. T. S. Moss and W. Paul, Vol. 1, pp. 713-803, North Holland, Amsterdam, 1982.

13. P. Lawaetz,"Low-Field Mobility and Galvomagnetic Properties of Holes in Germanium with Phonon Scattering", Phys. Rev., Vol. 174 No. 3, pp. 867-880, 1968.

14. J. W. Harrison and J. R. Hauser, "Alloy Scattering in Ternary III-V Compounds", Phys. Rev. B, Vol. 13 No. 12, pp. 5347-5350, 1976.

15. R. Brunetti, C. Jacoboni, V. Dienys, and A. Matulionis, "Effect of Interparticle Collision On Energy Relaxation of Carriers in Semiconductors", Physica B Vol. 134, pp. 369-373, 1985.

16. M. Mosko, and A. Moskova,"Ensemble Monte Carlo simulations of electron-electon scatteriing: Improvements of conventional methods", Phys. Rev. B,Vol. 44, No. 16, pp. 10794-10803, 1991

Band Renormalization and Dynamic Screening in Near Band Gap Femtosecond Optical Probing of InGaAs

J. E. Bair, D. Cohen, J. P. Krusius, C. R. Pollock
Cornell University, School of Electrical Engineering and School of Applied Engineering Physics, Ithaca New York

The effect of band renormalization and dynamic screening in near band edge femtosecond optical probing of $In_{0.53}Ga_{0.47}As$ has been investigated. Measured relaxation times for electrons and holes are on the order of 110 fs. Simulated results, obtained from an ensemble Monte Carlo formulation, are in excellent agreement with measured equal pulse correlation data only if both processes are included. Band renormalization is found to be roughly twice as important as dynamic screening for these conditions.

The use of femtosecond lasers for probing carrier scattering processes in compound semiconductors has become common during the 1980's. In this type of experiment carriers are first excited by an initial optical excitation pulse and then probed by a second pulse after a short time delay. The transmission of the second pulse, or the combined transmission of the two pulses, as a function of the delay is determined by the relaxation of the excited carriers. Numerous experiments of this type have been performed to date [1-5], and theoretical analysis's attempted [6-9], in order to explore the contributions of the fundamental carrier scattering processes to the measured results. Nearly all of this effort has been for carriers excited far from the band edge, and thus primarily concerned with intervalley transfer rates. In this work we investigate the femtosecond carrier relaxation in a largely unexplored energy range, excitation within 100 meV of the fundemental band edge.

In the near band gap regime, several processes traditionally ignored are expected to have increased importance. Significant among these is band renormalization. Despite the higher excitation energy, several groups have observed behavior interpreted as band renormalization, both on femtosecond [1-3] and picosecond [4] time scales. Also, it has recently become clear that an accurate treatment of carrier-carrier scattering including the dynamic free carrier dielectric function is essential in analyzing these experiments [5-7]. Several methods of dealing with this problem have been developed [2,6-8]. However, these methods either make assumptions about the quasi-equilibrium nature of the dielectric function or are difficult to generalize to inhomogenous situations.

In this work we report the first quantitative demonstration of the role of band renormalization and dynamic screening in the initial femtosecond relaxation of carriers optically excited near the band gap. To this end a new ensemble Monte Carlo simulation has been developed which accounts for both dynamic screening of all long range carrier scattering processes and band renormalization. The free carrier dielectric function is obtained directly from the Lindhard (RPA) formula and thus no assumption of the quasi-equilibrium character of the dielectric function or carrier distributions is required. Both dynamic screening and band renormalization are handled with simple extensions of standard Monte Carlo techniques and thus are easily generalized to inhomogenous systems.

The measurements were performed using a tunable NaCl color center laser [10] which generates femtosecond pulses with photon energies near the band gap of the chosen semiconductor $In_{0.53}Ga_{0.47}As$. The measurements described here were performed with 180 fs full-width-half-maximum (FWHM) $sech^2$-shaped pulses of 0.787 eV photons. Thus the electron-hole pairs are excited 37 meV above the fundamental optical gap of 0.75 eV. The transient transmission at 300 K was measured using the equal-pulse correlation technique [11]. The laser was operated at a repetition rate of 164 MHz while the energy of each pulse was 2.5×10^{13} eV/cm^2. A typical experimental result from a 1.0 μm thick $In_{0.53}Ga_{0.47}As$ film on a transparent InP substrate is shown in fig. 1.

The measured results have been analyzed using an ensemble Monte Carlo particle simulation technique, parts of which have been described elsewhere [12]. It includes electrons and holes from the conduction, heavy hole and light hole bands with provisions for band warping and nonparabolicity. All important scattering processes are accounted for including carrier-carrier scattering. Materials parameters for $In_{0.53}Ga_{0.47}As$ have been obtained from Ref. [13]. All carrier processes, including carrier scattering, optical excitation and one dimensional carrier motion, are described self-consistently using the instantaneous values of the carrier distribution function, optical field, electric field and free carrier dielectric function. Contrary to what has been done before, the inclusion of one dimensional carrier transport allows any sample thickness dependence to be fully accounted for.

Unique features of the present simulation method are the inclusion of a dynamic free carrier dielectric function and band renormalization. Rather than introduce dynamic screening through a molecular dynamics approach [7], as several other groups have done, we chose to develop a method within the standard Monte Carlo framework that does not compromise flexibility. This method is based on a direct evaluation of the Lindhard (RPA) dielectric function given by

$$\varepsilon(\mathbf{q},\omega) = \varepsilon_0 + \frac{4\pi e^2}{q^2} \lim_{\alpha\to 0} \sum_{\mathbf{k},n} \frac{f_n(\mathbf{k}) - f_n(\mathbf{k}+\mathbf{q})}{E_n(\mathbf{k}+\mathbf{q}) - E_n(\mathbf{k}) - h\omega + ih\alpha} \qquad (1)$$

where $f_n(\mathbf{k})$ denotes the carrier distribution function, $E_n(\mathbf{k})$ the carrier energy, and ε_0 the static dielectric constant. The sum is taken over all crystal momentum states $\mathbf{k}$ and all bands n. This equation is evaluated and tabulated at the beginning of each Monte Carlo time step and the results are used in the computation of carrier scattering for that time step. To reduce the computational work several simplifying assumptions are made. The anisotropy of the carrier distribution functions, dielectric function, and band structure is neglected, and the band structure is taken to be both parabolic and spherical for the purpose of calculating the free carrier dielectric function. We have also run simulations using a static screening model for comparison [6].

Band renormalization is included within the "quasi-static" approximation developed by Haug and Schmitt-Rink [14]. Within this approximation the energy shift experienced by a state with a crystal momentum $\mathbf{k}$ in a single uncoupled band is given by

$$\begin{aligned} \Sigma_i(\mathbf{k}) &= \Sigma_i^{sx}(\mathbf{k}) + \Sigma_i^{Ch}(\mathbf{k}) \\ \Sigma_i^{sx}(\mathbf{k}) &= -\frac{1}{V}\sum_{\mathbf{k}'} V_s(\mathbf{k}-\mathbf{k}') f_i(\mathbf{k}') \quad , \\ \Sigma_i^{Ch}(\mathbf{k}) &= -\frac{1}{2V}\sum_{\mathbf{k}} [V(\mathbf{k})/\varepsilon_0 - V_s(\mathbf{k}) \end{aligned} \qquad (2)$$

where $\Sigma_i^{sx}(\mathbf{k})$ and $\Sigma_i^{Ch}(\mathbf{k})$ are the screened exchange and coulomb hole contributions to the electron self-energy, $V_S(\mathbf{k})$ and $V(\mathbf{k})$ the statically screened and unscreened coulomb potentials respectively, V the volume of the crystal. In the present simulations this expression is generalized to include the effects of coupling between the conduction and the two valence bands, and the overlap integrals between the Bloch states. This has been implemented assuming a rigid band shift using the value calculated for the Γ point in each band.

This selfconsistent Monte Carlo technique was used to simulate the optical probe experiments. Fig. 1. shows a comparison of both a simulated equal-pulse experiment and a simulated pulse-probe experiment with the actual measured equal pulse correlation data. Since the experiment does not presently give the absolute transmission, all results are presented in a normalized fashion. The fit between the simulated and measured equal-pulse curves is excellent for delays longer than 150 fs. A large coherent artifact, evident for shorter delays, is as expected, since the experiment was performed with both pulses having the same polarization with a 280 fs FWHM pulse autocorrelation. The simulated pulse-probe curve also fits well in the range between 150 - 400 fs but flattens out somewhat too rapidly for longer delays. This is almost certainly due to subtle differences between the two types of experiments. From the excellent correlation between measured and simulated results we conclude that the present model provides a firm basis for understanding femtosecond optical probing in the near band gap regime.

In order to determine the role of free carrier screening and band renormalization in near band gap femtosecond optical probing additional simulations of pulse-probe experiments were performed in which each of the processes was turned off. From the results shown in Fig. 2 it is obvious that band renormalization is responsible for a significantly reduced probe transmission. It is further clear that the static screening shows a slower recovery than dynamic screening. To characterize the overall relaxation results and to extract the relative importance of the two processes, exponential fits were performed for the range 200- 800 fs for each curve in Figs. 1 and 2. The resulting "effective" relaxation times are given in Table. I. It is clear from these results that both processes have strong effects that are indispensable in analyzing such experiments. It is somewhat surprising that band renormalization is by far the most important of these two effects for these conditions.

The significance of band renormalization becomes more clear, if the simulated magnitude and time dependence of the band shifts are examined (Fig. 3). The maximum reduction of the band gap is approximately 14 meV, which is a large fraction of the initial excess carrier energy of 37 meV. This results in dramatic differences in the form of the excited carrier distribution functions due to the large renormalization of the bands during excitation. There is a small recovery in the band gap for delays less than 500 fs, which results primarily from the warming of the very cold heavy holes. This introduces additional transient effects during the relaxation due to changes in the effective photon energy of the probe.

The effect of dynamic screening on the carrier-carrier scattering rates, computed with dynamic screening and static screening, is shown in fig. 4. Band renormalization has been included in both cases. Significant increases in all carrier-carrier scattering rates are observed for dynamic screening. The electron-electron scattering rate is most effected by screening. This is especially true at early times due to the highly non-equilibrium nature of the free carrier dielectric function. This has been discussed briefly in another publication [12] and will be examined in detail in a future publication. The heavy hole-heavy hole and heavy hole-electron scattering rates are less effected because of the larger momentum and smaller energy transfers involved. The electron-polar optic phonon scattering rates are also increased by about 30% for dynamic screening. These results explain why dynamic screening had such a large effect on the observed carrier relaxation processes.

In conclusion, band renormalization and dynamic screening significantly affect near band gap femtosecond optical probing. Both effects significantly reduce measured "effective" relaxation times with band renormalization being by the farther the more important of the two. Dynamic screening markedly increases the important carrier scattering rates, while band renormalization results in changes in the distribution of carriers within the bands and changes the effective probe energy with respect to the band edge. By including these processes in our novel Monte Carlo technique we have succeeded in successfully reproducing measured data. This work demonstrates for the first time quantitatively both the importance and the role of renormalization and dynamic screening in near band gap femtosecond probing.

Acknowledgment. This work has been supported by the Joint Service Electronics Program at Cornell University (Contract Number F49620-90-C-0039, Program Monitor Dr. H. Wittmann of AFOSR). The authors are grateful to Dr. W. Schaff of Cornell University for growing the $In_{0.53}Ga_{0.47}As$ thin film MBE samples and Dr. Steven Weinzierl for collaboration in the Monte Carlo code development.

[1]T. Gong, P. M. Fauchet, Jeff F. Young, and P. J. Kelley, Phys. Rev. B **44**, 6542 (1991).
[2]J. Nunnenkamp, J. H. Collet, J. Klebniczki, J. Kuhl, and K.Ploog, Phys. Rev. B **43**, 14047 (1991).
[3]T. Gong, P. Mertz, W. L. Nighan, Jr., and P. M. Fauchet, Appl. Phys. Lett. **59**, 721 (1991).
[4]H. Roskos, B. Reik, A. Seilmeier, W. Kaiser, and G. G. Baumann, Rhys. Rev. B **40**, 1396 (1989).
[5]T. Elsaesser, J. Shah, L. Rota, and P. Lugli, Phys. Rev. Lett. **66**, 1757 (1991).
[6]M. A. Osmann and D. K. Ferry, Phys. Rev. B **36**, 6018 (1987).
[7]A. M. Kriman, R. P. Joshi, and D. K. Ferry, in Ultrafast Lasers Probe Phenomena in Semiconductors and Superconductors, edited by R. R. Alfano (Proc. SPIE 1677), p. 2.
[8]Tilmann Kuhn, and Fausto Rossi, Phys. Rev. B **46**, 7496 (1992).
[9]C. J. Stanton, D., W. Bailey, and K. Hess, Phys. Rev. Lett. **65**, 231 (1990).
[10]C.P. Yakamyshyn, J.F. Pinto and C.R. Pollock, Opt. Lett. **14**, 621 (1989).
[11]A. J. Taylor, D. J. erskine, and C. L. Tang, Appl. Phys. Lett. **43**, 989 (1983).
[12]J. E. Bair and J. P. Krusius, in Ultrafast Lasers Probe Phenomena in Semiconductors and Superconductors, edited by R. R. Alfano (Proc. SPIE 1677), p. 157. 13.
[13]S. Adachi, J. Appl.Phys. **53**, 8775 (1982).
[14]H. Haug and S. Schmitt-Rink, Progress in Quantum Electronics, **9**, 3 (1984).

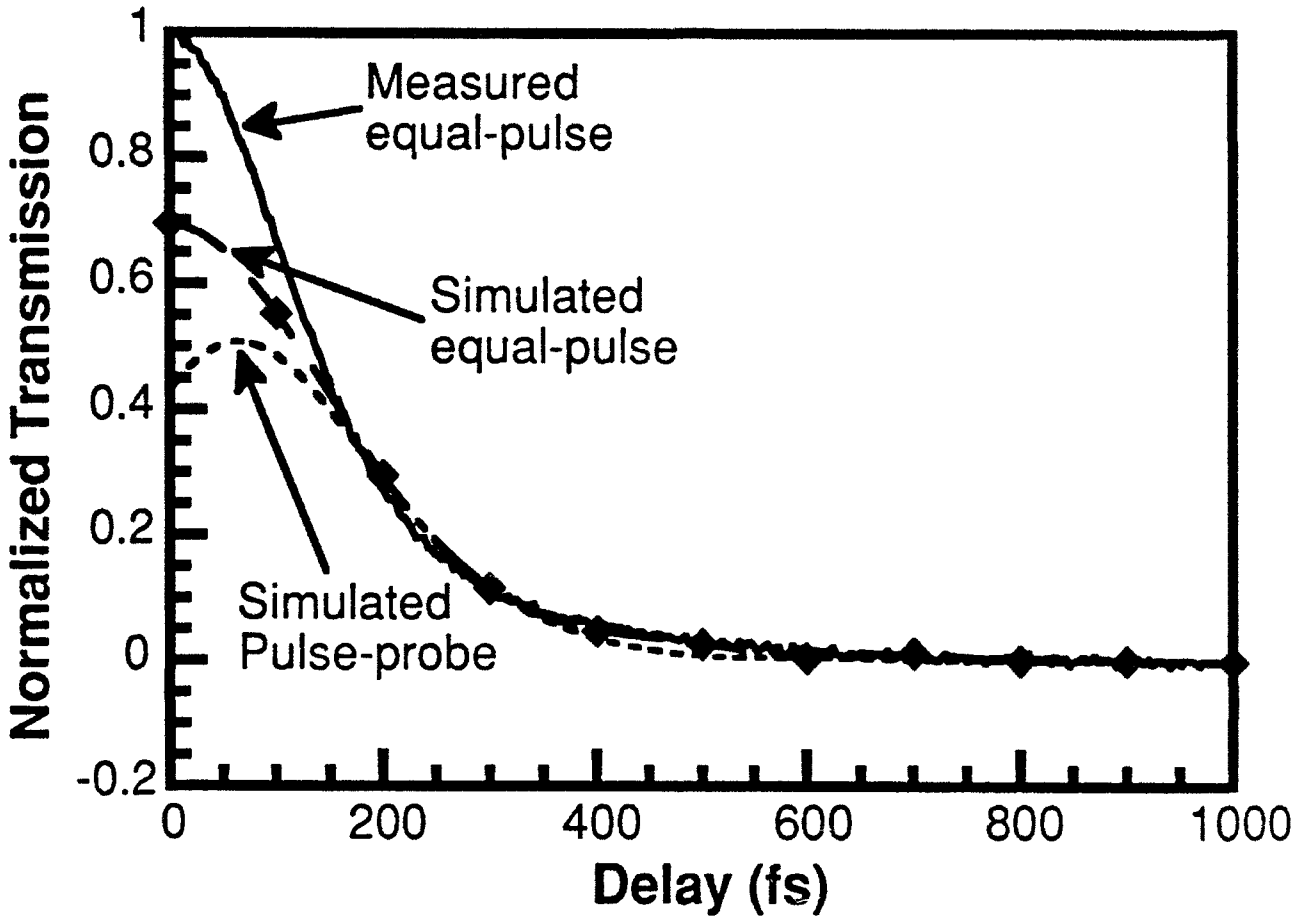
Fig. 1. Measured and simulated normalized optical transmission as a function of the delay between exctitation and probe pulses.

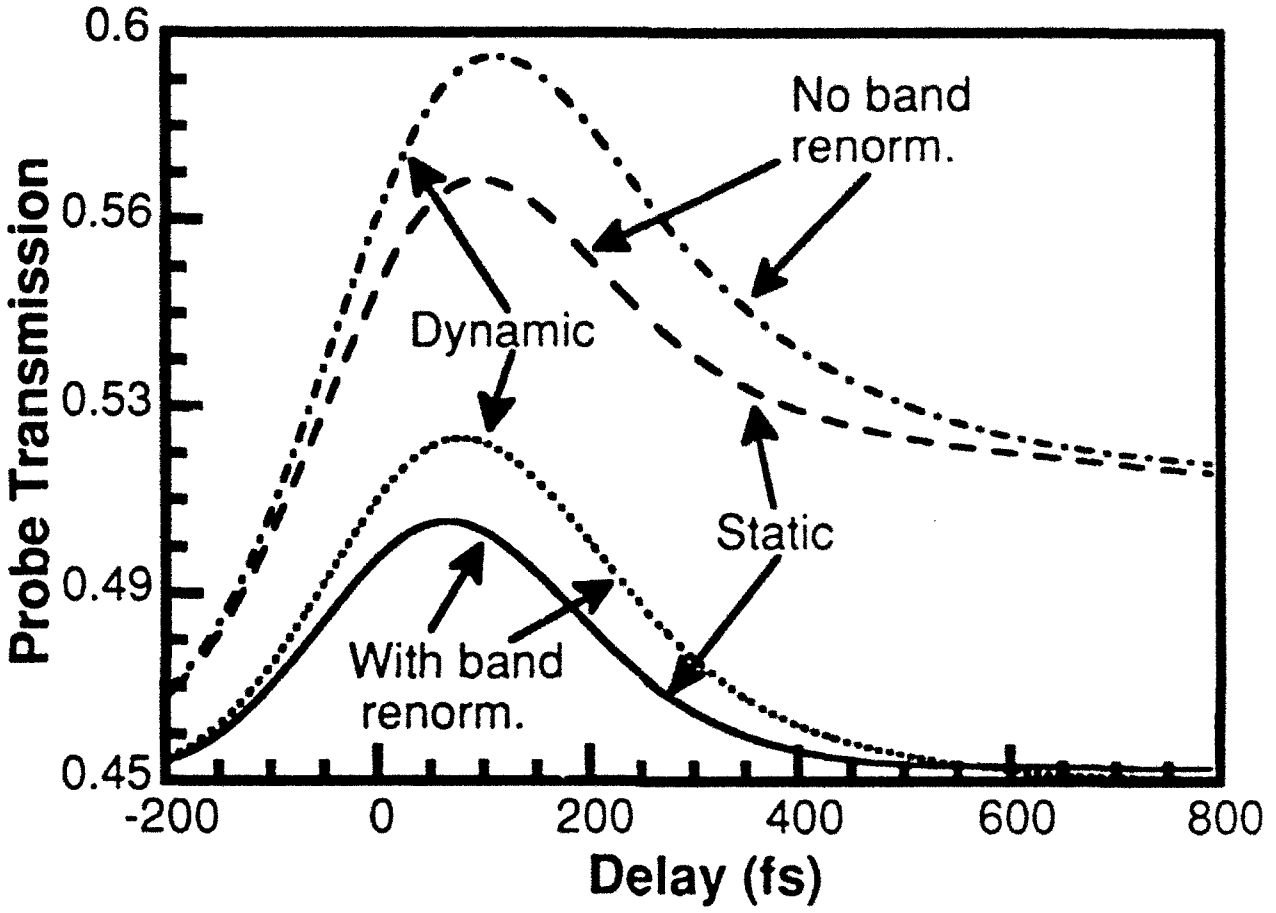
Fig. 2. Simulated optical transmission as a function of the delay between excitation for pulse-probe configuration. Curves with/without static and dynamic screening and band renormalization have been labeled accordingly.

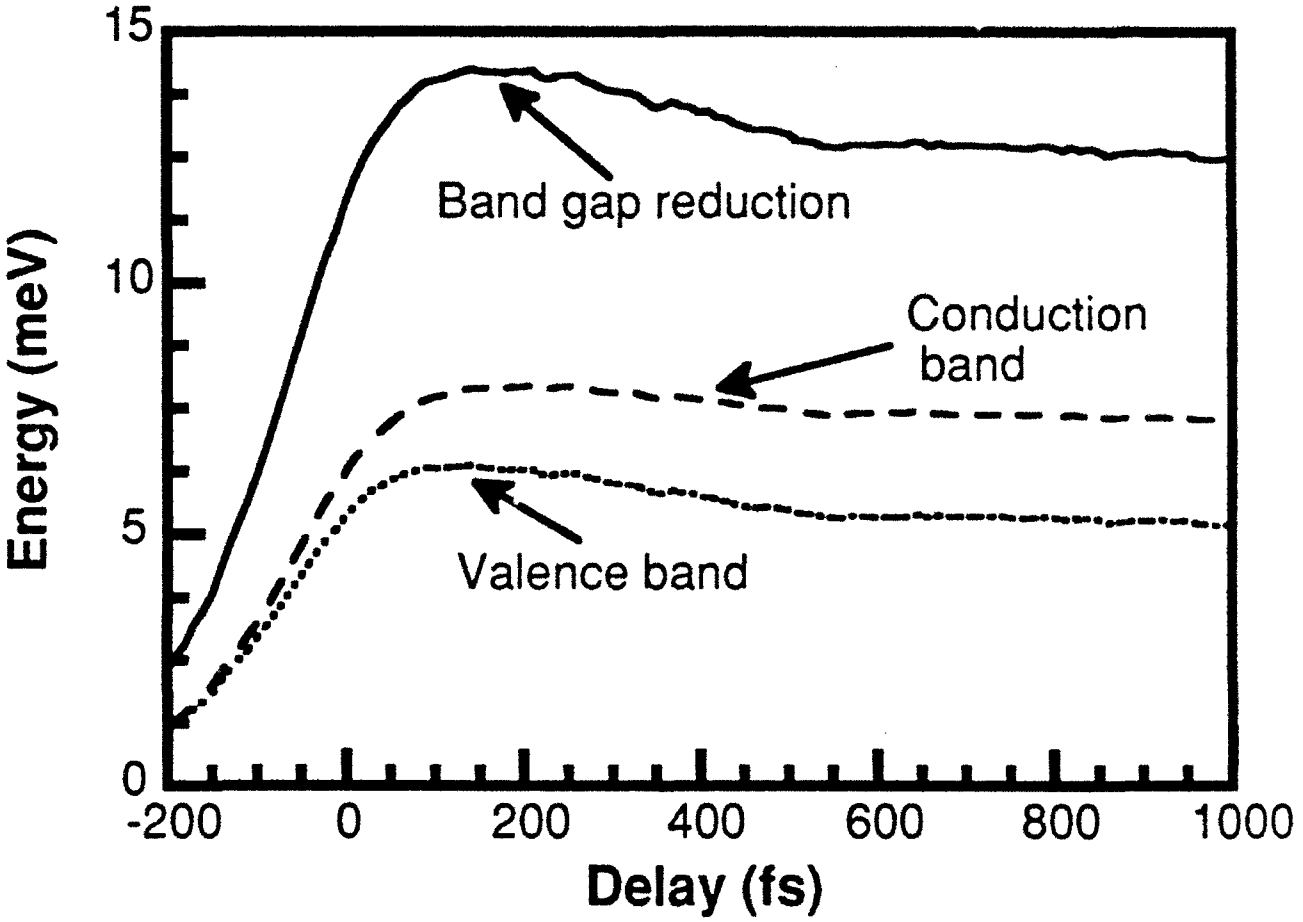
Fig. 3. Simulated rigid band shifts due to band renormalization as a function of probe delay.

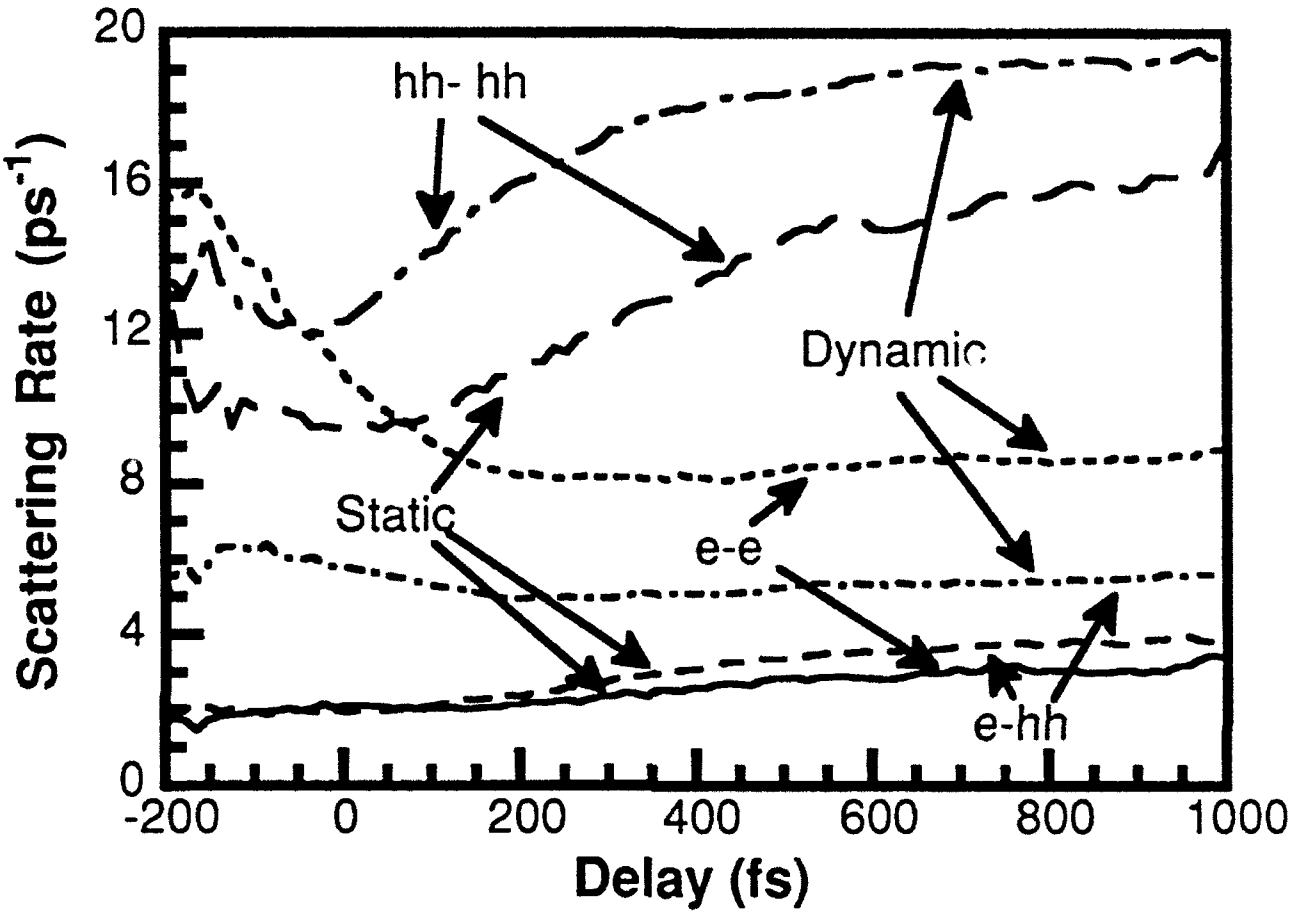
Fig. 4. Simulated carrier-carrier scattering rates as a function of probe delay. hh-hh, e-e, and e-hh denote heavy hole - heavy hole, electron - electron, electron - heavy hole scattering respectively. Static and dynamic refer to static and dynamic screening.

Table I. Effective relaxation times calculated by fitting a single exponential to the measured and simulated tranmission over the range probe delay of 200-800 fs .


Normalized Transmission
1
0.8
0.6
0.4
0.2
0
-0.2
0
200
400
600
800
1000
Delay (fs)
Measured
equal-pulse
Simulated
equal-pulse
Simulated
Pulse-probe


0.6
0.56
0.53
0.49
0.45
Probe Transmission
No band
renorm.
Dynamic
Static
With band
renorm.
-200
0
200
400
600
800
Delay (fs)


15
10
5
0
Energy (meV)
Band gap reduction
Conduction band
Valence band
-200
0
200
400
600
800
1000
Delay (fs)



Curve	Relaxation Time (fs)
Exp. Equal-Pulse	116
Sim. Equal-Pulse	106
Sim. Pulse-Probe	100
Static Screening	139
No Renorm.	169
Static-No renorm.	199

TASK 5 PARALLEL STRUCTURES FOR REAL-TIME ADAPTIVE SIGNAL PROCESSING

A. W. Bojanczyk

Row Householder Transformations for Rank-k Cholesky Inverse Modifications

Adam W. Bojanczyk,* James G. Nagy,† and Robert J. Plemmons ‡

March 4, 1993

Abstract

Householder transformations applied from the left are generally used to zero a contiguous sequence of entries in a column of a matrix A. Our purpose in this paper is to introduce new *row Householder* and *row hyperbolic Householder* transformations which are also applied from the left, but now zero a contiguous sequence of entries in a row of A. We then show how these row Householder transformations can be used to design efficient sliding data window recursive least squares covariance algorithms, which are based upon rank-k modifications to the inverse Cholesky factor, R^{-1}, of the covariance matrix. The algorithms are rich in matrix-matrix BLAS-3 computations, making them efficient on vector and parallel architectures. Preliminary numerical experiments are reported, comparing these row Householder-based rank-k modification schemes with k applications of the classical updating and downdating covariance schemes which use Givens and hyperbolic rotations.

Abbreviated Title. Row Householder Transformations

Key Words. (Row) Householder transformations, Cholesky updating and downdating, recursive least squares, BLAS-3 computations.

AMS(MOS) Subject Classifications. 15A12, 65F10, 65F20, 65F35

*School of Electrical Engineering, Cornell University, E&TC Building, Ithaca, NY 14850-3801. Research supported in part by the Joint Services Electronics Program, contract no. F49620-90-C-0039.

†Institute for Mathematics and its Applications, University of Minnesota, Minneapolis, MN 55455. Permanent address: Department of Mathematics, Southern Methodist University, Dallas, TX 75275-0461.

‡Department of Mathematics and Computer Science, Wake Forest University, P.O. Box 7388, Winston-Salem, NC 27109. Research supported in part by the US Air Force under grant AFOSR-91-0163.

1 Introduction

In this paper we introduce new *row Householder* and *row hyperbolic Householder* transformations, which zero one *row* of a matrix at a time when applied from the left. These transformations are a generalization of an idea first proposed by Bartels and Kaufman [3] and, as in classical Householder transformations, are rank-1 modifications to the identity matrix. We will discuss their use in developing efficient algorithms for recursive least squares problems of the sliding window type.

In [3], Bartels and Kaufman consider schemes for modifying R, where $X = QR$ and X is the given data matrix, subject to rank-2 updates of X. To solve these problems efficiently, they introduce a modified Householder transformation which, when applied from the left, can zero entries simultaneously in two column vectors. Here we suggest a generalization to this transformation which, when applied from the left, can eliminate all elements in a row of a matrix. We then illustrate how these transformations can be very useful in developing efficient algorithms for modifying R^{-1} (rather than R) subject to rank-k changes in X. (Algorithms for modifying R subject to rank-k changes in X were considered in [17] and analyzed in [6]). We show, in terms of operations counts, that our algorithms are more efficient for modifying R^{-1} than k applications of the classical algorithms based on Givens and hyperbolic rotations (see, for example, Pan and Plemmons [14].) Moreover, as Bartels and Kaufman show for rank-2 modifications, our algorithms are rich in matrix-matrix BLAS-3 computations, making them even more economical on high performance architectures than k applications of the rank-1 modification schemes.

The outline of this paper is as follows. In Section 1 we introduce the new row Householder transformations. In Section 2 we show how these transformations can be used to efficiently update least squares solutions when observations are added and/or deleted from the linear system. In Section 4 we consider downdating computations. In Section 5 we discuss compact WY representation of products of row Householder transformations, and in Section 6 we provide some numerical experiments and some concluding remarks.

2 Row Householder Transformations

In this section we introduce a row Householder transformation, which is a rank 1 modification to the identity matrix, that when applied from the left will eliminate k elements in a row of a matrix at once. These row Householder transformations are still reflections. As pointed out to the authors by R. Funderlic upon reading a preliminary version of the manuscript, row Householder reflections can be interpreted geometrically in the following way. Given two three dimensional vectors in three space, what one is doing is finding a reflection that takes the plane determined by the two vectors into the $y - z$ plane. Moreover, it appears that Householder reflections of the type described in this paper can be used to eliminate contiguous sequences of elements in different rows by applying a single reflection from the left. That possibility is not considered in this paper, but is a topic of future investigation [7].

We will split our discussion into two subsections. The first will consider row Householder transformations which are orthogonal, and the second subsection will consider transformations which are pseudo orthogonal with respect to a signature matrix Φ.

2.1 Orthogonal Row Householder Transformations

The row Householder transformation we introduce in this section is a generalization of an idea first proposed by Bartels and Kaufman [3]. Let B be a $(k+1) \times k$ matrix of the form

$$B = \begin{bmatrix} b^T \\ D \end{bmatrix},$$

where D is nonsingular.

Suppose we wish to eliminate the first row of B (*i.e.*, b^T) by premultiplying by an orthogonal matrix. (Note that this discussion applies, in general, to the case where we want to eliminate the j^{th} row of B. In this case we simply permute the j^{th} row to the top of B.) In order to accomplish this we construct a Householder transformation

$$P = I - \frac{1}{\lambda} pp^T, \tag{1}$$

where $p \in \Re^{k+1}$ and $\lambda = p^T p/2$, such that

$$PB = \begin{bmatrix} 0^T \\ \tilde{D} \end{bmatrix}. \tag{2}$$

In order to illustrate how this can be done let

$$p = \begin{bmatrix} \pi \\ q \end{bmatrix}$$

where π is the first component of p and q is the vector consisting of the last k components of p.

If P has the form (1) and satisfies (2), then we obtain the relation

$$\begin{bmatrix} b^T \\ D \end{bmatrix} - \frac{1}{\lambda} p(\pi b^T + q^T D) = \begin{bmatrix} 0 \\ \tilde{D} \end{bmatrix}. \tag{3}$$

From the first row of the relation (3) we obtain

$$D^T q = \mu b, \tag{4}$$

where

$$\mu = (\lambda/\pi - \pi). \tag{5}$$

The relation (5) together with $\lambda = p^T p/2$ gives

$$\pi = -\mu \pm \sqrt{\mu^2 + q^T q}. \tag{6}$$

In order to avoid loss of accuracy in computer finite precision arithmetic we pick the sign so to maximize the magnitude of π. Then π can be expressed as follows

$$\pi = -\mu(1 + \sqrt{1 + z^T z}) \tag{7}$$

where $z = D^{-T}b$. We note that we have one degree of freedom here. Since μ is a free variable, we suggest choosing $\mu = 1/\|b\|_2$. If $\|b\|_2 = 0$, we simply set $P = I$.

In general, we have the following algorithm.

Algorithm ROWHT

Input: $B^T = [b \;\; D^T]$, where $D \in \Re^{k\times k}$ is nonsingular.

Output: $p \in \Re^{k+1}$, where $P = I - \frac{1}{\lambda}pp^T$, $\lambda = p^T p/2$, has the property that the first row of PB is all zeros.

if $\|b\|_2 = 0$
  $p = 0$, $P = I$
else
  $\mu = 1/\|b\|_2$
  solve $D^T q = \mu b$
  $\pi = -\mu - \sqrt{\mu^2 + q^T q}$
  $p^T = [\pi \;\; q^T]$

A Householder transformation (computed by algorithm ROWHT) which zeros elements in a row vector will be called a *row* Householder transformation to differentiate it from the classical *column* Householder transformation which zeros elements in a column vector.

The algorithm ROWHT will have good numerical properties as long as (4) is solved by a numerically stable method. This is made precise by the following lemma.

Lemma 1 *Let ϵ be the machine relative precision and $\bar{q}$ satisfies*

$$(D^T + \delta\bar{D}^T)\bar{q} = \mu b \tag{8}$$

with $\|\delta\bar{D}^T\| = O(\epsilon\|D\|)$. Further, let $\bar{\pi} = -\mu - \sqrt{\mu^2 + \bar{q}^T\bar{q}}$, $\bar{p}^T = [\bar{\pi} \;\; \bar{q}^T]$, $\lambda = \bar{q}^T\bar{q}/2$ and

$$\bar{P} = I - \frac{1}{\lambda}\bar{p}\bar{p}^T .$$

Then there exists a perturbation δB of the matrix B such that

$$\bar{P}(B + \delta B) = \begin{bmatrix} 0 \\ \bar{D} \end{bmatrix} \tag{9}$$

and $\|\delta B\| = O(\epsilon\|B\|)$.

Proof: The proof is straightforward and hence omitted.

REMARK: It is important to note that $\bar{P}$ does not have to be close to P defined by (1) and (2). Similarly, $\bar{D}$ does not have to be close to $\tilde{D}$ in (2). The situation here is analogous to that of the QR decomposition of a pertubed matrix $X+\delta X$ where the factors of the perturbed matrix can differ from the factors of the original matrix X by as much as $O(cond(X)||\delta X||)$, see Stewart [19]. However what is essential from the numerical analysis point of view is that $\bar{P}$ is orthogonal and zeros the first row of a nearby matrix $B+\delta B$.

Note that finding p requires solving a $k \times k$ system of linear equations which in general amounts to $O(k^3)$ operations. However, if the QR decomposition of D is available, the cost of finding p is decreased to $O(k^2)$ operations.

In the sequel we will encounter the problem of annihilating r rows, $r \geq 1$, of a $(k+r) \times k$ matrix by finding an orthogonal P such that

$$P \begin{bmatrix} b_1^T \\ \vdots \\ b_r^T \\ D \end{bmatrix} = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \tilde{D} \end{bmatrix}$$

Such a P can be constructed as a product of r row Householder transformations P_i, $P = P_r P_{r-1} \cdots P_1$, with P_i annihilating row i of the matrix. As the major cost of determining such transformations is in solving systems of linear equations, it is worthwhile to attempt to decrease this cost. This can be done by maintaining and updating the QR decomposition of the bottom $k \times k$ submatrix. For the sake of illustration we show the first step of this process. Let

$$B = \begin{bmatrix} b_1^T \\ \vdots \\ b_r^T \\ D \end{bmatrix}$$

and let

$$D_0 \equiv D = Q_0 R_0$$

be the QR decomposition of D which is assumed to be given. Let $P_1 = I - p_1 p_1^T/\lambda_1$ be a modified Householder transformation such that

$$P_1 \begin{bmatrix} b_1^T \\ b_2^T \\ \vdots \\ b_r^T \\ D \end{bmatrix} = \begin{bmatrix} 0 \\ b_2^T \\ \vdots \\ b_r^T \\ D_1 \end{bmatrix}.$$

Then the form of P_1 implies that

$$D_1 = D_0 - \frac{1}{\lambda_1} p_1 b_1^T = Q_0 R_0 - \frac{1}{\lambda_1} p_1 b_1^T \tag{10}$$

Thus the QR factorization of D_1 can be obtained at the cost of $13k^2$ multiplications (and $13k^2$ additions) by updating the QR factorization of D_0 after a rank-1 change of D_0, [11]. This has to be repeated $r-1$ times resulting in the total of $O(rk^2)$ operations for the overall process of computing all transformations P_i, $i = 1, \cdots, r$.

2.2 Row Hyperbolic Householder Transformations

Let Φ =diag(± 1) be an $(k+1) \times (k+1)$ diagonal matrix, and suppose p is a vector of length $k+1$ with $p^T \Phi p > 0$. Then a Hyperbolic Householder transformation is a matrix of the form

$$P = \Phi - \frac{1}{\lambda} p p^T \tag{11}$$

where $\lambda = \frac{1}{2} p^T \Phi p$. The matrix P is a *pseudo orthogonal* matrix with respect to Φ, *i.e.*,

$$P^T \Phi P = \Phi.$$

Hyperbolic Householder transformations are typically used to introduce zeros into a column of a matrix, and were studied in detail by Rader and Steinhardt [17]. Here we introduce a *row Hyperbolic Householder* transformation which eliminates entries in a row of a matrix. The discussion in this subsection is similar to that given in §2.1 for the (orthogonal) row Householder transformations.

Let B be a $(k+1) \times k$ matrix of the form

$$B = \begin{bmatrix} b^T \\ D \end{bmatrix}$$

where D is nonsingular. Suppose we wish to eliminate the first row of B using a transformation of the form (11). As in §2.1 this can be illustrated as follows. Let

$$p = \begin{bmatrix} \pi \\ q \end{bmatrix}$$

where π is the first component of p and q is a vector consisting of the last k components of p. Now suppose

$$P \begin{bmatrix} b^T \\ D \end{bmatrix} = \begin{bmatrix} 0^T \\ \tilde{D} \end{bmatrix}.$$

Then, assuming P has the form (11), where

$$\Phi = \begin{bmatrix} \phi_1 & 0 \\ 0 & \tilde{\Phi} \end{bmatrix},$$

we have

$$\begin{bmatrix} \phi_1 & 0 \\ 0 & \tilde{\Phi} \end{bmatrix} \begin{bmatrix} b^T \\ D \end{bmatrix} - \frac{1}{\lambda} \begin{bmatrix} \pi \\ q \end{bmatrix} (\pi b^T + q^T D) = \begin{bmatrix} 0^T \\ \tilde{D} \end{bmatrix}.$$

Thus, we obtain

$$D^T q = \mu b \tag{12}$$

where $\mu = (\lambda\phi_1/\pi - \pi)$.

Now, if we fix μ, then we can solve (12) for q. Once q is known, then, using $\mu = (\lambda\phi_1/\pi - \pi)$, we have

$$\pi^2 + \pi\mu - \phi_1\lambda = 0.$$

Thus, since

$$\lambda = \frac{1}{2}p^T\Phi p = \frac{1}{2}(\phi_1 \quad + q^T\tilde{\Phi}q),$$

and since $\phi_1^2 = 1$, we obtain the relation

$$\pi^2 + 2\pi\mu - \phi_1 q^T\tilde{\Phi}q = 0.$$

Thus, if

$$\mu^2 + \phi_1 q^T\tilde{\Phi}q \geq 0, \tag{13}$$

we have

$$\pi = -\mu - \mathrm{sgn}(\mu)\sqrt{\mu^2 + \phi_1 q^T\tilde{\Phi}q}.$$

We point out that the requirement $\mu^2 + \phi_1 q^T\tilde{\Phi}q \geq 0$ is satisfied for our problem of inverse matrix modifications. This will be discussed in further detail in Section 4.

As for the (orthogonal) row Householder transformations, we suggest choosing $\mu = 1/\|b\|_2$, and $P = \Phi$ if $\|b\|_2 = 0$. The following algorithm summarizes the above discussion.

Algorithm ROWHHT

Input: $B^T = [b \;\; D^T]$, where $D \in \Re^{k\times k}$ is nonsingular.

Output: $p \in \Re^{k+1}$, where $P = \Phi - \frac{1}{\lambda}pp^T$, $\lambda = p^T\Phi p/2$, has the property that the first row of PB is all zeros.

```
if ||b||_2 = 0
    p = 0, P = Φ
else
    μ = 1/||b||_2
    solve D^T q = μb
    π = −μ − sqrt(μ^2 + φ_1 q^T Φ̃ q)
    p^T = [π  q^T]
```

Similarly as for the orthogonal case, a hyperbolic Householder transformation (computed by algorithm ROWHHT) which zeros elements in a row vector will be called a *row* hyperbolic Householder transformation. If the QR decomposition of D is available the cost of finding p is of the order of $O(k^2)$ operations. For the problem of annihilating r rows, $r \geq 1$, of a

$(k+r) \times k$ matrix B that cost is of the order of $O(rk^2)$ operations (see the discussion at the end of Section 2.1).

3 Modifying the Inverse Cholesky Factor

Let X be a real $m \times n$ matrix with full column rank, and let s be a real vector of length m. Consider the least squares problem

$$\min \|s - Xw\|_2. \tag{14}$$

It is well known (see, for instance [12]) that this problem can be solved by finding the QR factorization of X. Specifically, let $X = QR$, where Q is an $m \times n$ matrix with orthonormal columns, and R is an $n \times n$ upper triangular matrix. Then the solution to (14) is given by

$$w = R^{-1}Q^T s.$$

In many applications, such as signal processing, it is often required to recalculate w when successive observations (*i.e.*, equations) are added to and/or deleted from (14). In this section we consider *updating* the solution w to $\hat{w}$ when k new observations are added to the system, and *downdating* w to $\check{w}$ when k observations are removed from the system. This method is called *recursive least squares* (RLS), and can be reformulated as a k-step process of k successive modifications of w after addition/deletion of a single observation. Such rank-1 modifications are most often realized by plane rotations and have been studied by many authors. In this paper we treat multiple addition/deletion of observation as a block process in a manner analogous to that presented in [17]. However, unlike in [17] where the the upper triangular factor in the QR decomposition of X was modified, this paper proposes algorithms for direct modification of the inverse of the triangular factor. This procedure is called the covariance method in RLS computations. We will show how the row Householder transformations described in Section 2 can be used to design efficient sliding data window RLS covariance algorithms.

3.1 Inverse Updating

Let $X = QR$ be the QR factorization of X. Suppose k new observations

$$\begin{bmatrix} Y^T & u \end{bmatrix},$$

where $Y^T \in \Re^{k \times n}$ and $u \in \Re^k$, are added to the data defining the least squares problem (14). We first show how R^{-1} can be updated to $\hat{R}^{-1}$, where

$$\hat{X} = \begin{bmatrix} X \\ Y^T \end{bmatrix} = \hat{Q}\hat{R}$$

is the QR factorization of $\hat{X}$. We then show how the solution w of (14) can be updated to the solution $\hat{w}$ of

$$\min \left\| \begin{bmatrix} s \\ u \end{bmatrix} - \begin{bmatrix} X \\ Y^T \end{bmatrix} \hat{w} \right\|_2 . \tag{15}$$

It is well known that there exists an orthogonal matrix H such that

$$H \begin{bmatrix} R \\ Y^T \end{bmatrix} = \begin{bmatrix} \hat{R} \\ 0^T \end{bmatrix} . \tag{16}$$

The matrix H can be constructed as a product of $(n+k) \times (n+k)$ Householder transformations H_i, $i = 1, \cdots, n$, such that H_i annihilates subdiagonal elements in column i, $i = 1, \cdots, n$, of the matrix

$$H_{i-1} \cdots H_2 H_1 \begin{bmatrix} R \\ Y^T \end{bmatrix} .$$

It is known that if H is orthogonal and satisfies (16), then H also updates the inverse of R, namely

$$H \begin{bmatrix} R^{-T} \\ 0^T \end{bmatrix} = \begin{bmatrix} \hat{R}^{-T} \\ E^T \end{bmatrix} , \tag{17}$$

where E is an $n \times k$ matrix. To see this, note that

$$I = \begin{bmatrix} R^{-1} & 0 \end{bmatrix} \begin{bmatrix} R \\ Y^T \end{bmatrix} = \begin{bmatrix} R^{-1} & 0 \end{bmatrix} H^T H \begin{bmatrix} R \\ Y^T \end{bmatrix} = \begin{bmatrix} U & E \end{bmatrix} \begin{bmatrix} \hat{R} \\ 0^T \end{bmatrix} .$$

Thus $U = \hat{R}^{-1}$.

We would like to be able to work with R^{-T}, and not with R explicitly, since the triangular solves needed in solving systems associated with R can then be replaced by matrix-vector or matrix-matrix multiplications. The following lemma shows how we can construct an orthogonal matrix H satisfying (16) and avoid using R explicitly.

Lemma 2 *Let $\hat{V} = -R^{-T}Y$, and let $\hat{H}$ be an orthogonal matrix such that*

$$\hat{H} \begin{bmatrix} \hat{V} \\ I_k \end{bmatrix} = \begin{bmatrix} 0 \\ \hat{D} \end{bmatrix} , \tag{18}$$

where I_k is the $k \times k$ identity matrix and $\hat{D}$ is a $k \times k$ matrix. Then

$$\hat{H} \begin{bmatrix} R \\ Y^T \end{bmatrix} = \begin{bmatrix} U \\ 0 \end{bmatrix} \tag{19}$$

If U is upper triangular then $U = \hat{R}$ and

$$\hat{H} \begin{bmatrix} R^{-T} \\ 0^T \end{bmatrix} = \begin{bmatrix} \hat{R}^{-T} \\ E^T \end{bmatrix} . \tag{20}$$

where $E = R^{-1}\hat{V}\hat{D}^{-1}$.

Proof: The proof for $k = 1$ can be found in [14]. For $k > 1$ one proceeds as follows. Let

$$\hat{H} \begin{bmatrix} \hat{V} & R \\ I_k & Y^T \end{bmatrix} = \begin{bmatrix} 0 & U \\ \hat{D} & \hat{Y}^T \end{bmatrix} . \tag{21}$$

From the orthogonality of $\hat{H}$, the definition of $\hat{V}$ and the fact that $\hat{D}$ is nonsingular, it follows that $\hat{Y} = 0$ and hence

$$R^T R + Y Y^T = U^T U .$$

Thus if U is upper triangular with positive diagonal elements then $U = \hat{R}$. From (17), for the inverse we have an analogous relation, namely

$$\hat{H} \begin{bmatrix} \hat{V} & R^{-T} \\ I_k & 0 \end{bmatrix} = \begin{bmatrix} 0 & \hat{R}^{-T} \\ \hat{D} & E^T \end{bmatrix} . \tag{22}$$

Now (22) implies

$$\begin{bmatrix} \hat{V}^T V + I & \hat{V}^T R^{-T} \\ R^{-1} \hat{V} & R^{-1} R^{-T} \end{bmatrix} = \begin{bmatrix} \hat{D}^T \hat{D} & \hat{D}^T E^T \\ E \hat{D} & \hat{R}^{-1} \hat{R}^{-T} + E E^T \end{bmatrix}$$

from which one obtains that

$$E = R^{-1} \hat{V} \hat{D}^{-1} .$$

This completes the proof. □

The relation (22) shows that it is possible to work with the inverses only. The condition that has to be satisfied is that application of the transformation $\hat{H}$ in (22) has to result in a lower triangular matrix U^{-T}.

We now show how to construct an orthogonal matrix $\hat{H}$ satisfying (18) and (19). To do this, we will use the row Householder transformation. More precisely, suppose that we have constructed row Householder transformations $P_1, P_2, \ldots, P_j$ such that

$$P_j \cdots P_2 P_1 \begin{bmatrix} \hat{V} \\ I \end{bmatrix} = \begin{bmatrix} 0_j \\ \hat{V}_j \\ \hat{D}_j \end{bmatrix} ,$$

where 0_j denotes the $j \times k$ matrix of all zeros, and $\hat{V}_j \in \Re^{(n-j)\times k}$ and $\hat{D}_j \in \Re^{k\times k}$. Then using Algorithm ROWHT, we find $\hat{p}_j^T = [\pi_j \;\; q_j]$ so that

$$\hat{P}_{j+1} \begin{bmatrix} \hat{v}_j^T \\ \hat{D}_j \end{bmatrix} = \begin{bmatrix} 0^T \\ \hat{D}_{j+1} \end{bmatrix} ,$$

where $\hat{v}_j^T$ is the first row of $\hat{V}_j$ and $\hat{P}_{j+1} = I - \frac{1}{\lambda_j} \hat{p}_j \hat{p}_j^T$. Then P_{j+1} is simply given by

$$P_{j+1} = I - \frac{1}{\lambda_j} p_j p_j^T ,$$

where $p_j = [0, \cdots, 0, \pi_j, 0, \cdots, 0, q_j]$ (the j-th component of p_j is π_j, the last k components of p_j form the vector q_j, and all other components are zeros). It is now easy to see that $P = P_n \cdots P_2 P_1$ satisfies (18) and hence

$$P_n \cdots P_2 P_1 \begin{bmatrix} R^{-T} \\ 0 \end{bmatrix} = \begin{bmatrix} \hat{R}^{-T} \\ E^T \end{bmatrix},$$

as $\hat{R}^{-T}$ is by construction lower triangular and hence the desired downdated factor.

Now that we have a scheme for updating R^{-T}, we need to use this information to efficiently update the least squares solution w to $\hat{w}$. The following theorem shows how this can be done.

Theorem 1 *Let $\hat{H}$ satisfies (22), that is*

$$\hat{H} \begin{bmatrix} \hat{V} & R^{-T} \\ I_k & 0 \end{bmatrix} = \begin{bmatrix} 0 & \hat{R}^{-T} \\ \hat{D} & E^T \end{bmatrix}.$$

If w is the solution to (??), then the solution to (15) is given by

$$\hat{w} = w - E\hat{D}^{-T}(u - Y^T w)$$

Moreover $E = R^{-1}\hat{V}\hat{D}^{-1}$.

Proof: Let

$$X = Q \begin{bmatrix} R \\ 0 \end{bmatrix}, \quad Q^T s = \begin{bmatrix} s_1 \\ s_2 \end{bmatrix}$$

where $s_1 \in \Re^n$ and $s_2 \in \Re^k$. Then (15) can be rewritten as

$$\min \left\| \begin{bmatrix} s \\ u \end{bmatrix} - \begin{bmatrix} X \\ Y^T \end{bmatrix} \hat{w} \right\|_2 = \min \left\| \begin{bmatrix} s_1 \\ s_2 \\ u \end{bmatrix} - \begin{bmatrix} R \\ 0 \\ Y^T \end{bmatrix} \hat{w} \right\|_2. \tag{23}$$

Furthermore, let

$$\hat{H} \begin{bmatrix} s_1 \\ u \end{bmatrix} = \begin{bmatrix} \hat{s}_1 \\ \hat{u} \end{bmatrix}. \tag{24}$$

Note that $w = R^{-1}s_1$ and hence from the definition of $\hat{V}$ we have the relation

$$\begin{bmatrix} u - Y^T w \\ w \end{bmatrix} = \begin{bmatrix} \hat{V}^T s_1 + u \\ R^{-1} s_1 \end{bmatrix} = \begin{bmatrix} \hat{V}^T & I \\ R^{-1} & 0 \end{bmatrix} \begin{bmatrix} s_1 \\ u \end{bmatrix}. \tag{25}$$

Using the definition (24) and the assumption of the theorem the right hand side of (25) simplifies as follows

$$\begin{bmatrix} \hat{V}^T & I \\ R^{-1} & 0 \end{bmatrix} \begin{bmatrix} s_1 \\ u \end{bmatrix} = \begin{bmatrix} 0 & \hat{D}^T \\ \hat{R}^{-1} & E \end{bmatrix} \begin{bmatrix} \hat{s}_1 \\ \hat{u} \end{bmatrix} = \begin{bmatrix} \hat{D}^T \hat{u} \\ \hat{R}^{-1}\hat{s}_1 + E\hat{u} \end{bmatrix} = \begin{bmatrix} \hat{D}^T \hat{u} \\ \hat{w} + E\hat{u} \end{bmatrix}, \tag{26}$$

as from (23) we have that $\hat{w} = \hat{R}^{-1}s_1$. Now the theorem follows from the fact that the left hand side of (25) and the right hand side of (26) are equal.

□

Thus, summarizing the results of this section, we obtain the following algorithm.

Algorithm IUP-k
Given: R^{-T} and w, where $X = QR$ and w solves (??).
Input: New set of k observations $[Y^T \ u]$.
Then this algorithm computes $\hat{R}^{-T}$ and $\hat{w}$, where

$$\hat{X} = \begin{bmatrix} X \\ Y^T \end{bmatrix} = \hat{Q}\hat{R}$$

and $\hat{w}$ solves (15).

1. Compute $\hat{V} = -R^{-T}Y$.

   Cost $kn^2/2$ multiplications.

2. Find $\hat{H} = P_n \cdots P_2 P_1$, where P_i are row Householder Transformations, such that

$$\hat{H}\begin{bmatrix} \hat{V} \\ I \end{bmatrix} = \begin{bmatrix} 0 \\ \hat{D} \end{bmatrix}$$

   Cost $13 \cdot k^2 n$ multiplications.

3. Update R^{-T} to $\hat{R}^{-T}$:

$$\hat{H}\begin{bmatrix} R^{-T} \\ 0 \end{bmatrix} = \begin{bmatrix} \hat{R}^{-T} \\ E^T \end{bmatrix}$$

   Cost kn^2 multiplications.

4. Update w to $\hat{w}$:

$$\hat{w} = w - E\hat{D}^{-T}(u - Y^T w)$$

   Cost $\frac{3}{2}k^2 + 2kn$ multiplications (as from (10) the QR decomposition of D is already available from step 2).

The total cost for Algorithm IUP-k is $\frac{3}{2} \cdot kn^2 + 13 \cdot k^2 n + 2 \cdot kn + \frac{3}{2}k^2$ multiplications. We note that the straight forward implementation of the rank-1 method of Pan and Plemmons [14] would require $\frac{5}{2}kn^2 + O(kn)$ multiplications. Thus, roughly speaking speaking

Algorithm IUP-k requires less multiplications than the method described in [14] when $n > 13k$. The major advantage of Algorithm IUP-k is that it is rich in BLAS level 2 and BLAS level 3 operations which may lead to a more efficient implementation on parallel computers.

3.2 Inverse Downdating

Let

$$s = \begin{bmatrix} \check{s} \\ d \end{bmatrix}, \quad X = \begin{bmatrix} \check{X} \\ Z^T \end{bmatrix}$$

where $Z^T \in \Re^{k \times n}$ and $d \in \Re^k$, are to be deleted from the data describing (14). We now describe a method for downdating w to the solution $\check{w}$ of

$$\min \|\check{s} - \check{X}\check{w}\|_2. \tag{27}$$

We first show how R^{-1} can be updated to $\check{R}^{-1}$, where

$$\check{X} = \check{Q}\check{R}$$

is the QR factorization of $\check{X}$.

Note that as long as $\check{X}$ is full rank then

$$R^T R - ZZ^T > 0\,. \tag{28}$$

The Cholesky factor $\check{R}$ of $\check{X}$ satisfes

$$\check{R}^T\check{R} = R^T R - ZZ^T\,.$$

In [17] it is shown that there exists a pseudo orthogonal transformation $\check{H}$ with respect to the signature Φ,

$$\Phi = \begin{bmatrix} I_n & 0 \\ 0 & -I_k \end{bmatrix},$$

such that

$$\check{H}\begin{bmatrix} R \\ Z^T \end{bmatrix} = \begin{bmatrix} \check{R} \\ 0^T \end{bmatrix}. \tag{29}$$

The matrix $\check{H}$ can be constructed as a product of $(n+k)\times(n+k)$ Hyperbolic Householder transformations H_i, $i = 1, \cdots, n$, such that H_i annihilates subdiagonal elements in column i, $i = 1, \cdots, n$, of the matrix

$$H_{i-1}\cdots H_2 H_1 \begin{bmatrix} R \\ Z^T \end{bmatrix}.$$

Similarly as for orthogonal transformations, if the hyperbolic $\check{H}$ satisfies (29) then $\check{H}$ also downdates the inverse of R. To see this, note that

$$I = \begin{bmatrix} R^{-1} & 0 \end{bmatrix} \Phi \begin{bmatrix} R \\ Z^T \end{bmatrix} = \begin{bmatrix} R^{-1} & 0 \end{bmatrix} \check{H}^T \Phi \check{H} \begin{bmatrix} R \\ Z^T \end{bmatrix} = \begin{bmatrix} U & F \end{bmatrix} \begin{bmatrix} \check{R} \\ 0^T \end{bmatrix}.$$

Thus $U = \check{R}^{-1}$, and

$$\check{H} \begin{bmatrix} R^{-T} \\ 0 \end{bmatrix} = \begin{bmatrix} \check{R}^{-T} \\ F^T \end{bmatrix}. \tag{30}$$

We would like to work with the inverses directly and hence need a way for constructing H satisfying (29) without any explicit reference to R. The following lemma provides means just for that.

Lemma 3 *Assume $R^TR - ZZ^T > 0$. Let $\check{V} = R^{-T}Z$, and let $\check{H}$ be a hyperbolic (with respect to Φ) transformation such that*

$$\check{H} \begin{bmatrix} \check{V} \\ I_k \end{bmatrix} = \begin{bmatrix} 0 \\ \check{D} \end{bmatrix}, \tag{31}$$

where I_k is the $k \times k$ identity matrix and $\check{D}$ is a $k \times k$ matrix. Then

$$\check{H} \begin{bmatrix} R \\ Z^T \end{bmatrix} = \begin{bmatrix} \check{U} \\ 0 \end{bmatrix} \tag{32}$$

If $\check{U}$ is upper triangular, then $\check{U} = \check{R}$ and

$$\check{H} \begin{bmatrix} R^{-T} \\ 0^T \end{bmatrix} = \begin{bmatrix} \check{R}^{-T} \\ F^T \end{bmatrix}. \tag{33}$$

where $F = -R^{-1}\check{V}\check{D}^{-1}$.

Proof: The proof for $k = 1$ can be found in [14]. For $k > 1$ one proceeds as follows. Let

$$\check{H} \begin{bmatrix} \check{V} & R \\ I_k & Z^T \end{bmatrix} = \begin{bmatrix} 0 & \check{U} \\ \check{D} & \check{Z}^T \end{bmatrix}. \tag{34}$$

From the definition of $\check{V}$ and the that fact that H is hyperbolic (with respect to Φ) we obtain that

$$\begin{bmatrix} \check{V}^T\check{V} - I_k & 0 \\ 0 & R^TR - ZZ^T \end{bmatrix} = \begin{bmatrix} -\check{D}^T\check{D} & -\check{D}^T\check{Z}^T \\ -\check{Z}\check{D} & \check{U}^T\check{U} - \check{Z}\check{Z}^T \end{bmatrix}. \tag{35}$$

Comparing upper left entries on both sides we get

$$-\check{D}^T\check{D} = \check{V}^T\check{V} - I_k = Z^TR^{-1}R^{-T}Z - I_k.$$

Now, as $R^TR - ZZ^T > 0$ then $I_k - Z^TR^{-1}R^{-T}Z > 0$ and hence $\check{D}$ is nonsingular.

From (35) and the nonsingularity of $\check{D}$ it follows that $\check{Z} = 0$ and hence

$$R^T R - Z Z^T = \check{U}^T \check{U} .$$

Thus if $\check{U}$ is upper triangular (with positive diagonal elements) then $U = \check{R}$. From (30), for the inverse we have an analogous relation, namely

$$\check{H} \begin{bmatrix} \check{V} & R^{-T} \\ I_k & 0 \end{bmatrix} = \begin{bmatrix} 0 & \check{R}^{-T} \\ \check{D} & F^T \end{bmatrix} . \tag{36}$$

Now (36) implies

$$\begin{bmatrix} \check{V}^T \check{V} - I_k & \check{V}^T R^{-T} \\ R^{-1} \check{V} & R^{-1} R^{-T} \end{bmatrix} = \begin{bmatrix} -\check{D}^T \check{D} & -\check{D}^T F^T \\ -F \check{D} & \check{R}^{-1} \check{R}^{-T} - F F^T \end{bmatrix}$$

from which one obtains that

$$F = -R^{-1} \check{V} \check{D}^{-1} .$$

This completes the proof.

□

The relation (36) shows that, as for updating the inverse, it is also possible to downdate the inverse directly. The condition that has to be satisfied is that application of $\check{H}$ in (36) has to result in a lower triangular matrix U^{-T}.

The construction of $\check{H}$ satisfying (36) is analogous to that described at the end of Section 3.2. Now however $\check{H}$ is constructed as a product of row hyperbolic Householder transformations. The only thing that needs to be verified is that the condition (13) is always satisfied for each factor that makes up $\check{H}$.

Suppose that we have constructed row hyperbolic (with respect to Φ) Householder transformations $P_1, P_2, \ldots, P_j$ such that

$$P_j \cdots P_2 P_1 \begin{bmatrix} \check{V} \\ I \end{bmatrix} = \begin{bmatrix} 0_j \\ \check{V}_j \\ \check{D}_j \end{bmatrix} ,$$

where 0_j denotes the $j \times k$ matrix of all zeros, $\check{V}_j \in \Re^{(n-j)\times k}$ and $\check{D}_j \in \Re^{k \times k}$. Let $\check{v}_j^T$ be the first row of $\check{V}_j$ and let

$$\check{\Phi} = \begin{bmatrix} 1 & 0 \\ 0 & -I_k \end{bmatrix} .$$

We wish to use Algorithm ROWHHT to find $\check{p}_j^T = [\check{\pi}_j \;\; \check{q}_j]$ so

$$\check{P}_{j+1} = \check{\Phi} - \frac{1}{\check{\lambda}_j} \check{p}_j \check{p}_j^T ,$$

satisfies

$$\check{P}_{j+1} \begin{bmatrix} \check{v}_j^T \\ \check{D}_j \end{bmatrix} = \begin{bmatrix} 0^T \\ \check{D}_{j+1} \end{bmatrix} .$$

Note first that $\check{D}_j$ is nonsingular. The condition (13) for P_j becomes

$$\check{\mu}_j^2 - \check{q}_j^T \check{q}_j > 0 \tag{37}$$

where from (12) $\check{q}_j$ is given by

$$\check{q}_j = \check{\mu}_j \check{D}_j^{-T} \check{v}_j \, . \tag{38}$$

Substituting (38) in to (37) we obtain

$$\check{\mu}_j^2 (1 - \check{v}_j^T \check{D}_j^{-1} \check{D}_j^{-T} \check{v}_j) > 0 \tag{39}$$

Note however that from

$$\check{D}_j^T \check{D}_j - \check{v}_j \check{v}_j^T > 0 \tag{40}$$

(which is satisfied because $\check{D}_j^T \check{D}_j - \check{V}_j^T \check{V}_j > 0$) it follows that

$$1 - \check{v}_j^T \check{D}_j^{-1} \check{D}_j^{-T} \check{v}_j > 0 \, ,$$

which shows that (13) is satisfied.

Now, the construction of $\check{H}$ proceeds in a straightforward manner, exactly as in the (orthogonal) updating case.

The scheme for downdating R^{-T} can be extended to downdating the least squares solution w to $\check{w}$. The following theorem shows how this can be done.

Theorem 2 *Let $\check{H}$ satisfy (31), that is*

$$\check{H} \begin{bmatrix} \check{V} & R^{-T} \\ I_k & 0 \end{bmatrix} = \begin{bmatrix} 0 & \check{R}^{-T} \\ \check{D} & F^T \end{bmatrix} .$$

If w is the solution to (??), then the solution to (27) is given by

$$\check{w} = w + F \check{D}^{-T} (d - Z^T w)$$

Moreover $F = -R^{-1} \check{V} \check{D}^{-1}$.

Proof: The proof is analogous to that of Theorem 1 and hence is omitted.

□

Thus, summarizing the results of this section, we obtain the following algorithm.

Algorithm IDOWN-k
Given: R^{-T} and w, where $X = QR$ and w solves (??).
Input: Set of k observations $[Z^T \ d]$.
Then this algorithm computes $\check{R}^{-T}$ and $\check{w}$, where

$$X = \begin{bmatrix} \check{X} \\ Z^T \end{bmatrix} = QR,$$

$\check{X} = \check{Q}\check{R}$ and $\check{w}$ solves (27).

1. Compute $\check{V} = -R^{-T}Z$.

   Cost $kn^2/2$ multiplications.

2. Find $\check{H} = P_n \cdots P_2 P_1$, where P_i are row hyperbolic Householder transformations, such that

$$\check{H}\begin{bmatrix} \check{V} \\ I \end{bmatrix} = \begin{bmatrix} 0 \\ \check{D} \end{bmatrix}$$

   Cost $13 \cdot k^2 n$ multiplications.

3. Downdate R^{-T} to $\check{R}^{-T}$:

$$\check{H}\begin{bmatrix} R^{-T} \\ 0 \end{bmatrix} = \begin{bmatrix} \check{R}^{-T} \\ F^T \end{bmatrix}$$

   Cost kn^2 multiplications.

4. Downdate w to $\check{w}$:

$$\check{w} = w - F\check{D}^{-T}(d - Z^T w)$$

   Cost $\frac{3}{2}k^2 + 2kn$ multiplications (as from (10) the QR decomposition of D is already available from step 2).

It is easy to see that the complexity analysis for the above algorithm is the same as Algorithm IUP-k. That is, the total cost is $\frac{3}{2} \cdot kn^2 + 13 \cdot k^2 n + 2 \cdot kn + \frac{3}{2}k^2$ multiplications. Moreover, the straight forward implementation of the rank-1 downdating method of Pan and Plemmons [14] requires $\frac{5}{2}kn^2 + O(kn)$ multiplications.

4 Block WY Representation for Products

We are interested in row Householder methods that are rich in matrix-matrix operations in order to increase the efficiency of our algorithms on vector and parallel machines. To that end, it is important to accumulate and apply products of Householder transformations in block form [12].

It is known (see e.g., Schreiber and Van Loan [18]), that products

$$Q = H_n H_{n-1} \cdots H_1$$

of column oriented Householder transformation matrices

$$H_i = I - w_i w_i^T, \quad i = 1, \cdots, n, \tag{41}$$

defined by m-vectors w_i with $w_i^T w_i = 2$, can be accumulated in a compact WY form

$$Q = I - YTY^T \tag{42}$$

where Y is an $m \times n$ rectangular matrix, and each of its columns is a Householder vector w_i, and T is a unit lower triangular $n \times n$ matrix. Obviously, then, if A is an $m \times n$ matrix then $H_n H_{n-1} \cdots H_1 A$ can be accumulated using matrix-matrix operations as

$$H_n H_{n-1} \cdots H_1 A = QA = A - YT(Y^T A).$$

An algorithm for constructing and applying Q in the form (42) is in the new LAPACK software system [1]. We remark that Puglisi [16] has extended the work in [18] by giving a scheme to compute and apply the product form (42) which involves more BLAS-3 matrix-matrix operations, but which also requires additional work and storage.

Clearly, since orthogonal row Householder transformation matrices P as given in (1) can also be written in the form (41), the same results on accumulation and application of products of Householder transformations in block form apply for our case. Thus the use of row Householder orthogonal transformations for modifying the inverse QR factorization is rich in level-3 BLAS operations, and the compact WY representation block algorithms in LAPACK can be used for our application.

The case of row hyperbolic Householder transformations, used for downdating, requires some further discussion. Recall that for an m-vector p_i and a signature matrix Φ, an $m \times m$ row (or column) hyperbolic Householder transformation matrix can be written in the form

$$P_i = \Phi - \frac{2}{p_i^T \Phi p_i} p_i p_i^T, \tag{43}$$

provided that $0 < p_i^T \Phi p_i$. The matrix P_i is pseudo orthogonal with respect to Φ, i.e., $P_i^T \Phi P_i = \Phi$. Observe also that $P = P^T$. We now proceed to show how to accumulate and apply products of hyperbolic Householder transformations in a compact WY-type representation block form similar to (42).

First, we write (43) in the form

$$P_i = \Phi - w_i w_i^T, \tag{44}$$

where w_i is an m-vector given by

$$w_i = \left(\sqrt{\frac{2}{p_i^T \Phi p_i}} \right) p_i.$$

Note that $w_i^T \Phi w_i = 2$.

It will be shown that products

$$Q_\Phi = P_n P_{n-1} \cdots P_1$$

of row or column oriented hyperbolic Householder transformation matrices (44), defined by m-vectors w_i, and associated with the same signature matrix Φ, can be accumulated in a compact WY form

$$Q_\Phi = \Phi^n - \Phi^{n-1} Y T Y^T. \tag{45}$$

A method for computing the block representation (45) is given by the following theorem.

Theorem 3 *Suppose $Q_\Phi = \Phi^i - \Phi^{i-1} Y T Y^T$ is an $m \times m$ matrix, pseudo orthogonal with respect to Φ, with Y $m \times i$ and with T a unit lower triangular $i \times i$ matrix. If $P = \Phi - w w^T$, with w an n-vector such that $0 < w^T \Phi w$, and $z^T = -w^T \Phi^{i-1} Y T$, then the product $P Q_\Phi$ is given by*

$$P Q_\Phi = \Phi^{i+1} - \Phi^i Y_+ T_+ Y_+^T, \tag{46}$$

where

$$Y_+ = \left[Y, \Phi^i w \right], \quad T_+ = \begin{bmatrix} T & 0 \\ z^T & 1 \end{bmatrix}. \tag{47}$$

Proof: It can be seen that

$$\begin{gathered} P Q_\Phi = \left(\Phi - w w^T \right) \left(\Phi^i - \Phi^{i-1} Y T Y^T \right) = \\ \Phi^{i+1} - \Phi^i Y T Y^T + w w^T \Phi^{i-1} Y T Y^T - w w^T \Phi^i = \\ \Phi^{i+1} - \Phi^i Y T Y^T - w z^T Y^T - w w^T \Phi^i = \\ \Phi^{i+1} - \Phi^i \left[Y, \Phi^i w \right] \begin{bmatrix} T & 0 \\ z^T & 1 \end{bmatrix} \begin{bmatrix} Y^T \\ w^T \Phi^i \end{bmatrix} = \\ \Phi^{i+1} - \Phi^i Y_+ T_+ Y_+^T. \end{gathered}$$

□

Notice that $Q_\Phi = P_n P_{n-1} \cdots P_1$ reduces to $\Phi - Y T Y^Y$ if n is odd, and to $I - \Phi Y T Y^T$ if n is even.

The scheme described in Theorem 3 for accumulating products of hyperbolic Householder transformation matrices has the same advantages as the storage-efficient compact WY representation scheme for the orthogonal case given in [18]. In summary, the row orthogonal and row hyperbolic Householder methods considered in this paper are rich in matrix-matrix operations, and this fact can be used to increase the efficiency of our algorithms on vector and parallel machines.

5 Numerical Experiments

In this section we provide numerical experiments which consist of sliding window recursive least squares problems (RLS), and are designed to compare the accuracy of our block method with k applications of the rank-1 covariance inverse factorization RLS method of Pan and Plemmons [14]. In each of the examples given below, we indicate the length of the window used, and the number of observations which will be added and deleted.

The set of examples we use here have been used to test the effectiveness of condition estimators [9, 10, 15], and have also been used by Björck, Park and Eldén [5] to illustrate how the corrected semi-normal equations can be used to stablize rank-1 downdating. These examples are described as follows.

Example 1: In this example we construct a 100×10 data matrix whose entries are generated randomly from a uniform distribution in (-50,50). We then scale the first column of this matrix by multiplying the entries in the first column by 10^{-3}. This causes the windowed data to have a condition number on the order of 10^3. Here we choose the window length to be 20, and the number of observations added and deleted is $k = 5$.

Example 2: In this example we construct a 50×5 data matrix from a uniform distribution in (0,1). In this case, though, we add an outlier of the form $r \times 10^3$, where r is again a random number in (0,1), to the (18,3) entry. The effect of this outlier causes the data to become ill-conditioned when the 18^{th} row is added to the system. Here we choose the window length to be 8, and the number of observations added and deleted is $k = 3$.

Example 3: In this example we construct a 50×5 matrix. The first 25 rows are the first 25 rows of the Hilbert matrix. The second 25 rows are simply the first 25 rows given in reverse order. We then add a random number, δ, to all the entries in order to control the degree of ill-conditioning of the data. The smaller the value of δ, the more ill-conditioned is the data. As is done in [5], we use $\delta = 10^{-5}$ and $\delta = 10^{-9}$. Here, we again take the window length to be 8, and $k = 3$.

The numerical tests for the above examples were performed using Matlab, and the right hand side vector was chosen to be the row sums of the data matrix. Thus the exact solution is known, and is the vector of all ones. The quantities reported are the relative errors and residuals for our block method, and the rank-1 rotation based method of Pan and Plemmons [14]. The results are summarized in Figures 1-8, where the solid line is the plot of the rank-1 method and the dashed line is a plot of our block method. Also shown in the figures is a plot of $1/cond(X)$ for each window, indicated by + signs.

We see from the figures that numerically our block method performs in a similar manner to k applications of the rank-1 method of Pan and Plemmons. But since our methods are rich in BLAS-3 computations, our block method is better suited for vector and parallel architectures.

We note that, as for the rank-1 method of Pan and Plemmons, our block method can give

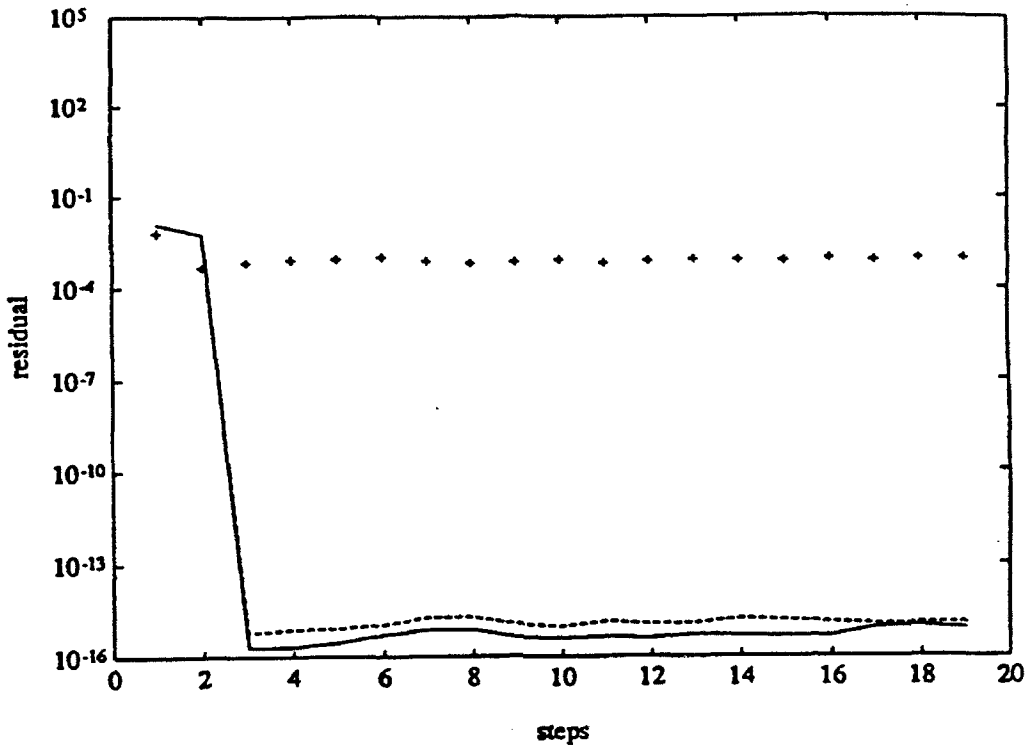


Figure 1: Residuals for Example 1.

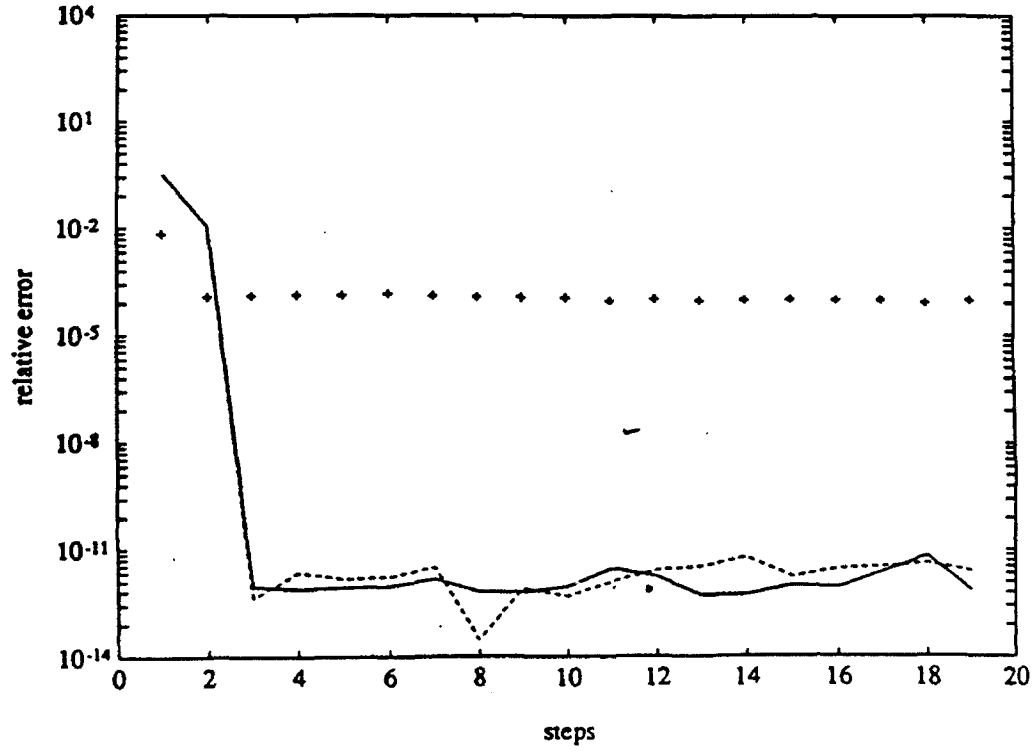


Figure 2: Relative errors for Example 1.

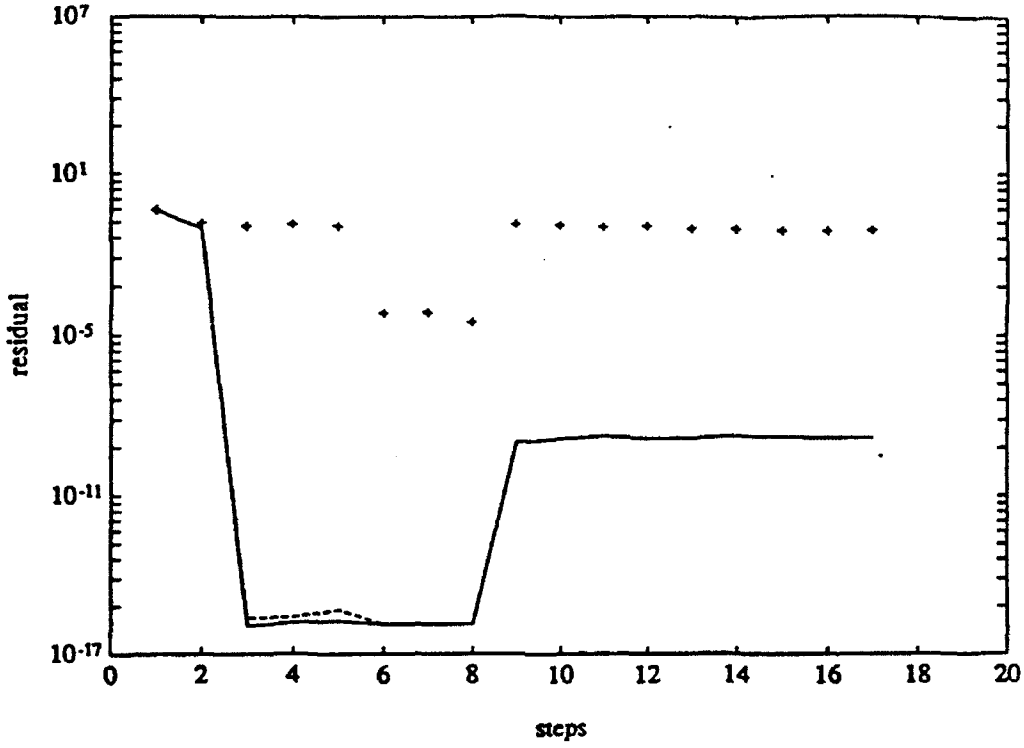


Figure 3: Residuals for Example 2.

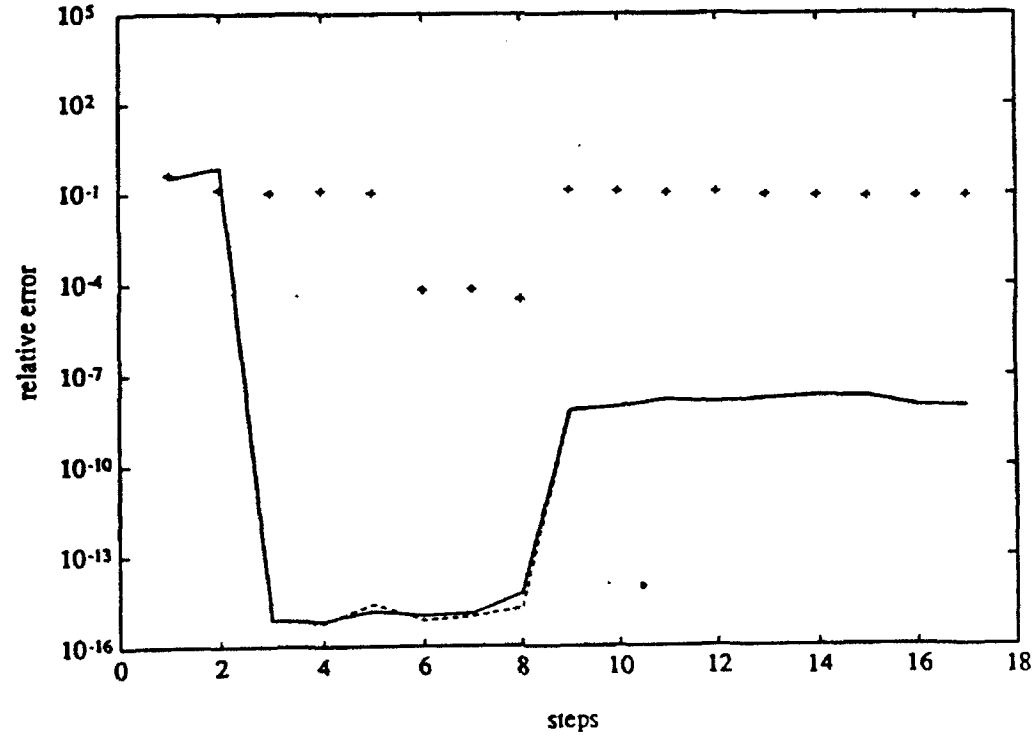


Figure 4: Relative errors for Example 2.

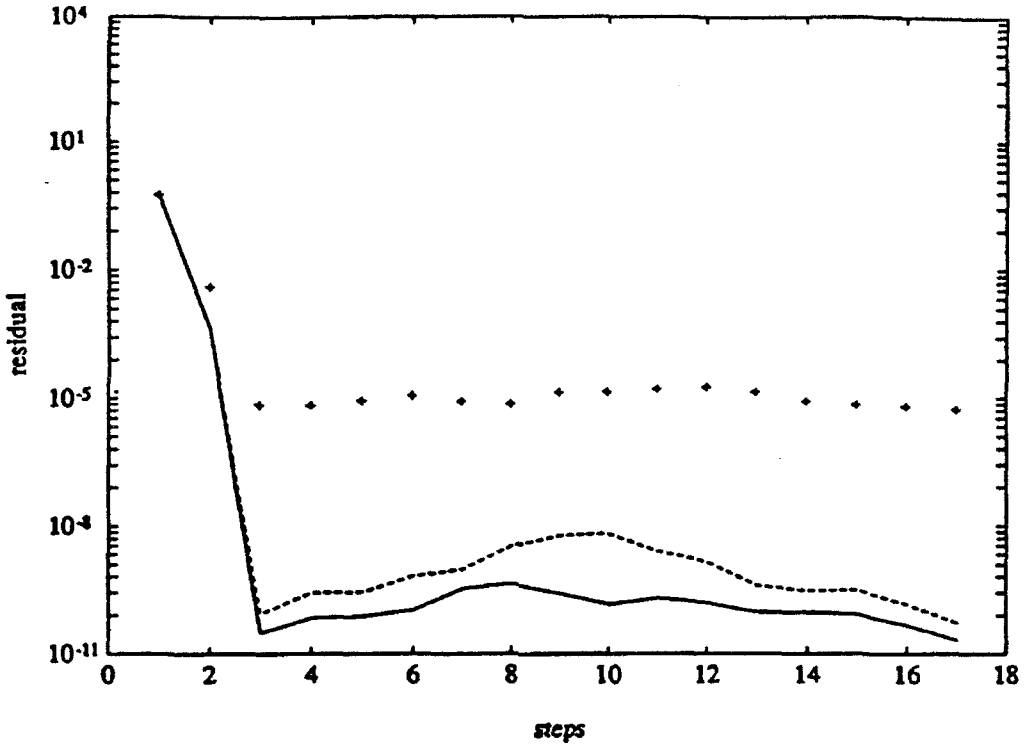


Figure 5: Residuals for Example 3, with $\delta = 10^{-5}$.

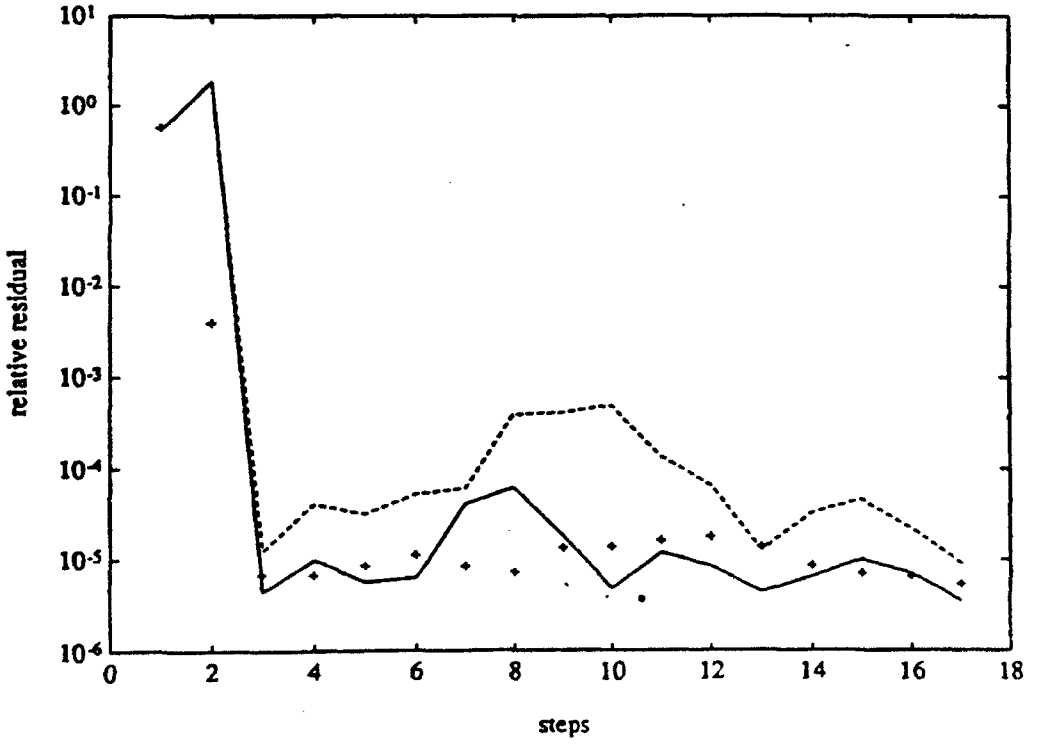


Figure 6: Relative errors for Example 3, with $\delta = 10^{-5}$.

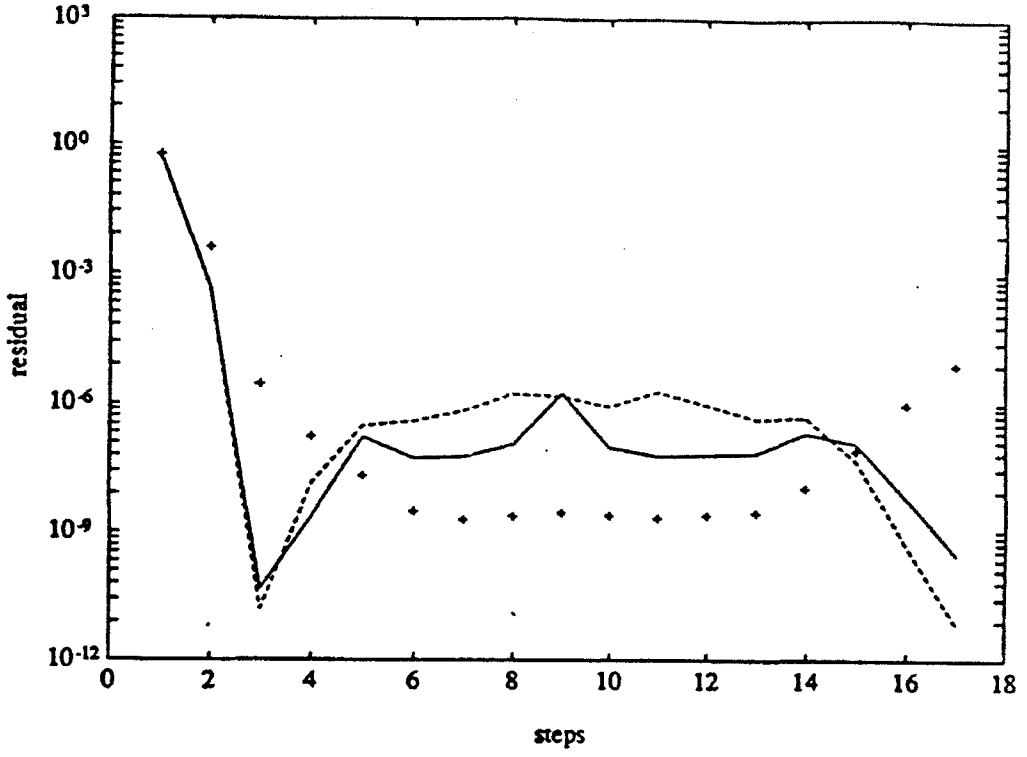


Figure 7: Residuals for Example 3, with $\delta = 10^{-9}$.

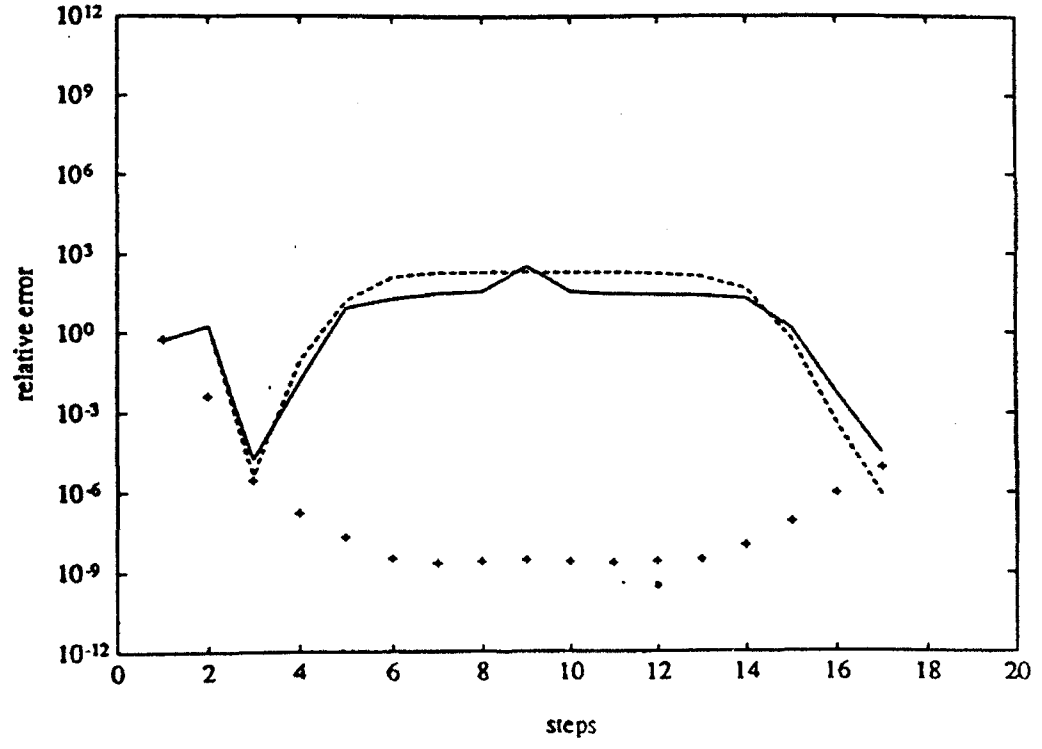


Figure 8: Relative errors for Example 3, with $\delta = 10^{-9}$.

inaccurate results if the data becomes too ill-conditioned. This is to be expected, though, since the downdating is sensitive to ill-conditioning, see [20]. To obtain a more reliable block method when the data is ill-conditioned, one can apply schemes which also modify the Q factor, such as a generalization of the method proposed in [8] for updating the Gram-Schmidt QR factorization to the block case. Another approach is to use the original data, X. This could be done by extending the work of Björck, Park and Eldén [5], which uses the corrected semi-normal equations for rank-1 modifications, to the rank-k case. These two approaches are the subject of the ongoing research and will be reported elsewhere.

Perhaps a more straight forward approach is to use a condition estimation technique, such as ACE [15], and, if the problem becomes ill-conditioned, re-initialize by computing a new inverse orthogonal factorization, producing a new R^{-1}. That is, ACE could be used to monitor the conditioning of the data, which can be done in $O(n) + O(k^3)$ operations per time step. The $O(k^3)$ comes from solving an eigenvalue problem required in ACE. If the data becomes ill-conditioned, one would then compute an explicit QR factorization of the current data, to re-initialize the RLS process, and continue with the updating and downdating. This approach would be most useful for problems such as Example 2, where the data is well conditioned except for a small number of windows, made ill-conditioned by outliers. Of course, if the problem is well conditioned, then our scheme is very efficient and needs no stabilizing modifications.

Acknowledgements: The authors would like to thank the anonymous reviewers whos comments improved the quality of the paper. We are indebted to the rewiever who proposed a very elegant proof of Theorem 1. The authors also wish to acknowledge helpful comments from R. Funderlic on a preliminary version of this paper. The paper was written while all three authors were visiting the Institute for Mathematics and Its Applications at the University of Minnesota.

References

[1] E. Anderson, Z. Bai, C. Bischof, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, S. Ostrouchov, and D. Sorensen, *LAPACK Users' Guide, Release 1.0*, SIAM Press, (1992), Philadelphia.

[2] S. T. Alexander, C.-T. Pan and R. J. Plemmons, *Analysis of a recursive least squares hyperbolic rotation algorithm for signal processing*, Linear Algebra Appl., 98 (1988), pp. 3-40.

[3] R. Bartels and L. Kaufman, *Cholesky factor updating techniques for rank 2 matrix modifications*, SIAM J. Matrix. Anal. Applic., 10 (1989), pp. 557-592.

[4] Å. Björck, Least Squares Methods, in Handbook of Numerical Methods, ed. P. Ciarlet and J. Lions, Elsevier/North Holland Vol. 1, 1989.

[5] Å. Björck, H. Park and L. Eldén, *Accurate Downdating of Least Squares Solutions*, Preprint, 1992, to appear in SIAM J. on Matrix Annal. and Appl.

[6] A.W. Bojanczyk and A.O. Steinhardt, *Stability Analysis of a Householder-Based Algorithm for Downdating the Cholesky Factorization*, SIAM J. on Sci. and Stat. Comp., vol 12, no 5, September 1991.

[7] A. W. Bojanczyk, J. G. Nagy and R. J. Plemmons, *Multi-Row Householder Reflections*, in preparation.

[8] J. Daniel, W.B. Gragg, L. KAufman and G.W. Stewart, *Reorthogonalization and stable algorithms for updating the Gram-Schmidt QR factorization*, Math. Comp., vol 30, pp 772-795, 1976.

[9] W. R. Ferng, *Lanczos-Based Condition Estimation in Signal Processing and Optimization*, Ph.D. Thesis, North Carolina State University, Raleigh, NC, 1991.

[10] W. R. Ferng, G. H. Golub and R. J. Plemmons, *Adaptive Lanczos Methods in Recursive Condition Estimation*, Numerical Algorithms, 1, pp. 1-20, 1991.

[11] P.E. Gill, G.H. Golub, W. Murray and M.A. Saunders, *Methods for Modifying Matrix Factorizations*, Math. Comp., vol 28, pp505-535, 1975.

[12] G. H. Golub and C. Van Loan, Matrix Computations, Johns Hopkins Press, Second Edition, 1989.

[13] A. V. Oppenheim and R. W. Schafer, Discrete-Time signal processing, Prentice-Hall, Englewood Cliffs, NJ 1989.

[14] C.-T. Pan and R. J. Plemmons, *Least Squares modifications with inverse factorizations: parallel implications*, Computational and Appl. Math., 27 (1989), pp. 109-127.

[15] D. J. Pierce and R. J. Plemmons, *Fast Adaptive Condition Estimation*, SIAM J. Matrix. Anal. Applic., 13 (1992), pp. 274-291.

[16] C. Puglisi, *Modification of the Householder method based on the compact WY representation*, SIAM J. Sci. Stat. Comp. 13 (1992) 723-726.

[17] C.M. Rader and A.O. Steinhardt, *Hyperbolic Householder Transformations*, IEEE ASSP, vol 34, no 6, pp 1589-1602, 1986.

[18] R. Schreiber and C. Van Loan, *A storage-efficient WY representation for products of Householder transformations*, SIAM J. Sci. Stat. Comp. 10 (1989) 52-57.

[19] G.W. Stewart, *Perturbation bounds for the QR factorization of a matrix*, SIAM J. Numer. Anal., vol 14, pp 509-518, 1977.

[20] G.W. Stewart, *The effects of rounding errors on an algorithm for downdating a Cholesky factorization*, J. Inst. Maths. Applics., vol 23, pp 203-213, 1979.

Rank-k Modification Methods for Recursive Least Squares Problems *

Serge J. Olszanskyj James M. Lebak Adam W. Bojanczyk
School of Electrical Engineering
Cornell University
E&TC Building
Ithaca, NY 14853-3801

March 4, 1993

Abstract

In least squares problems, it is often desired to solve the same problem repeatedly but with several rows of the data either added, deleted, or both. Methods for adding or deleting one row of data at a time are known. In this paper we introduce fundamental rank-k updating and downdating methods and show how extensions of rank-1 modifications for LINPACK, Corrected Semi-Normal Equations (CSNE), and Gram-Schmidt factorizations can all be derived from these fundamental results. We then analyze the cost of each new algorithm, and make comparisons to k applications of the corresponding rank-1 algorithms. We provide experimental results comparing the numerical accuracy of the various algorithms, paying particular attention to the downdating methods, due to their potential numerical difficulties for ill-conditioned problems.

Abbreviated Title.

Key Words.

AMS(MOS) Subject Classifications.

1 Introduction

A problem which frequently arises in signal processing is the *linear least squares* problem:

$$\min_{x \in \Re^n} \|Ax - b\|_2 \tag{1}$$

*Research supported in part by the Joint Services Electronics Program, contract no. F49620-90-C-0039.

where $A \in \Re^{m \times n}$, A is rank n, $b \in \Re^m$, and $m > n$. This problem may be solved through a QR factorization of the augmented $m \times (n+1)$ matrix $(A\ b)$,

$$\begin{pmatrix} A & b \end{pmatrix} = QR = Q \begin{pmatrix} U & u \\ 0 & \rho \end{pmatrix} \tag{2}$$

where $Q \in \Re^{m \times (n+1)}$ with orthonormal columns, $U \in \Re^{n \times n}$, with U upper triangular, $u \in \Re^{n \times 1}$, and ρ is a scalar. We then insert the factorization into the problem:

$$\begin{aligned} \min_{x \in \Re^n} \|Ax - b\|_2^2 &= \min_{x \in \Re^n} \left\| Q^T A x - Q^T b \right\|_2^2 \\ &= \min_{x \in \Re^n} \left\| \begin{pmatrix} U \\ 0 \end{pmatrix} x - \begin{pmatrix} u \\ \rho \end{pmatrix} \right\|_2^2 \\ &= \min_{x \in \Re^n} \|Ux - u\|_2^2 + \rho^2 \end{aligned}$$

The solution vector x is then found by solving

$$Ux = u \tag{3}$$

by the method of back substitution. The two-norm of the residual of the problem is ρ. Note that once the factorization is found, only U, u, and ρ are needed to solve the problem.

Frequently, one has already found the QR factorization in (2), and wishes to solve (1) with one or more rows added to or deleted from the data $(A\ b)$. This is known as the *recursive least squares* problem. Computing the QR factorization of a matrix is computationally expensive. Since one already has Q and R (or just R) from a problem that is close to the one we wish to solve, we would like to save on computation by just finding the new factorization (say Q_{new} and R_{new}) from the old factorization and the data to be added or deleted. The new solution is then computed by using U_{new} and u_{new} as above in (3).

For example, say one has k new rows, $(Y\ c) \in \Re^{k \times (n+1)}$, and it is desired to append them to the end of the data $(A\ b)$. Then the problem becomes given (2), solve

$$\min_{\hat{x} \in \Re^n} \left\| \begin{pmatrix} A \\ Y \end{pmatrix} \hat{x} - \begin{pmatrix} b \\ c \end{pmatrix} \right\|_2$$

This is called a *rank-k update* of a linear least squares problem. To accomplish this, first make the following construction:

$$\begin{pmatrix} \hat{A} & \hat{b} \end{pmatrix} = \begin{pmatrix} A & b \\ Y & c \end{pmatrix} = \tilde{Q}\tilde{R} = \begin{pmatrix} Q & 0 \\ 0 & I_k \end{pmatrix} \begin{pmatrix} U & u \\ 0 & \rho \\ Y & c \end{pmatrix}$$

Since $\tilde{Q}$ has orthonormal columns, all that needs to be done at this point is to apply an orthogonal transformation, say H, to the factorization, designed to reduce $\tilde{R}$ to upper triangular form (i.e., zero out $(Y\ c)$) while preserving the orthogonal property of the columns of $\tilde{Q}$.

This leads to the following:

$$(\hat{A} \quad \hat{b}) = \begin{pmatrix} Q & 0 \\ 0 & I_k \end{pmatrix} HH^T \begin{pmatrix} U & u \\ 0 & \rho \\ Y & c \end{pmatrix} = \begin{pmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{pmatrix} \begin{pmatrix} \hat{U} & \hat{u} \\ 0 & \hat{\rho} \\ 0 & 0 \end{pmatrix} = \hat{Q}\hat{R} \tag{4}$$

The new solution can be obtained from $\hat{U}\hat{x} = \hat{u}$. The two-norm of the residual of the new problem is $\hat{\rho}$. The problem of determining $\hat{Q}$ and $\hat{R}$ in (4) is known as *updating the QR factorization.*

The other possibility is to remove data from the problem. Assume that we desire to remove the first k rows, $(Z \; d)$, of the data $(A \; b)$. This would be a *rank-k downdate* of the linear least squares problem. Then given (2), the problem to be solved is

$$\min_{\check{x} \in \Re^n} \left\| \check{A}\check{x} - \check{b} \right\|_2, \tag{5}$$

where

$$(A \quad b) = \begin{pmatrix} Z & d \\ \check{A} & \check{b} \end{pmatrix} = QR = \begin{pmatrix} Q_{11} \\ Q_{21} \end{pmatrix} \begin{pmatrix} U & u \\ 0 & \rho \end{pmatrix} \tag{6}$$

Here Z is $k \times n$, d is $k \times 1$, Q_{11} is $k \times (n+1)$, Q_{21} is $(m-k) \times (n+1)$, $\check{A}$ is $(m-k) \times n$, $\check{b}$ is $(m-k) \times 1$, and of course Q has orthonormal columns. The problem of finding the new factorization $(\check{A} \; \check{b}) = \check{Q}\check{R}$ is known as *downdating the QR factorization.* Recall that the updated triangular factor $\hat{R}$ was needed to solve the updated linear least-squares problem: similarly, $\check{R}$ is needed to solve the downdated linear least-squares problem. Depending on the algorithm used, the matrix Q may or may not need to be stored and downdated.

Downdating the QR factorization is the reverse of the updating process (4). That is, the downdating process begins with the augmented factorization (7).

$$\begin{pmatrix} A & b \end{pmatrix} = \begin{pmatrix} Z & d \\ \check{A} & \check{b} \end{pmatrix} = \bar{Q}\bar{R} = \begin{pmatrix} Q_{11} & \tilde{Q}_{12} \\ Q_{21} & \tilde{Q}_{22} \end{pmatrix} \begin{pmatrix} R \\ 0 \end{pmatrix} \tag{7}$$

Here, as in (4), $\bar{Q}$ is an orthonormal column matrix containing Q augmented with k new orthonormal columns. Then an orthogonal transformation H (similar to that used in updating) is applied to obtain $\check{Q}$ and $\check{R}$.

$$\begin{pmatrix} Z & d \\ \check{A} & \check{b} \end{pmatrix} = \begin{pmatrix} Q_{11} & \tilde{Q}_{12} \\ Q_{21} & \tilde{Q}_{22} \end{pmatrix} H^T H \begin{pmatrix} U & u \\ 0 & \rho \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & I_k \\ \check{Q} & 0 \end{pmatrix} \begin{pmatrix} \check{U} & \check{u} \\ 0 & \check{\rho} \\ Z & d \end{pmatrix} = \begin{pmatrix} (Z \quad d) \\ \check{Q}\check{R} \end{pmatrix} \tag{8}$$

Notice that the final terms in (8) have permuted rows (compared to the corresponding terms in (4)) since rows are being deleted from the top as opposed to being added at the bottom. Still, the two-step procedure defined by (7) and (8) is the logical reverse of the updating process.

In downdating, the k new orthogonal columns are not a by-product of applying H as in (4), but must be found before applying H. There is some difficulty in determining $\tilde{Q}_{12}$ and

$\tilde{Q}_{22}$, as they are not unique (observe that an orthogonal transformation may be inserted in the middle of the factorization (7) that alters $\tilde{Q}_{12}$ and $\tilde{Q}_{22}$ but not the other terms). To assist in determining $\tilde{Q}_{12}$ and $\tilde{Q}_{22}$, we examine other requirements on $\bar{Q}$.

From equation (8), $\bar{Q}$ and H must satisfy

$$\begin{pmatrix} Q_{11} & \tilde{Q}_{12} \\ Q_{21} & \tilde{Q}_{22} \end{pmatrix} = \begin{pmatrix} 0 & I_k \\ \check{Q} & 0 \end{pmatrix} H.$$

Multiplication by $(Q_{11}\ \tilde{Q}_{12})^T = H^T(0 I_k)^T$ on both sides of this equation yields

$$\begin{pmatrix} Q_{11} & \tilde{Q}_{12} \\ Q_{21} & \tilde{Q}_{22} \end{pmatrix} \begin{pmatrix} Q_{11}^T \\ \tilde{Q}_{12}^T \end{pmatrix} = \begin{pmatrix} I_k \\ 0 \end{pmatrix}. \tag{9}$$

Therefore the first k *rows* of $\bar{Q}$ must be orthogonal. This orthogonality condition may be combined with expression (7) as shown in (10).

$$\begin{pmatrix} \begin{pmatrix} Z & d \\ \check{A} & \check{b} \end{pmatrix} & I_k \\ & 0 \end{pmatrix} = \begin{pmatrix} Q_{11} & \tilde{Q}_{12} \\ Q_{21} & \tilde{Q}_{22} \end{pmatrix} \begin{pmatrix} R & Q_{11}^T \\ 0 & \tilde{Q}_{12}^T \end{pmatrix} \tag{10}$$

The last k columns of $\bar{Q}$ are still not uniquely determined in (10). However, choosing $\tilde{Q}_{12}$ *to be an upper-triangular matrix*, denoted simply as Q_{12}, and making the corresponding choice Q_{22} for $\tilde{Q}_{22}$, makes (10) an expression of the downdating problem in terms of the QR factorization of an enhanced matrix.

$$\boxed{\begin{pmatrix} \begin{pmatrix} Z & d \\ \check{A} & \check{b} \end{pmatrix} & I_k \\ & 0 \end{pmatrix} = \begin{pmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{pmatrix} \begin{pmatrix} R & Q_{11}^T \\ 0 & Q_{12}^T \end{pmatrix}} \tag{11}$$

This relation is fundamental to all rank-k downdating methods described in this paper. Since it compactly represents all of the conditions necessary to determine $\bar{Q}$, (11) can be used to determine Q_{12} and Q_{22}. The relation (11) is an extension to rank-k of an analogous relation derived in [DGKS76].

Once the factorization (11) is obtained, we then proceed as in (8), constructing some orthogonal transformation H which when applied will produce the downdated factorization. Here H operates on the columns of $\bar{Q}$ to transform $(Q_{11}\ Q_{12})$ to $(0\ P)$, where P is an orthogonal matrix. H can be constructed such that H^T, when applied to R, will change the element values in R but still preserve its upper triangular property. This produces the following result:

$$\begin{pmatrix} \begin{pmatrix} Z & d \\ \check{A} & \check{b} \end{pmatrix} & I_k \\ & 0 \end{pmatrix} = \begin{pmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{pmatrix} HH^T \begin{pmatrix} R & Q_{11}^T \\ 0 & Q_{12}^T \end{pmatrix} = \begin{pmatrix} 0 & P \\ \check{Q} & 0 \end{pmatrix} \begin{pmatrix} \check{U} & \check{u} & 0 \\ 0 & \check{\rho} & 0 \\ \check{Z} & \check{d} & P^T \end{pmatrix} \tag{12}$$

Because $\bar{Q}$ has orthonormal columns, this transformation will also implicitly zero out Q_{22}. The equation $\check{U}\check{x} = \check{u}$ is solved to find the solution $\check{x}$ to the downdated problem. The two-norm of the residual corresponding to the downdated problem is $\check{\rho}$.

This paper is devoted to the discussion of rank-k downdating methods derived from (11). We emphasize downdating and not updating because downdating is numerically harder. The k new columns found in the downdating problem have several special properties: orthogonality in columns, orthogonality in rows, and triangularity. This allows for considerable variation and discussion in possible implementations. In all of these implementations, however, *the most important quantities are the k orthogonal rows,* $(Q_{11}\ Q_{12})$. These are needed for finding the triangular factor $\check{R}$ of $(\check{A}\ \check{b})$. If the matrix $\check{Q}$ is to be determined as well, then Q_{22} is also necessary. In Section 2 we discuss methods than maintain Q and R. Section 3 presents methods that only maintain R. In both cases, the methods for obtaining and applying H are the same, and Section 4 examines possible implementations for this part of the downdating process. Section 5 contains an analysis of the computation involved in all of the methods presented.

Since rank-one updating and downdating methods are known, one might achieve a rank-k modification to the data by k applications of these rank-one methods. However, rank-k methods make use of matrix-vector and matrix-matrix operations, as opposed to vector-vector and matrix-vector methods in rank-one algorithms. This may make rank-k methods faster on processors with caches and parallel computers than the corresponding repetitive applications of rank-one methods. Section 6 of this paper presents experimentation on the numerical properties of our algorithms.

2 Rank-k Downdating of the Gram-Schmidt Factorization

In this section we discuss downdating the recursive least squares problem where the Gram-Schmidt factorization is maintained, i.e., modifying both Q and R as described in Section 1. Specifically we discuss several methods by which to obtain the k new columns in the orthogonal factor. In Section 4 we discuss methods by which elements in the constructed factorization are zeroed out to produce the desired downdated Q and R.

2.1 Classical Gram-Schmidt on augmented problem

The first method is to use classical Gram-Schmidt with reorthogonalization (CGS) to build on the orthonormal columns already given. Equation (11) represents a QR factorization, which could have been accomplished by classical Gram-Schmidt. Since we have Q_{11}, Q_{21}, and R already, we have completed $n+1$ iterations (recall that the ith iteration of classical Gram-Schmidt produces the ith column of Q and the ith column of R). We may then proceed with the remaining k iterations of the orthogonalization process to get the new orthogonal columns.

2.2 Modified Gram-Schmidt on augmented problem

Modified Gram-Schmidt (MGS) can also be used to get the new orthonormal columns. Recall that the ith iteration of MGS produces the ith column of Q and the ith row of R in and updates the columns of Q to be formed in later iterations. Again we use the fact that (11) represents a QR factorization that is partially completed. After $n+1$ iterations of MGS, we have the following:

$$\left(\begin{Bmatrix} Z & d \\ \check{A} & \check{b} \end{Bmatrix} \begin{matrix} I_k \\ 0 \end{matrix} \right) = \begin{pmatrix} Q_{11} & T_1 \\ Q_{21} & T_2 \end{pmatrix} \begin{pmatrix} R & Q_{11}^T \\ 0 & I_k \end{pmatrix}$$

The only part of the above that we do not have is $T = (T_1^T \; T_2^T)^T$. We can find T by performing only the update portion of MGS (i.e., $T = T - q_i q_i^T T$) for each of the $n+1$ orthogonal columns from $(Q_{11}^T \; Q_{21}^T)^T$ one at a time in ascending order on the new k columns $(T_1^T \; T_2^T)^T$. Then we can proceed with the remaining k iterations of MGS.

2.3 Small QR factorization

Note that from (11)

$$\begin{pmatrix} I_k \\ 0 \end{pmatrix} = \begin{pmatrix} Q_{11}Q_{11}^T + Q_{12}Q_{12}^T \\ Q_{21}Q_{11}^T + Q_{22}Q_{12}^T \end{pmatrix}$$

which can be rearranged into the following:

$$\begin{pmatrix} Q_{12} \\ Q_{22} \end{pmatrix} Q_{12}^T = \begin{pmatrix} I_k - Q_{11}Q_{11}^T \\ -Q_{21}Q_{11}^T \end{pmatrix} \tag{13}$$

Since Q_{12}^T is upper triangular, this is a QR factorization and Q_{12} and Q_{22} can be obtained by any QR factorization algorithm, in particular by either using MGS or CGS.

2.4 Separation of equations

We can rewrite (13) as follows:

$$\begin{pmatrix} Q_{12}Q_{12}^T \\ Q_{22}Q_{12}^T \end{pmatrix} = \begin{pmatrix} I_k - Q_{11}Q_{11}^T \\ -Q_{21}Q_{11}^T \end{pmatrix} \tag{14}$$

Instead of computing a QR factorization on the whole factor, we can solve for Q_{12} and Q_{22} in separate steps. The top equation represents a Cholesky factorization that can be used to solve for Q_{12}. Then Q_{12} can be used in the bottom equation to get Q_{22} by forward substitution.

2.5 QR of the residual matrix T

The left-hand side of (13) is related to a least squares problem associated with the matrix defined in (11), namely,

$$\min_X \left\| \left(\begin{pmatrix} A & b \end{pmatrix} \begin{pmatrix} I_k \\ 0 \end{pmatrix} \right) \begin{pmatrix} X \\ -I_k \end{pmatrix} \right\|_F . \tag{15}$$

Now (11) implies

$$\min_X \left\| \left(\begin{pmatrix} Z & d \\ \check{A} & \check{b} \end{pmatrix} \begin{matrix} I_k \\ 0 \end{matrix} \right) \begin{pmatrix} X \\ -I \end{pmatrix} \right\|_F = \min_X \left\| \begin{pmatrix} R & Q_{11}^T \\ 0 & Q_{12}^T \end{pmatrix} \begin{pmatrix} X \\ -I \end{pmatrix} \right\|_F \tag{16}$$

and hence the solution X satisfies

$$RX = Q_{11}^T \tag{17}$$

while the residual error has the following representation:

$$T = \begin{pmatrix} A & b \end{pmatrix} X - \begin{pmatrix} I_k \\ 0 \end{pmatrix} = \begin{pmatrix} Q_{12} \\ Q_{22} \end{pmatrix} Q_{12}^T. \tag{18}$$

Note that in (18) the right-hand side is the QR factorization of T. This is not surprising as T is the residual in the least squares problem (16), and hence the columns of T span a subspace orthogonal to the column space of the defining matrix $(A\ b)$.

However it may not be easy to get a numerically accurate orthonormal base for T which is orthogonal to the column space of $(A\ b)$. This will be the case if, for example, the columns of T have substantially different magnitudes of norms. The following method for factorizing T uses a step of refinement (or reorthogonalization) in order to provide improved numerical results.

We first insert T into an augmented factorization problem to ensure that its orthogonal columns will span a subspace orthogonal to the subspace of $(A\ b)$:

$$\left(\begin{pmatrix} A & b \end{pmatrix} \ T \right) = \begin{pmatrix} Q_{11} & -Q_{12} \\ Q_{21} & -Q_{22} \end{pmatrix} \begin{pmatrix} R & R_{12} \\ 0 & R_{22} \end{pmatrix} \tag{19}$$

From this, we have an equation for T from which we can get the desired factorization:

$$T = \begin{pmatrix} Q_{11} \\ Q_{21} \end{pmatrix} R_{12} - \begin{pmatrix} Q_{12} \\ Q_{22} \end{pmatrix} R_{22} \tag{20}$$

First we need to multiply both sides of (20) by $(Q_{11}^T\ Q_{21}^T)$ to get R_{12}:

$$R_{12} = \begin{pmatrix} Q_{11}^T & Q_{21}^T \end{pmatrix} T$$

Now we "correct" T:

$$T_c = T - \begin{pmatrix} Q_{11} \\ Q_{21} \end{pmatrix} R_{12}$$

We can then factor T_c:

$$-\begin{pmatrix} Q_{12} \\ Q_{22} \end{pmatrix} R_{22} = T_c \tag{21}$$

Note that if we multiply both sides of (21) by $-(Q_{12}^T\ Q_{22}^T)$ and use (18), we have $R_{22} = Q_{12}^T$, and we are done.

3 Downdating R Without Storing Q

Methods for rank-k downdating of R without maintaining Q have been proposed in [RS86] and [BS89]. The method proposed in [RS86] is based on the fact that as long as $R^TR - Z^TZ$ is positive definite then there exists a pseudo-orthogonal transformation H with respect to the signature matrix $\Phi = diag(I_n, -I_k)$ such that

$$H\begin{pmatrix} R \\ Z \end{pmatrix} = \begin{pmatrix} U \\ 0 \end{pmatrix} \tag{22}$$

where U is the Cholesky factor of $R^TR - Z^TZ$. It is shown in [RS86] that the transformation H can be constructed as a product of hyperbolic Householder transformations.

An alternative approach for rank-k downdating of R has been proposed in [BS89]. Their downdating of R is treated as an implicit updating of U. More precisely, they show that there exists an orthogonal H such that

$$H^T\begin{pmatrix} U \\ Z \end{pmatrix} = \begin{pmatrix} R \\ 0 \end{pmatrix} \tag{23}$$

where U is the desired Cholesky factor of $R^TR - Z^TZ$. It is shown in [BS89] that H can be constructed as a product of orthogonal Householder transformations and U can be recovered in a row by row fashion from R, Z, and H. Note that (22) and (23) do not require any explicit information about the orthogonal factor Q.

Recall that (8) represented downdating as implicit updating as well. Equation (23) is actually embedded in (8) (i.e., remove the Q related factors and associated applications of H, leaving (8)). Further, if one desired to determine Q from H, R, Z determined above, Q_{21}, Q_{22} and thus the downdated data $\check{A}$, $\check{b}$ would not necessarily be unique.

In this section we present generalizations to the block case of two other methods for downdating the R factor without storing the matrix Q.

The first method is based on the rank-1 downdating method proposed by [Saunders] that was later implemented in LINPACK. The second method is based on the method of corrected semi-normal equations (CSNE) proposed by [Björ87]. These two methods can be derived from (11) and require partial information about Q.

The primary difference between the methods presented in Section 2 and the LINPACK and CSNE methods is that the latter methods do not store any part of the matrix Q. Instead,

Q_{11} is recovered from the following relation

$$\begin{pmatrix} U & u \\ 0 & \rho \end{pmatrix}^T Q_{11}^T = \begin{pmatrix} Z^T \\ d^T \end{pmatrix} \tag{24}$$

which comes from the first k rows of the QR decomposition of $(A\ b)$ in (11). Of the other blocks of Q, namely Q_{12} , Q_{21}, and Q_{22}, only Q_{12} needs to be found in order to update R. Each method finds Q_{12} in a different way. Once Q_{11} and Q_{12} are found, both the LINPACK method and CSNE method use the orthogonal transformation H defined by (12) to produce the downdated R-factor. For the purposes of this discussion, we will concentrate only on the ways in which the two algorithms find Q_{11} and Q_{12}, and so "a rank-k LINPACK algorithm" refers to an algorithm for rank-k downdating which finds Q_{11} and Q_{12} in an analogous way to the rank-one LINPACK algorithm, and a "rank-k CSNE algorithm" refers to an algorithm for rank-k downdating which finds Q_{11} and Q_{12} in an analogous way to the rank-one CSNE algorithm.

3.1 The LINPACK Downdating Algorithm

The rank-one LINPACK algorithm is actually a part of method of separation of equations (14). Once Q_{11} has been found by (24), Q_{12} is found from the top k rows in (14), namely from

$$Q_{12}Q_{12}^T = I_k - Q_{11}Q_{11}^T. \tag{25}$$

Notice that in the rank-one case, Q_{12} is found by a single square root operation. The lower $m-k$ rows of (14) are not used as Q does not need to be maintained.

Note that (25) itself can be viewd as a downdating problem. Thus Q_{12} can be formed either directly as the Cholesky factor of $I_k - Q_{11}Q_{11}^T$, or indirectly by constructing an orthogonal G as in (23) such that

$$G\begin{pmatrix} Q_{11}^T \\ Q_{12}^T \end{pmatrix} = \begin{pmatrix} I \\ 0 \end{pmatrix},$$

see [BS89].

3.2 The CSNE Downdating Algorithm

The CNSE downdating algorithm presented in [BPE92] is shown in that paper to be more stable than the LINPACK method for rank-one downdating. Thus it is desirable to extend this method to the rank-k case. The basis of this method is in the semi-normal equations for a least squares problem

$$\min \|By - c\|.$$

If U is the triangular factor from the QR factorization of B, then the SNE for a single least-squares problem are given by (26),

$$U^T U x = A^T b. \tag{26}$$

The corrected semi-normal equations (CSNE) method for solving a least-squares problem proposed in [Bjö87] uses refinement of the solution obtained by the SNE, as shown in (27).

$$\begin{aligned} U^T U y &= B^T c, \quad t = c - By \\ U^T U \delta y &= B^T t, \quad y_c = y + \delta y, \quad t_c = t - B\delta y \end{aligned} \tag{27}$$

The corrected solution y_c and the corrected residual vector t_c will have consistently better accuracy than the accuracy of the solution obtained by the SNE method, and often comparable to that given by a standard QR factorization method (see [Bjö87]).

The CSNE downdating method in [BPE92] is derived by using (27) in a least-squares problem to approximate the first column of the identity matrix. In the rank-k case, the CSNE will be used to solve k simultaneous least-squares problems approximating the first k columns of the identity matrix as in (15).

$$\min_{V,\Phi} \left\| \begin{pmatrix} A & b \end{pmatrix} \begin{pmatrix} V \\ \Phi \end{pmatrix} - \begin{pmatrix} I_k \\ 0 \end{pmatrix} \right\|_F = \min_{V,\Phi} \left\| \begin{pmatrix} A & b & \begin{pmatrix} I_k \\ 0 \end{pmatrix} \end{pmatrix} \begin{pmatrix} V \\ \Phi \\ -I_k \end{pmatrix} \right\|_F \tag{28}$$

Here $V \in \Re^{n \times k}$ and Φ is a length k row vector. Following the relation (27), it is now straightforward to derive a block version of the CSNE downdating method.

Using the SNE, k systems of equations are obtained:

$$\begin{pmatrix} U & u \\ 0 & \rho \end{pmatrix}^T \begin{pmatrix} U & u \\ 0 & \rho \end{pmatrix} \begin{pmatrix} V \\ \Phi \end{pmatrix} = \begin{pmatrix} A^T \\ b^T \end{pmatrix} \begin{pmatrix} I_k \\ 0 \end{pmatrix} = \begin{pmatrix} Z^T \\ d^T \end{pmatrix} \tag{29}$$

Equation (29) can easily be broken down into two triangular systems of equations, the first of which (30) turns out to be exactly equation (24). Therefore, Q_{11} is solved for in exactly the same manner as in the LINPACK method.

$$\begin{pmatrix} U & u \\ 0 & \rho \end{pmatrix}^T Q_{11}^T = \begin{pmatrix} Z^T \\ d^T \end{pmatrix} \tag{30}$$

$$\begin{pmatrix} U & u \\ 0 & \rho \end{pmatrix} \begin{pmatrix} V \\ \Phi \end{pmatrix} = Q_{11}^T \tag{31}$$

The residual error in the k systems of equations is an $m \times k$ matrix T,

$$T = \begin{pmatrix} I_k \\ 0 \end{pmatrix} - \begin{pmatrix} A & b \end{pmatrix} \begin{pmatrix} V \\ \Phi \end{pmatrix},$$

each column of which represents the error in one system. By substituting this error back into the same systems and solving again, correction factors for Q_{11} and $\begin{pmatrix} V^T & \Phi^T \end{pmatrix}^T$ may be found.

$$\begin{pmatrix} U & u \\ 0 & \rho \end{pmatrix}^T \delta Q = \begin{pmatrix} A & b \end{pmatrix}^T T$$

$$\begin{pmatrix} U & u \\ 0 & \rho \end{pmatrix} \begin{pmatrix} \delta V \\ \delta\Phi \end{pmatrix} = \delta Q$$

The correction factor δQ is added to the matrix Q_{11}, producing a corrected factor Q_{11}^c. The quantity $\left(\delta V^T\ \delta\Phi^T\right)^T$ is not actually used to update $\left(V^T\ \Phi^T\right)^T$ as the latter quantity is not important for our purposes. However, δV and $\delta\Phi$ are used to correct T,

$$T_c = T - \begin{pmatrix} A & b \end{pmatrix} \begin{pmatrix} \delta V \\ \delta\Phi \end{pmatrix}$$

The remaining block Q_{12}^T is found from (18) as the R factor in the QR factorization of T_c.

3.3 Hybrid approach

The rank-k CSNE algorithm has a much greater cost than the rank-k LINPACK algorithm in terms of floating-point operations. Much of this cost may be attributed to the "refinement" process, which may not be necessary if the matrix is well-conditioned. For this reason, [BPE92] suggests a *hybrid* algorithm. In the rank-one hybrid downdating algorithm used by [BPE92], the quantity $\gamma = 1 - \|Q_{11}\|^2$ (where Q_{11} here is just a vector) is checked against a user-specified tolerance. If γ is greater than a user-specified tolerance, the downdating problem is considered well-conditioned and the LINPACK downdating algorithm is used. Otherwise, the CSNE algorithm is used. The suggested range of the tolerance is [0.25, 0.5] [BPE92].

4 Determining the orthogonal reduction factor H

This section discusses the problem expressed in (12), that is developing and applying the orthogonal reduction factor H to the augmented factorization obtained in (11) to produce the downdated factors. Given that H has to be orthogonal, the requirements for H can be more compactly represented as follows:

$$H^T \begin{pmatrix} R & Q_{11}^T \\ 0 & Q_{12}^T \end{pmatrix} = \begin{pmatrix} \check{U} & \check{u} & 0 \\ 0 & \check{\rho} & 0 \\ \check{Z} & \check{d} & P^T \end{pmatrix}. \tag{32}$$

Thus we must preserve the triangularity of R while zeroing out Q_{11}^T. Note that methods for Gram-Schmidt factorizations and for Cholesky factorizations can make use of the same reduction methods. The difference is whether Q is maintained at all. If so, H must also be applied to Q_{21} and Q_{22} as in (12).

4.1 Givens rotations - column dominant ordering

We can construct the orthogonal reduction factor H in (32) using Givens rotations. The order of the rotations preserves the triangular factor, while the values for each rotation are chosen in such a way as to produce the desired reduction in Q_{11}^T.

The Givens rotations will act on pairs of rows of the target matrix. Let $G^T_{i,j}$ represent a Givens rotation acting on the ith row of $\left(R\ Q^T_{11}\right)$ and the jth row of $\left(0\ Q^T_{12}\right)$. We can exploit the fact that Q^T_{12} is upper triangular and zero out Q^T_{11} a column at a time in the following way.

For the first column of Q^T_{11}, apply the product of Givens rotations, $G^T_{1,1}G^T_{2,1}...G^T_{n+1,1}$, to the target matrix from the left, with each rotation zeroing out an element of the first column of Q^T_{11} from bottom to top. Then we have the following result:

$$G^T_{1,1}G^T_{2,1}...G^T_{n+1,1}\begin{pmatrix} R & Q^T_{11} \\ 0 & Q^T_{12} \end{pmatrix} = \begin{matrix} \\ n+1 \\ 1 \\ k-1 \end{matrix}\begin{matrix} \begin{matrix} n+1 & 1 & k-1 \end{matrix} \\ \begin{pmatrix} \tilde{R} & 0 & Q^T_{11} \\ z^T_1 & 1 & 0 \\ 0 & 0 & Q^T_{12} \end{pmatrix} \end{matrix}$$

Note that when the first column of Q^T_{11} is zeroed out, the $(1,1)$ element of Q^T_{12} becomes 1 since $(Q_{11}\ Q_{12})^T$ has orthogonal columns. The remaining row of Q^T_{12} is zeroed out due to the orthogonality property expressed in (11). $\tilde{R}$ remains triangular.

This process is repeated for the remaining columns of Q^T_{11}, where for the ith column of Q^T_{11}, the (i,i) element of Q^T_{12} is used as the pivot. H is the cumulative product of the Givens rotations. Incidentally, $P^T = I_k$.

4.2 Givens rotations - diagonal dominant ordering.

The ordering of Givens rotations to perform the necessary data reduction is by no means unique. To show this, we first note that we can solve (12) differently, that is we could just as easily reduce down to the *first* k columns of Q:

$$\begin{pmatrix} \begin{pmatrix} Z & d \\ \check{A} & \check{b} \end{pmatrix} & I_k \\ 0 & \end{pmatrix} = \begin{pmatrix} Q_{11} & Q_{12} \\ Q_{21} & Q_{22} \end{pmatrix} H_1 H^T \begin{pmatrix} R & Q^T_{11} \\ 0 & Q^T_{12} \end{pmatrix} = \begin{pmatrix} P & 0 \\ 0 & \hat{Q} \end{pmatrix}\begin{pmatrix} \hat{Z} & \hat{d} & P^T \\ \hat{U} & \hat{u} & 0 \\ 0 & \hat{\rho} & 0 \end{pmatrix}$$

Our new requirement is now

$$H^T\begin{pmatrix} R & Q^T_{11} \\ 0 & Q^T_{12} \end{pmatrix} = \begin{pmatrix} \hat{Z} & \hat{d} & P^T \\ \hat{U} & \hat{u} & 0 \\ 0 & \hat{\rho} & 0 \end{pmatrix}. \tag{33}$$

One way to do this with Givens rotations is to zero out $(Q_{11}\ Q_{12})^T$ a *diagonal* at a time, starting with the main diagonal of Q_{12} and working upwards. K Givens rotations are needed to zero each diagonal. Each rotation works on consecutive rows (say i and $i+1$), and each rotation in one diagonal operation works on consecutively higher numbered rows (i.e., the first rotation works on $(i, i+1)$, the second on $(i+1, i+2)$, etc.).

Thus the first step, zeroing the main diagonal of Q^T_{12}, involves Givens rotations working on rows $n+1$ through $n+1+k$ of the target matrix. After application, we would have the

following result:

$$G^T_{n+k,n+k+1} G^T_{n+k-1,n+k} \cdots G^T_{n+1,n+2} \begin{pmatrix} R & Q^T_{11} \\ 0 & Q^T_{12} \end{pmatrix} = \begin{matrix} n+1 \\ k-1 \\ 1 \end{matrix} \begin{pmatrix} \begin{matrix} n-1 \\ \bar{R} \end{matrix} & \begin{matrix} 1 & k-1 \\ \bar{Q}^T_{11} \end{matrix} \\ \begin{pmatrix} 0 & \bar{u} \\ 0 & \check{\rho} \end{pmatrix} & \begin{pmatrix} 0 & \bar{Q}^T_{12} \\ 0 & 0 \end{pmatrix} \end{pmatrix}$$

$\bar{Q}^T_{12}$ is upper triangular. Incidentally, note that $\check{\rho}$ is found after only k Givens rotations. This process is repeated for the proceeding diagonals until Q^T_{11} is upper triangular. The new triangular factor and downdated rows are produced, and due to orthogonality, $P^T = I_k$.

4.3 Row Householder transformations

We return to the requirements for H specified in (32). We can also use a special form of the Householder transformation to zero out Q^T_{11}. Using a standard, column Householder transformation will not preserve the triangularity of R. However, a *row Householder transformation* [BNP92], can preserve the triangularity of R.

Each row Householder transformation will zero out one row of Q^T_{11}, working from the last row of Q^T_{11} to the first. The application of a row Householder transformation is as follows. The transformation is of the same form as a standard Householder transformation, and is applied to the row that is to be partially zeroed (say the first row of the target matrix where q^T from Q^T_{11} is to be zeroed) and to the rows that contain the residual bottom square factor of the columns in which the zeroing takes place (here it is Q^T_{12}). The elements of the Householder vector are determined by solving the system of equations $D^T p = \mu q$ where μ is a normalization constant, and p is the Householder vector. After the first row Householder transformation is applied,

$$H^T_1 \begin{pmatrix} R & Q^T_{11} \\ 0 & Q^T_{12} \end{pmatrix} = \begin{matrix} n \\ 1 \\ k \end{matrix} \begin{pmatrix} \begin{matrix} n-1 \\ \bar{R} \end{matrix} & \begin{matrix} 1 & k \\ \bar{Q}^T_{11} \end{matrix} \\ \begin{pmatrix} 0 & \bar{z} \end{pmatrix} & \bar{Q}^T_{12} \end{pmatrix}$$

The triangularity of R is preserved. The process is repeated until all rows of Q^T_{11} have been zeroed out.

5 Work Analysis

In this section we attempt to make a theoretical comparison of the amount of work involved for the algorithms discussed in the previous sections. Table 1 compares each of the methods used to obtain the needed portions of the necessary k new orthogonal columns. Here the comparison is in terms of operations: multiplies, additions and subtractions, and divides and square roots. The calculations are based on a rank-k downdate being performed on an $m \times n$ data matrix (note that first order terms for multiplies and adds/subtracts have been

ignored). From Table 1, it can be seen that the most expensive methods appear to be CSNE, and potentially CGS and the LINPACK-CSNE hybrid method in worst case scenarios. The least expensive is easily LINPACK, with the remaining algorithms somewhere in between.

Algorithm		Total Operations		
		Multiply	Add, Subtract	Div, Sqrt
CGSAUG	*min*	$2mnk + mk^2 + 2mk$	$2mnk + mk^2 + 2mk + nk + 0.5k^2$	$m + 2k$
	max	$4mnk + 2mk^2 + 4mk$	$4mnk + 2mk^2 + 4mk + 2nk + k^2$	$m + 4k$
MGSAUG		$2mnk + mk^2 + 2mk$	$2mnk + mk^2 + 2mk$	$mk + k$
SMALLQR		$mnk + mk^2 + mk$	$mnk + mk^2 + mk + k^2$	$mk + k$
SMALLCHOL		$mnk+0.5mk^2-1/3k^3+1.5mk+0.5k^2$	$mnk+0.5mk^2-1/3k^3+0.5mk+2k^2$	$mk - 0.5k^2 + 3/2k$
RESQR		$3mnk+mk^2+0.5n^2k+3mk+0.5nk$	$3mnk+mk^2+0.5n^2k+5mk+0.5nk$	$mk + nk + 2k$
LINPACK		$0.5n^2k + nk^2 + 1/6k^3 + 0.5nk + k^2$	$0.5n^2k+nk^2+1/6k^3+0.5nk+2.5k^2$	$nk + 0.5k^2 + 2.5k$
CSNE		$3mnk + mk^2 + 2n^2k + 3mk + 2nk$	$3mnk + mk^2 + 2n^2k + 5mk + 2nk$	$mk + 4nk + 5k$
HYBRID	*min*	$0.5n^2k + nk^2 + 1/6k^3 + 0.5nk + k^2$	$0.5n^2k+nk^2+1/6k^3+0.5nk+2.5k^2$	$nk + 0.5k^2 + 2.5k$
	max	$3mnk + mk^2 + 2n^2k + 3mk + 2nk$	$3mnk + mk^2 + 2n^2k + 5mk + 2nk$	$mk + 4nk + 5k$

Table 1: Comparison of the work involved for each method via operation counts

If one were to implement these algorithms on a microprocessor that had a BLAS library available for it (say, for example, the Intel i860), then one would try to make use of the BLAS library wherever possible, since such a library is often well-optimized for the target processor. If the processor in question has pipelining or vectorization available, operation counts may not give an accurate prediction of relative execution time. Thus we also provide Table 2, which gives a breakdown of components of each algorithm in terms of BLAS functions and the size of the problem each call solves.

Algorithm		BLAS-3				BLAS-2		
		GEMM		TRSM		GEMV		GER
		$O(mnk)$	$O(nk^2)$	$O(mn^2)$	$O(n^2k)$	$O(mn)$	$O(mk)$	$O(mk)$
CGSAUG	*min*					$2k$		
	max					$4k$		
MGSAUG							$n+1$	$n+1$
SMALLQR		1						
SMALLCHOL		1		1				
RESQR		1			1			
LINPACK			1		1			
CSNE		3			4			
HYBRID	*min*		1		1			
	max	3			4			

Table 2: Comparison of the work involved for each method in terms of functions

In Table 4 we compare the reduction methods in terms of operation counts. Note that while the ordering of each of the Givens rotation sequences is different, the amount of work for each is essentially the same.

Table 5 compares the total work involved for various methods. Here we examine Classical Gram Schmidt (maximum work case) with a Givens rotation methods of reduction, and CSNE, also with a Givens rotation reduction.

Algorithm		BLAS-1 NRM2,AXPY,DOT,SCAL		
		$O(m)$	$O(n)$	$O(k)$
CGSAUG	*min*	$3k$	k	
	max	$4k$	$2k$	
MGSAUG	k^2+k			
SMALLQR	k^2+k			
SMALLCHOL				$0.5k^2+0.5k$
RESQR		k^2+k		
LINPACK				$0.5k^2+0.5k$
CSNE		k^2+k		
HYBRID	*min*			$0.5k^2+0.5k$
	max	k^2+k		

Table 3: Comparison of the work involved for each method in terms of functions

Algorithm		Total Operations		
		Multiply	Add, Subtract	Div, Sqrt
GIVENS1	*(GS)*	$4mnk+2n^2k+4mk+8nk$	$2mnk+n^2k+2mk+4nk$	$3nk+3k$
GIVENS2	*(GS)*	$4mnk+2n^2k+4mk+8nk$	$2mnk+n^2k+2mk+4nk$	$3nk+3k$
GIVENS1	*(R)*	$2n^2k+4nk^2+8nk+4k^2$	$n^2k+2nk^2+4nk+2k^2$	$3nk+3k$
GIVENS2	*(R)*	$2n^2k+4nk^2+8nk+4k^2$	$n^2k+2nk^2+4nk+2k^2$	$3nk+3k$

Table 4: Comparison of the work involved for each reduction method via operation counts

We should also note that the storage requirements for each of the algorithms are essentially the same. LINPACK and CSNE must both store R and $(A\ b)$. The Gram-Schmidt methods must maintain R and Q. This is the same amount of storage in both cases since $(A\ b)$ and Q are the same size. The residual QR method may be the most expensive in terms of storage since it has to store both $(A\ b)$ and Q (as well as R).

6 Numerical Experiments

The methods discussed in this paper were implemented and tested in Pro-Matlab (Version 3.5i). This section discusses the tests and matrices used to compare the various methods.

The tests are all of the *sliding window* type. This type of test uses *windows* consisting of w rows of an $m \times n$ matrix ($m >> w > n$). A series of least-squares problems, defined by the window and the corresponding subsection of an $m \times 1$ right-hand side vector, are solved. Originally, the window is set to be the top w rows of the larger matrix, and the QR factorization is computed. At each step, k rows of the larger matrix are added at the bottom

Algorithm		Total Operations		
		Multiply	Add, Subtract	Div, Sqrt
CGSAUG,	GIVENS	$8mnk+2n^2k+2mk^2+8mk+8nk$	$6mnk+n^2k+2mk^2+6mk+6nk+k^2$	$3nk+m+7k$
CSNE,	GIVENS	$3mnk+4n^2k+3mk^2+4nk^2-mk+10nk+4k^2$	$3mnk+3n^2k+3mk^2+2nk^2+mk+7nk+2k^2$	$7nk+9k$

Table 5: Comparison of the total work involved for some methods

of the window, requiring an update of the QR factorization, and k rows are deleted from the top of the window, requiring a downdate of the factorization. The updated/downdated factorization is then used to obtain the solution to the problem corresponding to the current window position. Our tests used a window size $w = 8$.

Two of the three matrices used in the tests come from [BPE92]: the third is adapted from [Bjö87].

Matrix I. The matrix A is of size 50×5 with elements taken from a uniform probability distribution in [0,1]. Element (18,3) has been perturbed by a uniform random sample from $[0, 10^3]$. The right-hand side vector b is constructed by multiplying A by the vector $[1, 1, 1, 1, 1]^T$ and adding to each element a random sample from $[0, 10^{-6}]$.

Matrix II. The matrix A is again of size 50×5. In the first 25 rows of A, element (i, j) is $(i + j)^{-1}$: these rows are the first five columns of a 25×25 Hilbert matrix. In the bottom 25 rows, element (i, j) is the same as element $(51 - i, j)$, that is, the bottom 25 rows are the reflection of the top 25 rows about the middle of the matrix. Each element of A is perturbed by a uniform random sample from $[0, 10^{-5}]$. The right-hand side vector b is again the product of A and the vector $[1, 1, 1, 1, 1]^T$, with each element perturbed by a uniform random sample from $[0, 1]$.

Matrix III. The matrix A, again 50×5, is the product of three matrices, W, V, and D. V is the 50×5 matrix in which element (i, j) has the value $(i - 1)^{(j-1)}$. D is the diagonal matrix which normalizes each column of V. W is a matrix which weights rows 15, 20, 25, ..., 50 by a factor of 100. The right-hand side vector b is constructed by multiplying A by a vector $x = D^{-1}[10^4, 10^3, 100, 10, 1]^T$.

Figures 1, 2, and 3 show data about the condition of the three matrices. Each figure shows the condition of the window matrix for each step of the sliding window process.

The remaining figures show the performance of the methods on the three test matrices. Two types of comparisons are made. First, the rank-k CSNE and LINPACK methods are compared to the rank-one methods presented in [BPE92]. Second, the various rank-k methods are compared to each other. The Gram-Schmidt methods of Section 2 and the CSNE/LINPACK methods of Section 3 are compared as separate groups, and then the best methods from each group are compared to each other.

First, consider the rank-k CSNE and LINPACK methods. Björck, Park and Eldén [BPE92] give results which show that in the rank-one case, for tests involving ill-conditioned matrices, the CSNE method outperforms the LINPACK method. Figures 4 and 5 show that this continues to be the case for the rank-k methods. In addition, the rank-two and rank-three methods perform at least as well as the rank-one methods for the tests presented here. It is particularly interesting to note that the error in the LINPACK method for Matrix I goes down by orders of magnitude as the rank increases: the rank-two CSNE method is also much better than the rank-one CSNE method on this matrix, although the rank-three CSNE method does not improve much on the accuracy of the rank-two method.

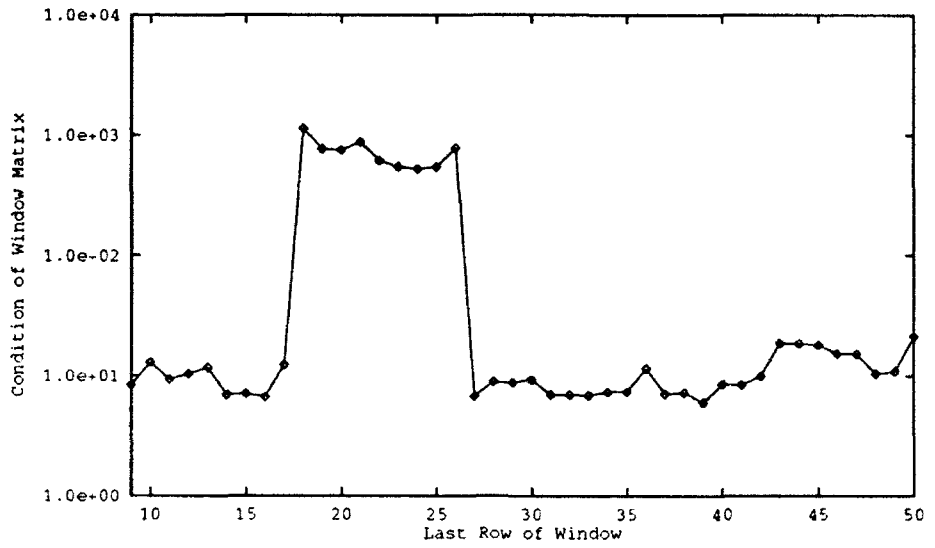


Figure 1: Condition of the Window Matrix – Matrix I

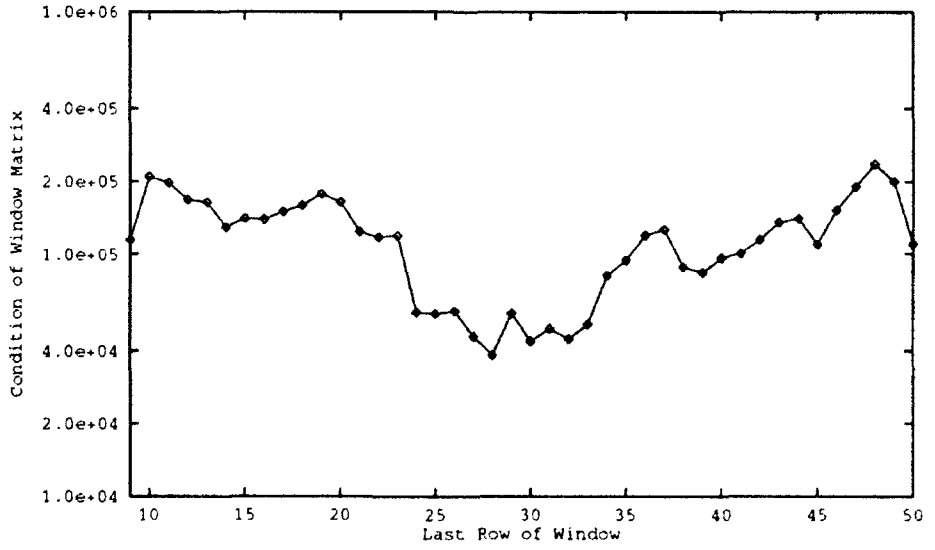


Figure 2: Condition of the Window Matrix – Matrix II

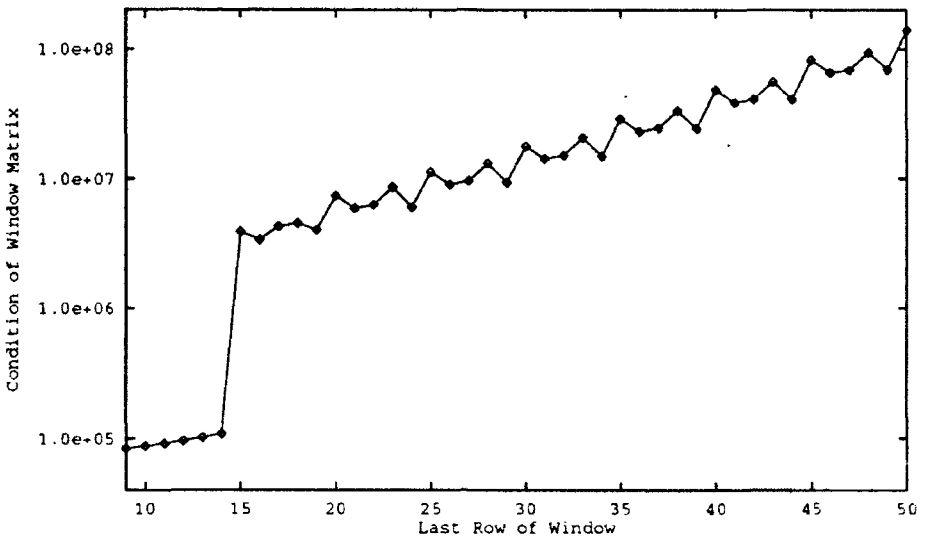


Figure 3: Condition of the Window Matrix – Matrix III

Figures 6 and 7 compare the five Gram-Schmidt based methods presented in this paper (Classical GS, Modified GS, the Small QR and Small Cholesky factorizations, and the Residual QR method). The methods are comparable on Matrix I. On Matrix II, the two "Small" factorization methods do not perform nearly as well as the others (the QR method seems to be better than the Cholesky method, as predicted). The other methods are comparable to one another and are in the same range as the CSNE method. However, it should be noted that the CGS method required one re-orthogonalization for all but one of the columns of Matrix II.

A comparison of the rank-two CSNE, MGS, and CGS methods (Figure 8) shows that these methods perform comparably on Matrix II, giving results similar to a QR decomposition. However, as noted in [Bjö87], the CSNE method has problems when dealing with matrices which are "weighted", that is, in which rows have been multiplied by a weight constant which gives one row a significantly higher norm than others around it. Matrix III was chosen because it is an example of an ill-conditioned, weighted matrix. Figure 9 shows that the CSNE method breaks down when the window includes the first weighted row. The Residual QR method also breaks down, but the CGS and MGS methods both closely approximate the results obtained by performing a full QR decomposition. The CGS method required re-orthogonalization for each column of Matrix III. From these figures, we conclude that the CGS method with re-orthogonalization is the most stable, although it is expensive in terms of computation and storage. The MGS methods is less expensive and performs comparably in all the cases shown here. The CSNE method performs nearly as well in some cases with lower storage and computation costs.

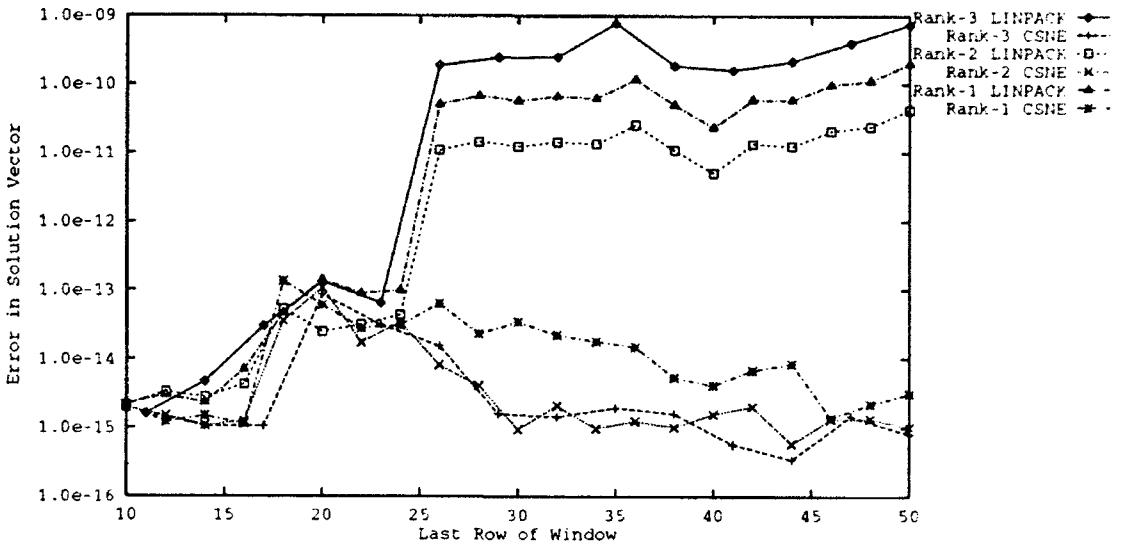


Figure 4: Rank-one vs. Rank-two LINPACK/CSNE Methods, Matrix I

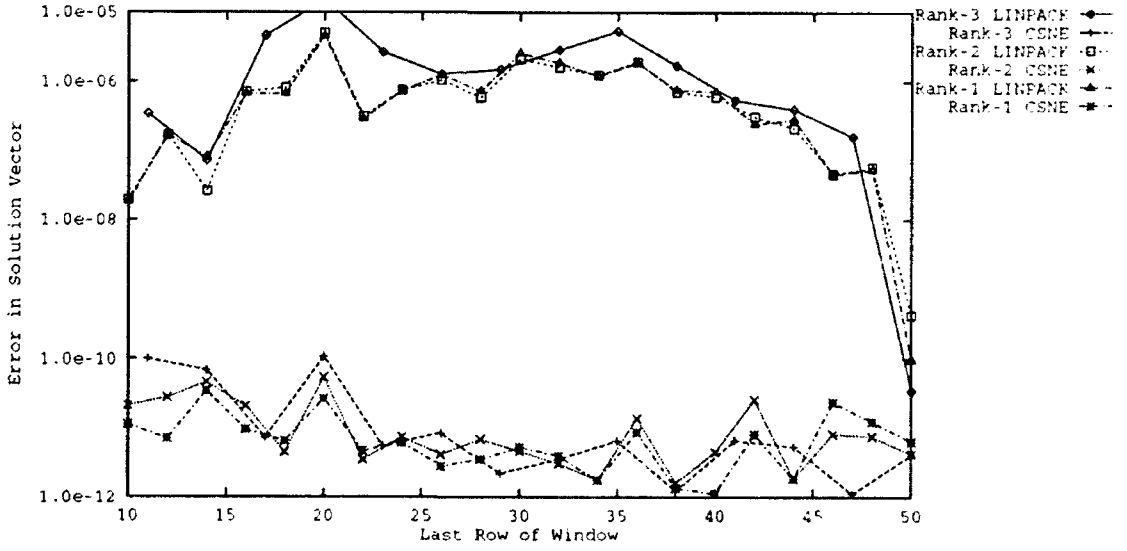


Figure 5: Rank-one vs. Rank-two LINPACK/CSNE Methods, Matrix II

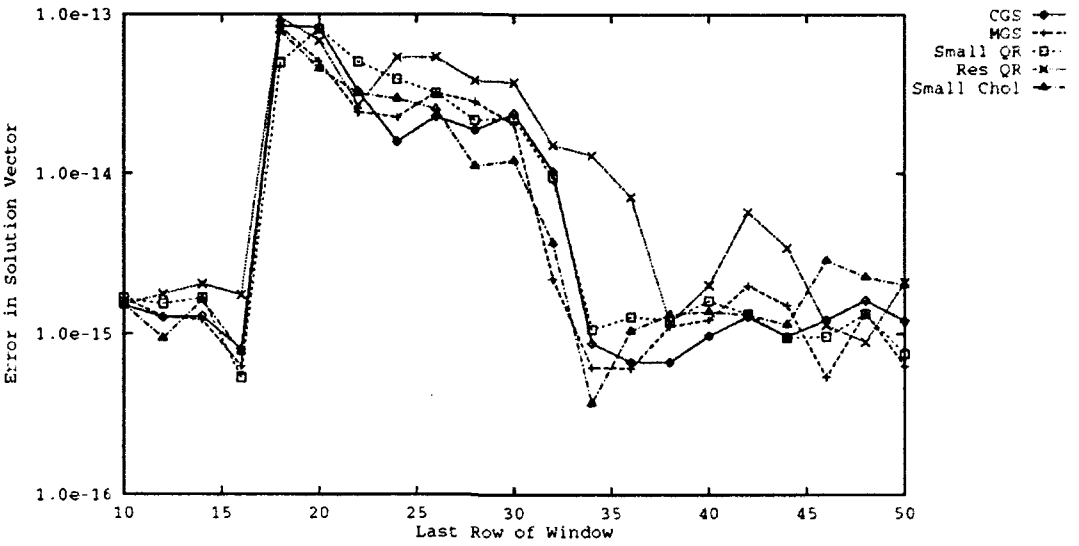


Figure 6: Rank-two Gram-Schmidt Methods, Matrix I

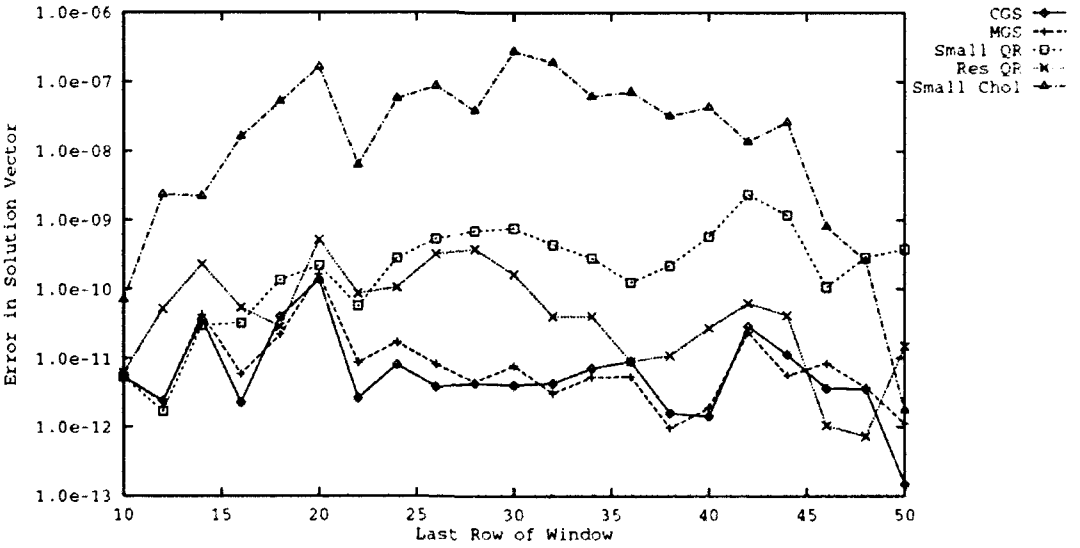


Figure 7: Rank-two Gram-Schmidt Methods, Matrix II

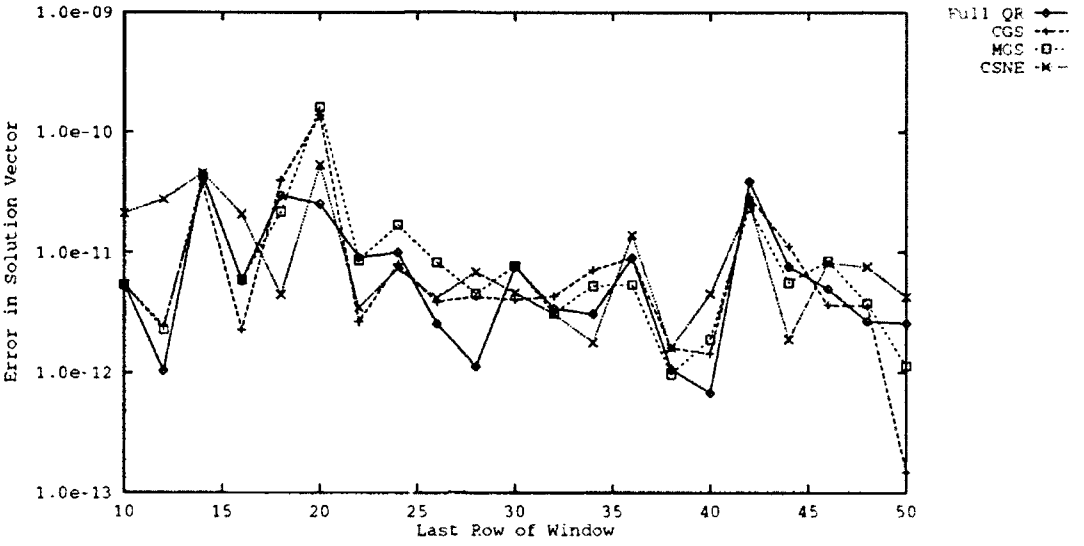


Figure 8: Rank-two CSNE and GS Methods, Matrix II

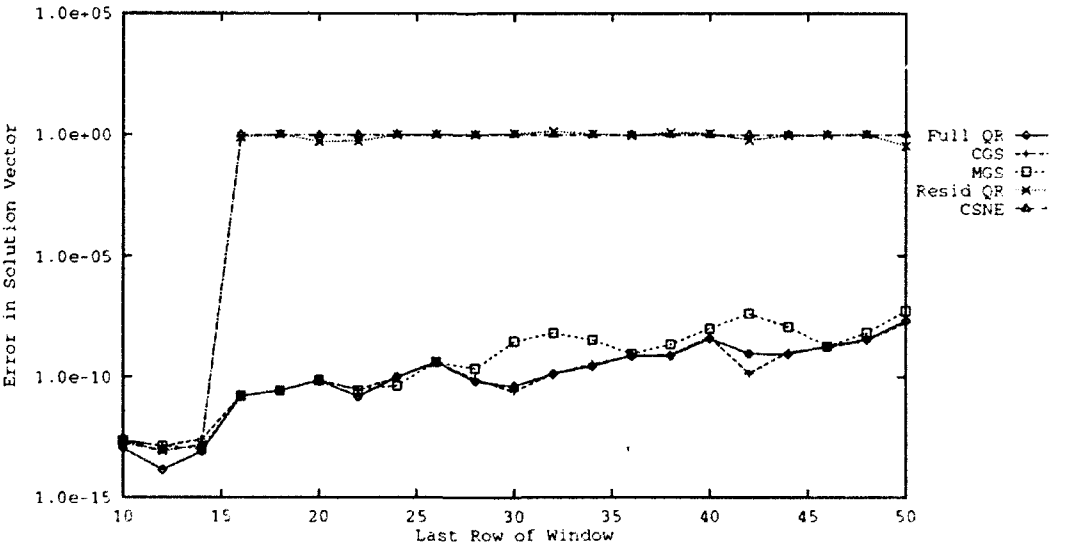


Figure 9: Rank-two CSNE and GS Methods, Matrix III

References

[Bjö87] Å. Björck. Stability analysis of the method of seminormal equations for linear least squares problems. *Linear Algebra and its Applications*, 88-89:31–48, 1987.

[BNP92] Adam W. Bojanczyk, James G. Nagy, and Robert J. Plemmons. Row householder transformations for rank-k cholesky inverse modifications. Technical Report IMA Preprint Series 978, Institute for Mathematics and Its Applications, University of Minnesota, May 1992.

[BPE92] Å. Björck, H. Park, and L. Eldén. Accurate downdating of least squares solutions. March 1992.

[BS89] Adam W. Bojanczyk and Allan O. Steinhardt. Stabilized hyperbolic Householder transformations. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 37(8):1286–1288, August 1989.

[DGKS76] J. W. Daniel, W. B. Gragg, L. Kaufman, and G. W. Stewart. Reorthogonalization and stable algorithms for updating the Gram-Schmidt QR factorization. *Mathematics of Computation*, 30(136):772–795, October 1976.

[RS86] Charles M. Rader and Allan O. Steinhardt. Hyperbolic Householder transformations. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, ASSP-34(6):1589–1602, December 1986.

Kent Conference on Parallel Scientific Computing
Kent University
April, 1992

On propagating orthogonal transformations in a product of 2×2 triangular matrices

Adam W. Bojanczyk
Cornell University, Dept. Electrical Engineering
Ithaca, NY 14853-3801

Paul Van Dooren
University of Illinois at Urbana-Champaign
Coordinated Science Laboratory
1308 W. Main Str., Urbana, IL 61801

Abstract

In this note, we propose an implicit method for applying orthogonal transformations on both sides of a product of upper triangular 2×2 matrices that preserve upper triangularity of the factors. Such problems arise in Jacobi type methods for computing the PSVD of a product of several matrices, and in ordering eigenvalues in the periodic Schur decomposition.

Introduction

The problem of computing the singular value decomposition (SVD) of a product of matrices have been considered in [1],[2], [3], [10]. The computation proceeds in two stages. In the first stage the matrices are transformed into the upper triangular forms. In the second iterative stage an implicit Jacobi-type method is applied to the triangular matrices. It is important that after each iteration the matrices stay triangular [8].

A crucial aspect in such implicit Jacobi iterations is the accurate computation of the PSVD of a product of 2×2 triangular matrices. There two conditions have to be satisfied [2]. First, one has to ensure that the orthogonal transformations applied to the triangular matrices must leave the matrices triangular, and second, that the transformations diagonalize the product accurately. It was shown in [1] and [2] that these two conditions are satisfied by a so-called *half-recursive* and *direct* method, respectively, for computing the SVD of the product of two matrices.

In this note we analyze an extension of the *half-recursive* method for computing the SVD of the product of many 2×2 triangular matrices. We also show that the extension of the *half-recursive* method can be used for swapping eigenvalues in the periodic Schur decomposition described in [4]. For simplicity we assume real matrices and real eigenvalues, but all results are easily extended to the complex case.

Criterion for numerical triangularity

Suppose we are given k, $k > 1$, upper triangular matrices A_i, $i = 1, 2..., k$,

$$A_i = \begin{pmatrix} a_i & b_i \\ 0 & d_i \end{pmatrix} .$$

We denote the product of A_i, $i = 1, 2..., k$, by A,

$$A = A_1 \cdots A_k = \begin{pmatrix} a & b \\ 0 & d \end{pmatrix} .$$

Let the orthogonal matrices Q_1 and Q_{k+1} be such that

$$A' = Q_1 A Q_{k+1}^T = \begin{pmatrix} a' & b' \\ 0 & d' \end{pmatrix} \tag{2.1}$$

is upper triangular. In case we are interested in finding the *Singular Value Decomposition* of A, one imposes the additional condition that $b' = 0$. This defines uniquely the above decomposition up to permutations that interchange the diagonal elements of A'. In case we are interested in finding the *Schur Form* of A, one imposes the additional condition that $Q_1 = Q_{k+1}$. Again, this defines uniquely the above decomposition up to the ordering of the diagonal elements of A'. In both cases the transformations Q_1 and Q_{k+1} are thus defined by the choice of ordering of diagonal elements in the resulting matrix A'. Our objective now is to find orthogonal matrices Q_j, $j = 2, 3, ..., k$, such that

$$A_i' = Q_i A_i Q_{i+1}^T = \begin{pmatrix} a_i' & b_i' \\ 0 & d_i' \end{pmatrix} \tag{2.2}$$

are meanwhile maintained in upper triangular form as well. It is easy to see that if $abd \neq 0$ then for a given pair of orthogonal transformations Q_1 and Q_{k+1} there exist unique (up to the sign) orthogonal transformations Q_2,...,Q_k such that (2.2) is satisfied. There are many mathematically equivalent strategies of determining Q_2,...,Q_k. However, as it was shown in [1], [2] and [3], some strategies may produce numerically significantly different results than other strategies. We will consider a particular method numerically acceptable if the triangular matrices after transformations have been applied to them stay numerically triangular in the sense described below.

Let $\bar{A}$ be the computed A, and let $\bar{Q}_i$, $i = 1, 2, ..., k+1$ be the computed transformations. Define

$$\tilde{A}' := \bar{Q}_1 \bar{A} \bar{Q}_{k+1}^T = \begin{pmatrix} \tilde{a}' & \tilde{b}' \\ \tilde{e}' & \tilde{d}' \end{pmatrix} \tag{2.3}$$

and

$$\tilde{A}_i' := \bar{Q}_i A_i \bar{Q}_{i+1}^T = \begin{pmatrix} \tilde{a}_i' & \tilde{b}_i' \\ \tilde{e}_i' & \tilde{d}_i' \end{pmatrix} . \tag{2.4}$$

Let ϵ denote the relative machine precision. Assume that we are given $\bar{Q}_1$ and $\bar{Q}_{k+1}$ such that

$$|\tilde{e}'| = O(\epsilon ||\bar{A}||) \tag{2.5a}$$

We will say that $\tilde{A}_i'$ is numerically triangular if

$$|\tilde{e}_i'| = O(\epsilon ||A_i||) , \tag{2.5b}$$

We will propose a method for computing nearly orthogonal $\bar{Q}_i$, $i = 2, ..., k$, for which, under a slightly stronger version of the assumption (2.5a), the (2,1) element $\tilde{e}_i'$ of $\tilde{A}_i'$ will satisfy (2.5b). Condition (2.5b) justifies truncating the (2,1) element e_i' of $\tilde{A}_i'$ to zero. Thus, $\tilde{e}'$ is also forced to zero.

The Algorithm

Our algorithm is a generalization of the algorithms presented in [1] and [3] for computing the PSVD of two and three matrices respectively. There the orthogonal transformations all had the form

$$Q = \begin{pmatrix} s & c \\ -c & s \end{pmatrix} , \tag{3.1}$$

where $c^2 + s^2 = 1$. As we will build on the results presented in those papers we retain this particular choice of orthogonal transformations. While each transformation Q_i is defined by the cosine-sine pair $c_i = \cos\theta_i$ and $s_i = \sin\theta_i$, we also associate Q_i with the tangent

$$t_i = \tan\theta_i .$$

Given t_i, we can easily recover c_i and s_i using the relations

$$c_i = \frac{1}{\sqrt{1 + t_i^2}} \quad \text{and} \quad s_i = t_i c_i . \tag{3.2}$$

Following the exposition in [1], [3], we consider the result of applying the left and right transformations Q_l (for the outer left transformation) and Q_r (for the outer right transformation) to a 2×2 upper triangular matrix A:

$$A' = Q_l A Q_r^T = \begin{pmatrix} a' & b' \\ e' & d' \end{pmatrix} = \begin{pmatrix} s_l & c_l \\ -c_l & s_l \end{pmatrix} \begin{pmatrix} a & b \\ 0 & d \end{pmatrix} \begin{pmatrix} s_r & c_r \\ -c_r & s_r \end{pmatrix}^T . \tag{3.3}$$

We can derive from (3.3) these four relations:

$$e' = c_l c_r(-a t_r + d t_l - b) , \tag{3.4a}$$

$$b' = c_l c_r(-a t_l + d t_r + b t_l t_r) , \tag{3.4b}$$

$$a' = c_l c_r(b t_l + d + a t_l t_r) , \tag{3.4c}$$

$$d' = c_l c_r(a - b t_r + d t_l t_r) , \tag{3.4d}$$

where $t_l = \tan\theta_l$ and $t_r = \tan\theta_r$.

The postulates that both e' and b' be zeros define two conditions on t_l and t_r, so that (3.3) represents an SVD of A [5]. The postulate that e' be zero and $t_l = t_r$ represent conditions for swapping eigenvalues of A.

The postulate that e' be zero defines a condition relating θ_l to θ_r, so that if one is known the other can be computed in order to reduce A' to an upper triangular form. For ease of exposition, we assume for now on that $abd \neq 0$. It implies that $c_l c_r \neq 0$, and so the postulate that $e' = 0$ in (3.4a) becomes

$$-a t_r + d t_l - b = 0 . \tag{3.5}$$

The consequence of (3.5) is that (3.4c) and (3.4d) simplify to

$$a' = c_l c_r(t_l^2 + 1) d \tag{3.6a}$$

and

$$d' = c_l c_r(t_r^2 + 1) a , \tag{3.6b}$$

respectively.

Assume that $Q_l = Q_1$ and $Q_r = Q_{k+1}$ are given, that is $t_l = t_1$ and $t_r = t_{k+1}$ are known. We will use relations of the type (3.5) with t_l and t_r as the reference tangents to compute the remaining transformations.

Our algorithm can be described recursively as follows. We split the sequence $A_1, A_2, \ldots, A_{k+1}$ into two subsequences of consecutive matrices $A_1, A_2, \ldots, A_m$ and $A_{m+1}, A_{m+2}, \ldots, A_{k+1}$ where $1 < m < k+1$. Let us denote

$$A_l \equiv \begin{pmatrix} a_l & b_l \\ 0 & d_l \end{pmatrix} = \prod_{i=1}^{m} A_i \quad \text{and} \quad A_r \equiv \begin{pmatrix} a_r & b_r \\ 0 & d_r \end{pmatrix} = \prod_{i=m+1}^{k} A_{i+1} . \tag{3.7}$$

Suppose that

$$|t_l d| \le |t_r a| .$$

Then we propose to compute t_s from the condition (3.5) by the forward substitution,

$$t_m = \frac{d_l t_l - b_l}{a_l} \tag{3.8a}$$

Otherwise, that is when

$$|t_l d| > |t_r a| ,$$

we propose to compute t_s from (3.5) by the backward substitution,

$$t_m = \frac{a_r t_r + b_r}{d_r} . \tag{3.8b}$$

Having defined the first step, the procedure can now be applied recursively to generate all the remaining orthogonal transformations Q_i, $i = 2, \ldots, k$. Note that there is a lot of freedom in splitting the sequence $A_1, A_2, \ldots, A_{k+1}$ into subsequent subsequences. This might be advantageous for a divide-and-conquer type of computation in a parallel environment.

As will be shown later, under mild conditions on Q_1 and Q_{k+1}, this particular way of generating orthogonal transformations Q_i, $i = 2, \ldots, k$, will guarantee that all A_i' will be numerically upper triangular in the sense that (2.5b) will be satisfied.

Error Analysis

In our error analysis, we adopt a convention that involves a liberal use of Greek letters. For example, by α we mean a relative perturbation of an absolute magnitude not greater than ϵ, where ϵ denotes the machine precision. All terms of order ϵ^2 or higher will be ignored in this first-order analysis.

The function $\mathrm{fl}(a)$ will denote the floating point approximation of a. For the purpose of the analysis, a "bar" denotes a computed quantity which is perturbed as the result of inexact arithmetic. For example, instead of a, b and d, we have the perturbed values $\bar{a}$, $\bar{b}$ and $\bar{d}$ which result from floating point computation of $\prod_{i=1}^{k+1} A_i$. We assume that exact arithmetic may be performed using these perturbed values. The "tilde" symbol is used to denote conceptual values computed exactly from perturbed data.

We start our procedure by computing elements of the product matrix A as the product of A_l and A_r defined by (3.7):

$$\bar{a} := \mathrm{fl}(a_l a_r) = \bar{a}_l \bar{a}_r (1 + \alpha) , \tag{3.1a}$$

$$\bar{d} := \mathrm{fl}(\bar{d}_l \bar{d}_r) = \bar{d}_l \bar{d}_r (1 + \delta) , \tag{4.1b}$$

$$\bar{b} := \mathrm{fl}(\bar{a}_l \bar{b}_r + \bar{b}_l \bar{d}_r) = \bar{a}_l \bar{b}_r (1 + 2\beta_1) + \bar{b}_l \bar{d}_l (1 + 2\beta_2) , \tag{4.1c}$$

where, according to our convention, the parameters α, δ, β_1, β_2, and β_3 are all quantities whose absolute values are bounded by ϵ.

Now we specify the condition that we impose on the computed Q_1 and Q_{k+1}.

Assumption I: Throught the rest of this note we will assume that the computed tangents $\bar{t}_l$ and $\bar{t}_r$ corresponding to the outer transformations $Q_l = Q_1$ and $Q_r = Q_{k+1}$ satisfy the following equality

$$\bar{a}(1 + C\psi)\bar{t}_r - \bar{d}(1 + C\phi)\bar{t}_l + \bar{b}(1 + C\chi) = 0 , \tag{4.2a}$$

where $C = C(k)$.

□

Lemma 4.1: The recurrence (3.8a) yields $\bar{t}_m$ such that

$$\bar{a}_l(1 + 2\psi_1)\bar{t}_m - \bar{d}_l(1 + \phi_1)\bar{t}_l + \bar{b}_l = 0 . \tag{4.3}$$

Likewise, the recurrence (3.8b) yields $\bar{t}_m$ such that

$$\bar{d}_r(1 + \phi_2)\bar{t}_m - \bar{a}_r(1 + 2\psi_2)\bar{t}_r - \bar{b}_r = 0 . \tag{4.4}$$

□

Proof. The proof easily follows from (3.8a) and (3.8b).

□

Theorem4.2: If $|\bar{t}_l \bar{d}| < |\bar{t}_r \bar{a}|$ and if $\bar{t}_m$ is computed via (3.8a) then $\bar{t}_m$ satisfies the relation

$$\bar{a}_r(1 + C_l\psi_l)\bar{t}_r - \bar{d}_r(1 + C_l\phi_l)\bar{t}_m + \bar{b}_r(1 + C_l\psi_l) = 0 \tag{4.5a}$$

where $C_l = C_l(k)$. Likewise, if $|\bar{t}_l \bar{d}| \geq |\bar{t}_r \bar{a}|$ and if $\bar{t}_m$ is computed via (3.8b) then t_m satisfies the relation

$$\bar{a}_l(1 + C_r\psi_r)\bar{t}_m - \bar{d}_l(1 + C_r\phi_r)\bar{t}_l + \bar{b}_l(1 + C_r\chi_r) = 0 \tag{4.5b}$$

where $C_r = C_r(k)$.

Proof. We give a proof of the relation (4.5a) only as the relation (4.5b) can be proved in an analogous way.

First from (4.3a)-(4.3b) we get

$$\bar{a}_l(1 + 2\psi_1)\bar{t}_m - \bar{d}_l(1 + \phi_1)\bar{t}_l + \bar{b}_l = 0 , \tag{4.6a}$$

while from Assumption I and (4.1a)-(4.1c) we have

$$\bar{a}_l\bar{a}_r(1 + \alpha + C\psi)\bar{t}_r - \bar{d}_l\bar{d}_r(1 + \delta + C\phi)\bar{t}_l +$$

$$\bar{a}_l\bar{b}_r(1 + 2\beta_1 + C\chi) + \bar{b}_l\bar{d}_r(1 + 2\beta_2 + C\gamma) = 0 . \tag{4.6b}$$

By multiplying both sides of (4.6a) by $d_r(1+2\beta_2+C\chi_l)$ and subtracting from (4.6b) we obtain

$$\bar{a}_l\{\bar{a}_r(1+\alpha+C\psi)\bar{t}_r - \bar{a}_r\left(\frac{\bar{d}_l\bar{d}_r}{\bar{a}_l\bar{a}_r}\right)(\delta+C\phi-\phi_1-2\beta_2-C\chi)\bar{t}_l +$$

$$\bar{b}_r(1+2\beta_1+C\chi) - \bar{d}_r(1+2\beta_2+C\chi+2\psi_1)\bar{t}_m\} = 0\,,$$

or, since $\bar{a}_l \neq 0$,

$$\bar{a}_r(1+\alpha+C\psi)\bar{t}_r - \bar{a}_r\bar{t}_r\left(\frac{\bar{d}\bar{t}_l}{\bar{a}\bar{t}_r}\right)(\delta+C\phi-\phi_1+2\beta_2+C\chi)+$$

$$\bar{b}_r(1+2\beta_1+C\chi) - \bar{d}_r(1+2\beta_2+2\psi_1+C\chi)\bar{t}_m = 0\,.$$

As we assumed that $|\bar{t}_l\bar{d}| < |\bar{t}_r\bar{a}|$, the above can be rewritten as

$$\bar{a}_r(1+C_l\psi_l)\bar{t}_r - \bar{d}_r(1+C_l\phi_l)\bar{t}_m + \bar{b}_r(1+C_l\chi_l) = 0 \tag{4.7}$$

where $C_l = C_l(k)$ completing the proof.

□

We now justify why the (2,1) element in the computed matrix A_i' can be set to zero. Let the cosine and sine pairs $\tilde{c}_i$ and $\tilde{s}_i$ satisfy $\bar{t}_i = \tilde{s}_i/\tilde{c}_i$, for $i = l, m, r$. From (4.2) we can derive that

$$\tilde{c}_i := \mathrm{fl}(\tilde{c}_i) = \tilde{c}_i(1+3\mu_i)\,, \tag{4.8a}$$

$$\tilde{s}_i := \mathrm{fl}(\tilde{s}_i) = \tilde{s}_i(1+4\nu_i)\,. \tag{4.8b}$$

Let $\tilde{A}_i'$ denote the exact updated matrix derived from $\bar{A}_i$, $i = l, r$, and $\tilde{c}_i$, $\tilde{s}_i$, $i = l, m, r$ that is

$$\tilde{A}_l' = \begin{pmatrix} \tilde{s}_l & \tilde{c}_l \\ -\tilde{c}_l & \tilde{s}_l \end{pmatrix}\begin{pmatrix} \bar{a}_l & \bar{b}_l \\ 0 & \bar{d}_l \end{pmatrix}\begin{pmatrix} \tilde{s}_m & -\tilde{c}_m \\ \tilde{c}_m & \tilde{s}_m \end{pmatrix}\,, \tag{4.9a}$$

and

$$\tilde{A}_r' = \begin{pmatrix} \tilde{s}_m & \tilde{c}_m \\ -\tilde{c}_m & \tilde{s}_m \end{pmatrix}\begin{pmatrix} \bar{a}_r & \bar{b}_r \\ 0 & \bar{d}_r \end{pmatrix}\begin{pmatrix} \tilde{s}_r & -\tilde{c}_r \\ \tilde{c}_r & \tilde{s}_r \end{pmatrix}\,. \tag{4.9b}$$

Our next result is a direct consequence of Theorem 4.2 and provides bounds on the elements $\tilde{e}_i'$, $i = l, r$, defined by the relations

$$\tilde{e}_l' := -\tilde{c}_l\tilde{s}_m a_l + \tilde{s}_l\tilde{c}_m d_l - \tilde{c}_l\tilde{c}_m b_l\,, \tag{5.10a}$$

$$\tilde{e}_r' := -\tilde{c}_m\tilde{s}_r a_r + \tilde{s}_m\tilde{c}_r d_r - \tilde{c}_r\tilde{c}_r b_r\,. \tag{5.10b}$$

Corollary 4.4: If $|\bar{t}_l\bar{d}| < |\bar{t}_r\bar{a}|$ and if $\bar{t}_m$ is computed via (3.8a) or if $|\bar{t}_l\bar{d}| \geq |\bar{t}_r\bar{a}|$ and if $\bar{t}_m$ is computed via (3.8b) then

$$|\tilde{e}_i'| \leq K_i\epsilon\|\bar{A}_i\|\,, \quad \text{for } i = l, r\,. \tag{4.11}$$

□

Proof. We prove the corollary for the case when $|\bar{t}_l\bar{d}| < |\bar{t}_r\bar{a}|$ and when $\bar{t}_m$ is computed via (4.8a). The other case can be proved in an analogous manner.

Using (4.3a) we can rewrite (4.10a) as

$$\tilde{e}_l' = -\bar{c}_l\bar{s}_m\bar{a}_l + \bar{s}_l\bar{c}_m\bar{d}_l - \bar{c}_l\bar{c}_m\bar{b}_l +$$

$$\bar{c}_l\bar{c}_m(\bar{a}_l(1+2\psi_1)\bar{t}_m - \bar{d}_l(1+\phi_1)\bar{t}_l + \bar{b}_l) \tag{4.12}$$

from which it follows that

$$|\tilde{e}_l'| \leq K_l\epsilon\|\bar{A}_l\| .$$

Similarly, using (4.5a) we can rewrite (4.10b) as

$$\tilde{e}_r' := -\bar{c}_m\bar{s}_r a_r + \bar{s}_m\bar{c}_r d_r - \bar{c}_r\bar{c}_r b_r +$$

$$\bar{c}_l\bar{c}_m(\bar{a}_r(1+C_l\psi_l)\bar{t}_r - \bar{d}_r(1+C_l\phi_l)\bar{t}_m + \bar{b}_r(1+C_l\psi_l)) \tag{4.13}$$

and thus

$$|\tilde{e}_r'| \leq K_r\epsilon\|\bar{A}_r\| ,$$

completing the proof of (4.10a).

□

Numerical examples

The SVD algorithms for 2×2 upper triangular matrices in [1],[2] or [5] give $\bar{t}_l$ and $\bar{t}_r$ which satisfy Assumption I. We will illustrate that by using our new scheme triangularity of te transformed factors is preserved.

Consider the case of three matrices in the product Assume that the given data matrices are

$$A_1 = \begin{pmatrix} 2.316797292247488e+00 & -1.437687878748196e-01 \\ 0 & -2.718295063593277e-02 \end{pmatrix} ,$$

$$A_2 = \begin{pmatrix} 1.222222234444442e+00 & 3.480474357220011e-01 \\ 0 & 5.674165405829751e+00 \end{pmatrix} ,$$

$$A_3 = \begin{pmatrix} 2.222222211111111e-01 & 1.732050807568877e+00 \\ 0 & 1.111111110000000e-12 \end{pmatrix} .$$

They generate the matrix product $\bar{A} := A1 \cdot A2 \cdot A3$

$$\bar{A} = \begin{pmatrix} 6.292535886949669e-01 & 4.904546363614013e+00 \\ 0 & -1.713783977472744e-13 \end{pmatrix} .$$

We are erested in computing orthogonal transformations Q_1, Q_2, Q_3 and Q_4 which satisfy (2.2) and (2. ith the (1,2) element zero. The SVD algorithm for the 2×2 upper triangular matrix $\bar{A}$ in [1] or gives $\bar{t}_1 = 3.437688760727056e-14$ and $\bar{t}_4 = -7.794228673031074e+00$ which satisfy Assumption I. In fact we have

$$\bar{Q}_1\bar{A}\bar{Q}_4 = \begin{pmatrix} -2.180909253067911e-14 & -7.494178599599612e-30 \\ 0 & 4.944748235423613e+00 \end{pmatrix}$$

We split $\bar{A}$ into the product of $A_{1,2} = A_1A_2$ and A_3. We note that the ratio

$$\frac{\bar{t}_1\bar{d}}{\bar{t}_4\bar{a}} = 1.201223412093697e-27$$

If we compute t_3 from t_1 as indicated by the ratio, and next t_2 as specified by (3.8a) or (3.8b) then Corollary 5.4 will guarantee that the transformed factors will stay (numerically) triangular. Suppose however that we compute t_3 from t_4 and next t_2 from t_3. Then Lemma 4.1 will guarantee that $Q_2A_2Q_3^T$ and $Q_3A_3Q_4^T$ will saty numerically triangular. However, for the computed $Q_1A_1Q_2^T$ we have

$$Q_1A_1Q_2^T = \begin{pmatrix} -2.713066430028558e-02 & -1.685188387402401e-03 \\ -1.360106941575845e-04 & 2.321253786046106e+00 \end{pmatrix}$$

which cannot be considered upper triangular. An error of order 10^{-4} has to be introduced to truncate the (2,1) element in $Q_1A_1Q_2^T$ so it becomes upper triangular.

Acknowledgements

Adam Bojanczyk was partially supported by the Joint Services Electronics Program (Grant F49620-90-C-0039 monitored by AFOSR). supported by the Research Board of the University of Illinois at Urbana-Champaign (Grant P 1-2-68114) and by the National Science Foundation (Grant CCR 9209349).

References

[1] G.E. Adams, A.W. Bojanczyk and F.T. Luk, "Computing the PSVD of Two 2×2 Triangular Matrices", *submitted to SIMAX.*

[2] Z. Bai and J.W. Demmel, "Computing the Generalized Singular Value Decomposition", Report No UCB/CSD 91/645, Computer Science Division, University of California, Berkeley, August 1991.

[3] A.W. Bojanczyk, L.M. Ewerbring, F.T. Luk and P. Van Dooren, "An Accurate Product SVD Algorithm", *Signal Processing*, 25 (1991), pp. 189-201.

[4] A.W. Bojanczyk, P. Van Dooren and G.H. Golub, "The periodic Schur decomposition. Algorithms and applications".

[5] J. P. Charlier, M. Vanbegin and P. Van Dooren, "On efficient implementations of Kogbetliantz's algorithm for computing the singular value decomposition," *Numer. Math.*, 52 (1988), pp. 279–300.

[6] B. L. R. De Moor and G. H. Golub, "Generalized singular value decompositions: A proposal for a standardized nomenclature," Manuscript NA-89-05, Numerical Analysis Project, Stanford University, Stanford, Calif., 1989.

[7] K. V. Fernando and S. J. Hammarling, "A product induced singular value decomposition for two matrices and balanced realisation," in *Linear Algebra in Signals, Systems and Control*, B. N. Datta et al., Eds., SIAM, Philadelphia, Penn., 1988, pp. 128–140.

[8] M. T. Heath, A. J. Laub, C. C. Paige, and R. C. Ward, "Computing the SVD of a product of two matrices," *SIAM J. Sci. Statist. Comput.*, 7 (1986), pp. 1147–1159.

[9] C. C. Paige, "Computing the generalized singular value decomposition," *SIAM J. Sci. Statist. Comput.*, 7 (1986), pp. 1126–1146.

[10] H. Zha, "A numerical algorithm for computing the restricted SVD of matrix triplets", to appear in *Linear Algebra and Its Applications.*

Proceedings of the NATO ASI Workshop on Real-time and Large Scale Computing, August 1-15, 1992, Leuven, Belgium

REORDERING DIAGONAL BLOCKS IN REAL SCHUR FORM

Adam W. Bojanczyk [1]
Cornell University, Dept. Electrical Engineering
Ithaca, NY 14853-3801

Paul Van Dooren
University of Illinois at Urbana-Champaign
Coordinated Science Laboratory
1308 W. Main Str., Urbana, IL 61801

KEYWORDS. Invariant subspaces, eigenvalues, reordering.

1. Introduction

The problem of reordering eigenvalues of a matrix in real Schur form arises in the computation of the invariant subspaces corresponding to a group of eigenvalues of the matrix. A basic step in such reordering is to swapp two neighboring 1×1 or 2×2 diagonal blocks by an orthogonal transformation. Swapping two 1×1 blocks or swapping 1×1 and 2×2 blocks are well understood [3]. Swapping two 2×2 blocks poses some numerical difficulties. Recently, Bai and Demmel [1] have proposed an algorithm for swapping two 2×2 blocks which is for all practical purposes backward stable. In this note we describe an alternative approach for swapping two 2×2 blocks which is based on an eigenvector calculation. It appears that the method guarantees small rounding errors in the (2,1) block of the transformed 4×4 matrix even if the two 2×2 blocks have almost the same eigenvalues.

2. Reordering eigenvalues

Assume that A is a 4×4 block triangular matrix,

$$A = \begin{pmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ 0 & 0 & a_{33} & a_{34} \\ 0 & 0 & a_{43} & a_{44} \end{pmatrix},$$

where A_{11} and A_{22} are 2×2 with pairs of complex conjugate eigenvalues λ_1, $\bar{\lambda}_1$ and λ_2, $\bar{\lambda}_2$. We can further assume that A_{11} and A_{22} are in the standard form,

$$A_{11} = \begin{pmatrix} \alpha_1 & \beta_1/k_1 \\ -\beta_1 k_1 & \alpha_1 \end{pmatrix} \quad \text{and} \quad A_{22} = \begin{pmatrix} \alpha_2 & \beta_2/k_2 \\ -\beta_2 k_2 & \alpha_2 \end{pmatrix}.$$

[1] Research supported in part by the Joint Services Electronics Program, contract no. F49620-90-C-0039.

We want to find an orthogonal transformation Q such that

$$\hat{A} \equiv QAQ^T = \begin{pmatrix} \hat{A}_{22} & \hat{A}_{12} \\ 0 & \hat{A}_{11} \end{pmatrix},$$

where $\hat{A}_{11}$ and $\hat{A}_{22}$ are similar to A_{11} and A_{22} respectively.

The standard form implies that $\lambda_2 = \alpha_2 + \beta_2 \cdot \mathrm{i}$ is the eigenvalue of A_{22}. Thus $A(\lambda_2) = A - \lambda_2 \cdot I$ is singular as its (2,2) diagonal block has rank 1. Now one can find a sequence of complex Givens rotations such that

$$\begin{pmatrix} a_{11}-\lambda_2 & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22}-\lambda_2 & a_{23} & a_{24} \\ 0 & 0 & a_{33}-\lambda_2 & a_{34} \\ 0 & 0 & a_{43} & a_{44}-\lambda_2 \end{pmatrix} G_{34}^{(1)} G_{12}^{(2)} G_{23}^{(3)} G_{12}^{(4)} = \begin{pmatrix} 0^{(4)} & \tilde{a}_{12} & \tilde{a}_{13} & \tilde{a}_{14} \\ 0^{(2)} & 0^{(3)} & \tilde{a}_{23} & \tilde{a}_{24} \\ 0 & 0 & 0^{(1)} & \tilde{a}_{34} \\ 0 & 0 & 0^{(1)} & \tilde{a}_{44} \end{pmatrix}$$

where $G_{ij}^{(k)}$ denotes a complex Givens rotation operating in the plane (i,j) introducing zero at the position marked as (k) on the right hand side of the relation. Let $G = G_{34}^{(1)} G_{12}^{(2)} G_{23}^{(3)} G_{12}^{(4)}$. Then $y = u + v \cdot \mathrm{i} = Ge_1$, where $u = [u_1, u_2, u_3, u_4]^T$ and $v = [v_1, v_2, v_3, v_4]^T$ are real vectors, is the complex eigenvector corresponding to λ_2. Hence

$$\begin{pmatrix} A_{11} & A_{12} \\ 0 & A_{22} \end{pmatrix} (u \;\; v) = (u \;\; v) \begin{pmatrix} \alpha_2 & \beta_2 \\ -\beta_2 & \alpha_2 \end{pmatrix}.$$

Moreover, because A_{22} is assumed to be in a standard form, $u_4 = v_3 = 0$. The similarity transformation Q can be expressed now as a product of real Givens rotations which triangularizes the matrix $[u\ v]$. More precisely, let $Q = J_{23}^{(4)} J_{34}^{(3)} J_{12}^{(2)} J_{23}^{(1)}$ be such that

$$J_{23}^{(4)} J_{34}^{(3)} J_{12}^{(2)} J_{23}^{(1)} \begin{pmatrix} u_1 & v_1 \\ u_2 & v_2 \\ u_3 & 0 \\ 0 & v_4 \end{pmatrix} = \begin{pmatrix} \tilde{u}_1 & \tilde{v}_1 \\ 0^{(2)} & \tilde{v}_2 \\ 0^{(1)} & 0^{(4)} \\ 0 & 0^{(3)} \end{pmatrix},$$

where $J_{ij}^{(k)}$ denotes the corresponding rotation. Then Q is the desired similarity transformation.

Numerous numerical tests suggest that in the presence of rounding errors the relative error in the (2,1) block of the transformed matrix $\hat{A}$ is proportional to the machine precision. The algorithm can be extended to cover the case of swapping diagonal blocks in the periodic Schur form [2].

References

[1] Z. Bai and J.W. Demmel, "On swapping diagonal blocks in real Schur form", Technical Report, IMA, University of Minnesota, 1992.

[2] A. Bojanczyk, G. Golub, P. Van Dooren, The periodic Schur form. Algorithms and Applications, SCCM Intern. Rept. NA-92-07, Stanford University, August 1992.

[3] P. Van Dooren, A generalized eigenvalue approach for solving Riccati equations, *SIAM Sci. & Stat. Comp.* **2** (1981) 121-135.

Proceedings of the SPIE
July 19-24, 1992
San Diego, CA

The periodic Schur decomposition. Algorithms and applications

Adam Bojanczyk
Cornell University, Dept. Electrical Engineering
Ithaca, NY 14853-3801

Gene Golub
Stanford University, Dept. Computer Science
Stanford, CA 94305

Paul Van Dooren
University of Illinois at Urbana-Champaign, Coordinated Science Laboratory
1308 W. Main Str., Urbana, IL 61801

Abstract.

In this paper we derive a unitary eigendecomposition for a sequence of matrices which we call the *periodic Schur decomposition*. We prove its existence and discuss its application to the solution of periodic difference equations arising in control. We show how the classical QR algorithm can be extended to provide a stable algorithm for computing this generalized decomposition. We apply the decomposition also to cyclic matrices and two point boundary value problems.

Key words. Numerical algorithms, linear algebra, periodic systems, K-cyclic matrices, two-point boundary value problems

1 Introduction

In the study of time-varying control systems in (generalized) state space form :

$$\begin{cases} E_k \cdot z_{k+1} &= F_k \cdot z_k + G_k \cdot u_k \\ y_k &= H_k \cdot z_k + J_k \cdot u_k \end{cases} \tag{1}$$

the *periodic coefficients* case has always been considered the simplest extension of the time-invariant case. Here the coefficients satisfy, for some $K > 0$ the periodicity conditions $E_k = E_{k+K}$, $F_k = F_{k+K}$, $G_k = G_{k+K}$, $H_k = H_{k+K}$, $J_k = J_{k+K}$. The last few years there has been a renewed interest in the area because such systems arise naturally in multi-rate sampling of continuous time systems [1]. Several papers were devoted to the algebraic structure of periodic discrete time systems and it appears that a lot of the algebra indeed carries over from the time-invariant case [9]. For period $K = 1$ one has the time invariant case $E_k = E$, $F_k = F$, $G_k = G$, $H_k = H$, $J_k = J$, and it is well-known that the generalized eigenvalues of particular pencils derived from these matrices then determine the behaviour of these difference equations [13]. In the case $K > 1$ one can derive a set of K time-invariant subsampled systems [2], [9] that describe the behaviour of the periodic system. Problems of pole placement, optimal control and robust control can then be solved via these K subsampled systems.

During the last few decades linear algebra has played an important role in advances being made in the area of systems and control [16]. The most profound impact has been in the computational and implementational aspects, where numerical linear algebraic algorithms have strongly influenced the ways in which problems are being solved. The most reliable numerical linear algebra methods proposed for particular control problems are related to particular eigenvalue and singular value decompositions of "special" matrices, such as special Schur decompositions for solving Riccati equations [10], [14]. Here we present a new decomposition called the *periodic Schur form* that has important applications in control theoretic problems of periodic systems. We present a few of these applications and predict that several other uses will be found.

The decomposition has also a direct application to K-cyclic matrices and pencils, which occur in the study of Markov chains and the solution of two point boundary value problems. We show how the periodic Schur form naturally decomposes the underlying $n \times n$ matrix problem into n scalar poblems with the same structure. This can then directly be used for the solution of Markov chains and two point boundary value problems in an elegant manner. The relation with K-cyclic pencils also allows to completely characterize the singular matrix case and give conditions for the existence of solutions in the singular case.

2 Periodic Schur decomposition

Consider the set of (homogenous) difference equations

$$B_i \cdot x_{i+1} = A_i \cdot x_i, \quad i = 1, \ldots \tag{2}$$

with *periodic coefficients* $A_i = A_{i+K}$, $B_i = B_{i+K}$. For period $K = 1$ one has the constant coefficient case $A_i = A$, $B_i = B$ and it is well-known that the generalized eigenvalues of the pair A, B yield important information about the system (2). When $K > 1$ one derives from (2) a set of K time invariant systems which describe completely the behavior of (2). For simplicity we first assume all B_i to be invertible. Then define the matrices $S_i = B_i^{-1} A_i$ yielding the system :

$$x_{i+1} = B_i^{-1} A_i \cdot x_i = S_i \cdot x_i, \quad i = 1, \ldots \tag{3}$$

which is an explicit system of difference equations in x_i, again with periodic coefficients $S_i = S_{i+K}$.

One can now consider *subsampled systems* which describe the evolution of (3) over K steps, and since the coefficient matrices of (3) are K-periodic, one may expect these subsampled systems to be *time invariant*. Indeed, defining the matrices

$$S^{(k)} = S_{k+K-1} \cdot \ldots S_{k+1} \cdot S_k, \; k = 1, \ldots, K. \tag{4}$$

then one obtains from (3), (4) the set of K *subsampled systems* :

$$\begin{aligned} x_{1+(i+1)K} &= S^{(1)} \cdot x_{1+iK}, \quad i = 0, 1, 2, \ldots \\ x_{2+(i+1)K} &= S^{(2)} \cdot x_{2+iK}, \quad i = 0, 1, 2, \ldots \\ &\vdots \\ x_{K+(i+1)K} &= S^{(K)} \cdot x_{K+iK}, \quad i = 0, 1, 2, \ldots \end{aligned} \tag{5}$$

One easily checks that the above set of difference equations, initialized with the vectors $x_i, i = 1, \ldots, K$ yields the same solution as (3). In order to describe the behaviour of these systems one thus requires the eigenvalues and eigenvectors of the *periodic matrix products* $S^{(k)}$. It is known from similar decompositions [11], [4], that explicitly forming the matrices $S^{(k)}$ ought to be avoided if possible. An implicit decomposition of these matrices is now obtained in the following theorem.

Theorem 1 *Let the matrices A_i, B_i, $i = 1, \ldots, K$ be all $n \times n$ and complex. Then there exist unitary matrices Q_i, Z_i, $i = 1, \ldots, K$ such that :*

$$\begin{array}{llll} \hat{B}_1 & = Z_1^* \cdot B_1 \cdot Q_2 & \hat{A}_1 & = Z_1^* \cdot A_1 \cdot Q_1 \\ \hat{B}_2 & = Z_2^* \cdot B_2 \cdot Q_3 & \hat{A}_2 & = Z_2^* \cdot A_2 \cdot Q_2 \\ & \vdots & & \\ \hat{B}_{K-1} & = Z_{K-1}^* \cdot B_{K-1} \cdot Q_K & \hat{A}_{K-1} & = Z_{K-1}^* \cdot A_{K-1} \cdot Q_{K-1} \\ \hat{B}_K & = Z_K^* \cdot B_K \cdot Q_1 & \hat{A}_K & = Z_K^* \cdot A_K \cdot Q_K \end{array} \tag{6}$$

where now all matrices $\hat{B}_i$, $\hat{A}_i$ are upper triangular. Moreover if the matrices B_i are invertible then each Q_i puts the matrix $S^{(i)}$ in upper Schur form, i.e. $Q_i^* S^{(i)} Q_i$ is upper triangular.

Proof : Because of its simplicity and constructive derivation, we give here a simple proof assuming all matrices A_i and B_i are non-singular, except possibly A_1. The more complex case of singular matrices is proven in section 3.2.

If all matrices B_i are invertible then all matrices $S^{(i)}$ exist. Compute the upper Schur form of $S^{(1)}$:

$$Q_1^* S^{(1)} Q_1 = \hat{S}^{(1)}.$$

This defines the matrix Q_1 and one can thus consider the matrix $B_K \cdot Q_1$ and its QR decomposition :

$$Z_K \cdot \hat{B}_K = [B_K Q_1]$$

which defines the unitary factor Z_K and upper-triangular factor $\hat{B}_K$. In turn, one then considers the matrix $Z_K^* \cdot A_K$ and its RQ decomposition (i.e. dual to the QR decomposition) :

$$\hat{A}_K \cdot Q_K^* = [Z_K^* A_K]$$

which defines the unitary factor Q_K and upper-triangular factor $\hat{A}_K$. Repeating this for all subsequent matrices defines :

- Z_i and $\hat{B}_i$ from the QR factorization of $B_i \cdot Q_{i+1}$ for $i = K, \ldots, 1$ and
- Q_i and $\hat{A}_i$ from the RQ factorization of $Z_i^* \cdot A_i$ for $i = K, \ldots, 2$.

Notice that each of these decompositions in fact corresponds to one of the equations in (6), starting from bottom to top. By now all transformation matrices Q_i and Z_i are defined but we have not proved that the last matrix $\hat{A}_1$ is upper-triangular, since in the equation

$$\hat{A}_1 = Z_1^* \cdot A_1 \cdot Q_1$$

the matrix Q_1 was already defined. But consider now the product

$$Q_1^* S^{(1)} Q_1 = [Q_1^* B_K^{-1} Z_K][Z_K^* A_K Q_K] \cdots [Q_3^* B_2^{-1} Z_2][Z_2^* A_2 Q_2][Q_2^* B_1^{-1} Z_1][Z_1^* A_1 Q_1] \tag{7}$$

or

$$\hat{S}^{(1)} = \hat{B}_K^{-1} \hat{A}_K \cdots \hat{B}_2^{-1} \hat{A}_2 \hat{B}_1^{-1} [Z_1^* A_1 Q_1]. \tag{8}$$

Now since all "hat" matrices in both sides of equation (8) are upper-triangular and invertible, this must also hold for the matrix $\hat{A}_1 = Z_1^* A_1 Q_1$. This completes the constructive proof of the existence of (6).

Notice that the proof shows how to derive all matrices Q_i and Z_i from just one of them. Moreover, by periodically interchanging the products in (7) one easily sees that also

$$Q_i^* S^{(i)} Q_i = \hat{S}^{(i)} = \hat{B}_{i-1}^{-1} \hat{A}_{i-1} \cdots \hat{B}_1^{-1} \hat{A}_1 \hat{B}_K^{-1} \hat{A}_K \cdots \hat{B}_{i+1}^{-1} \hat{A}_{i+1} \hat{B}_i^{-1} \hat{A}_i \tag{9}$$

is upper triangular and hence a Schur decomposition. So all Schur forms are actually dependent on one another via (6). ■

Corollary 1 *Let the matrices A_i, B_i, $i = 1, \ldots, K$ be all $n \times n$ and real. Then there exist orthogonal matrices Q_i, Z_i, $i = 1, ..., K$ such that the above decomposition (6) holds and all but one of the matrices $\hat{B}_i$, $\hat{A}_i$ are upper triangular. This last one is in quasi-upper triangular form with 1×1 and 2×2 diagonal blocks.*

Proof : Assume that all matrices are invertible except, say, A_1 (see section 3.2 for the general case). The proof then goes as before. Pick a *real* transformation Q_1 that puts $S^{(1)}$ in *real* Schur form $\hat{S}^{(1)} = Q_1^T S^{(1)} Q_1$. Then perform all QR factorizations as above to define the remaining transformation matrices Z_i, $i = K..., 1$ and Q_i, $i = K, ..., 2$ in decreasing order (these are *real* transformations, of course). In (8) $\hat{B}_K$, $i = K, ..., 1$, $\hat{A}_K$, $i = K, ..., 2$ (and their inverses) are upper triangular, and $\hat{S}^{(1)}$ is quasi upper-triangular. From this it follows that $\hat{A}_1$ must be of the same form as $\hat{S}^{(1)}$. If one would have started the definition of the transformations Z_i and Q_i from the other side (i.e. the QR factorization of $A_1 Q_1$ instead of $B_K Q_1$) then $\hat{B}_K$ (and its inverse) would have the same form as $\hat{S}^{(1)}$. Finally, by starting the above reasoning with a different index i it is clear that one can pick any matrix $\hat{A}_i$ or $\hat{B}_i$ to have the quasi-triangular shape. It is easy to move it around as well via a "post-processing" using updating Givens rotations. ■

In fact the matrices Q_i transform the vectors x_i to $\hat{x}_i = Q_i^* \cdot x_i$ and the difference equations (2) to the *equivalent system* :

$$Z_i^* B_i Q_{i+1} \cdot Q_{i+1}^* x_{i+1} = Z_i^* A_i Q_i \cdot Q_i^* x_i, \quad i = 1, \ldots \tag{10}$$

or

$$\hat{B}_i \cdot \hat{x}_{i+1} = \hat{A}_i \cdot \hat{x}_i, \quad i = 1, \ldots \tag{11}$$

with *periodic coefficients* $\hat{A}_i = \hat{A}_{i+K}$, $\hat{B}_i = \hat{B}_{i+K}$ which are now all *upper triangular* (except one quasi triangular one in the real case). The same transformations can of course be applied to the non-homogenous case, and this will be used later on.

An elegant consequence of the above theorem is the following corollary.

Corollary 2 *All periodic products $S^{(i)}$ have equal eigenvalues and their Schur forms $\hat{S}^{(i)}$ given by the implicit decomposition (6) have the same eigenvalues on diagonal.*

Proof : It is trivially seen that $S^{(i)}$ and $S^{(1)}$ have equal eigenvalues since

$$S^{(i)} = M_1 M_2, \quad S^{(1)} = M_2 M_1$$

with

$$M_2 = S_K \cdot \ldots \cdot S_i, \quad M_1 = S_{i-1} \cdot \ldots \cdot S_1.$$

Equality of spectrum indeed follows immediately from this. The Schur forms of the matrices $S^{(i)}$ will thus have the same diagonal elements, *up to their ordering.* But the Schur forms constructed by (6) have the additional property that the diagonal elements of the $\hat{S}^{(i)}$ matrices are all actually equal. Indeed, they are the products of the diagonal elements of the upper triangular matrices $\hat{B}_i^{-1} \hat{A}_i$. So, if one matrix $\hat{S}^{(i)}$ has a particular ordering of eigenvalues then all other matrices $\hat{S}^{(j)}$ have the same ordering of eigenvalues. ■

We give in the next section an algorithm to compute the above decomposition *implicitly*, i.e. without ever forming the products $S^{(i)}$. Moreover we show how to reorder the eigenvalues of these Schur forms. We call this the *periodic QR algorithm* as related to the above periodic Schur decomposition.

3 Periodic QR algorithm

We now consider the computation of the periodic Schur decomposition. Here we will not require the invertibility of the matrices A_i, B_i. In order to have a periodic QR algorithm we need the following ingredients to make the algorithm work :

1. a reduction to some kind of Hessenberg form
2. a direct deflation of the singular case
3. a shift calculation procedure
4. a method for performing QR steps
5. a procedure for reordering eigenvalues.

In the above list one should try to do as much as possible implicitly, i.e. without ever *constructing* the products $S^{(i)}$. Moreover one would like the total complexity of the algorithm to be comparable to the cost of K Schur decompositions, since this is what we implicitly compute. This means that the complexity should be $O(Kn^3)$ for the whole process. Notice that this indeed precludes the construction of the products $S^{(i)}$ since this would already require $O(K^2n^3)$ operations. We now derive such implicit solutions for each item. Below $\mathcal{H}(i,j)$ denotes the group of *Householder* transformations whereby (i,j) is the range of rows/columns they operate on. Similarly $\mathcal{G}(i,i+1)$ denotes the group of *Givens* transformations operating on rows/columns i and $i+1$.

3.1 Hessenberg-triangular reduction

We first consider the case where all B_i are the identity. We thus only have a product of matrices A_i and in order to illustrate the procedure we show its evolution on a product of 3 matrices only, i.e. $A_3A_2A_1$. Below is a sequence of "snapshots" of the evolution of the Hessenberg-triangular reduction. Each snapshot indicates the pattern of zeros ('0') and nonzeros ('x') in the three matrices.

First perform a Householder transformation $Q_3 \in \mathcal{H}(1,n)$ on the rows of A_2 and the columns of A_3. Choose Q_3 to annihilate all but one element in the first column of A_2 :

$$\begin{bmatrix} x & x & x & x & x & x \\ x & x & x & x & x & x \\ x & x & x & x & x & x \\ x & x & x & x & x & x \\ x & x & x & x & x & x \\ x & x & x & x & x & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ x & x & x & x & x & x \\ x & x & x & x & x & x \\ x & x & x & x & x & x \\ x & x & x & x & x & x \\ x & x & x & x & x & x \end{bmatrix}.$$

Then perform a Householder transformation $Q_1 \in \mathcal{H}(1,n)$ on the rows of A_3 and the columns of A_1. Choose Q_1 to annihilate all but one element in the first column of A_3 :

$$\begin{bmatrix} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ x & x & x & x & x & x \\ x & x & x & x & x & x \\ x & x & x & x & x & x \\ x & x & x & x & x & x \\ x & x & x & x & x & x \end{bmatrix}.$$

Then perform a Householder transformation $Q_2 \in \mathcal{H}(2,n)$ on the rows of A_1 and the columns of A_2. Choose Q_2 to annihilate all but two element in the first column of A_1 :

$$\begin{bmatrix} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \end{bmatrix}.$$

Notice that this third transformation did not destroy any of the previously created elements in A_2 because it did not transform its first column. A similar set of three transformations yields the following three snapshots :

$$\begin{bmatrix} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & x & x & x & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \end{bmatrix}$$

$$\begin{bmatrix} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & x & x & x & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & x & x & x & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & x & x & x & x & x \end{bmatrix}$$

$$\begin{bmatrix} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & x & x & x & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & x & x & x & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & x & x & x & x \end{bmatrix},$$

and this continues until we reach the Hessenberg-triangular form :

$$\begin{bmatrix} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x \end{bmatrix}.$$

When the matrices B_i are not the identity matrix, one starts with transforming each of them to triangular form. Then one proceeds with a similar reduction procedure for the matrices A_i as above. While the zero elements are being created in the matrices A_i one preserves the matrices B_i in upper triangular form at each step. Therefore, one can not make use of Householder transformations anymore. Indeed, applying a Householder transformation in $\mathcal{H}(k,n)$ (left or right) to a triangular matrix B_i fills it in and one can not find a Householder transformation in the same class operating on the other side of B_i, that will restore its triangular shape. On the other hand, this is easily done when using a Givens transformation in $\mathcal{G}(k,k+1)$ since then only the element $B_i(k+1,k)$ fills in below the diagonal and this can immediately be annihilated again using another Givens transformation in $\mathcal{G}(k,k+1)$ operating on the other side of

B_i. The above procedure of creating zeros in A_i, while maintaining the matrices B_i in upper triangular form, can thus go through. Notice that for the case $K = 1$ one retrieves *exactly* the Hessenberg-triangular reduction of the QZ algorithm [11]. Operation counts for this Hessenberg-triangular reduction are given in section 5.1.

3.2 Direct deflation of the singular case

In this section we show how to perform *direct deflations* in the Hessenberg-triangular form when either of the *pivot elements* is zero. With pivot element we mean the elements on the diagonal of each triangular matrix A_i, $i = 2, ..., K$, B_i, $i = 1, ..., K$ and below the diagonal in the Hessenberg matrix A_1. Below we treat three different cases and show how direct deflations can be performed to yield one or several subproblems of smaller dimensions where now all pivot elements are nonzero. This corresponds to subproblems without eigenvalues at zero or ∞.

Case 1. When an element below the diagonal of A_1 is zero, the problem trivially decomposes in two lower dimensional problems, as shown below for matrices B_2, A_2, B_1, A_1 where the $(4,3)$ element in A_1 is zero :

$$\left[\begin{array}{ccc|ccc} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ \hline 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & x \end{array}\right] \left[\begin{array}{ccc|ccc} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ \hline 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & x \end{array}\right] \left[\begin{array}{ccc|ccc} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ \hline 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & x \end{array}\right] \left[\begin{array}{ccc|ccc} x & x & x & x & x & x \\ x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ \hline 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x \end{array}\right].$$

This reduction is identical to what happens in the single matrix case and clearly can be repeated until one obtains smaller dimensional matrices A_1 with non-zero subdiagonals (i.e. unreduced Hessenberg forms). Moreover the reduction does not involve any transformation but only a partitioning. The next two cases are zero diagonal elements in any of the remaining matrices. One first deflates the zeros in the first matrix in the sequence $B_2, A_3, B_3, ..., A_K, B_K$, i.e. one first treats the "closest" matrix to A_1.

Case 2. If the closest matrix to A_1 with zero diagonal elements is A_i, then the partial product $A_i B_{i-1}^{-1} A_{i-1} ... B_1^{-1} A_1$ again decomposes in a block diagonal matrix, as indicated below with the sequence $A_2 B_1^{-1} A_1$ where A_2 has a zero diagonal in position $(4,4)$:

$$\left[\begin{array}{cccc|ccc} x & x & x & x & x & x & x \\ 0 & x & x & x & x & x & x \\ 0 & 0 & x & x & x & x & x \\ \hline 0 & 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & 0 & x \end{array}\right] \left[\begin{array}{cccc|ccc} x & x & x & x & x & x & x \\ 0 & x & x & x & x & x & x \\ 0 & 0 & x & x & x & x & x \\ 0 & 0 & 0 & x & x & x & x \\ \hline 0 & 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & 0 & x \end{array}\right]^{-1} \left[\begin{array}{ccc|cccc} x & x & x & x & x & x & x \\ x & x & x & x & x & x & x \\ 0 & x & x & x & x & x & x \\ 0 & 0 & x & x & x & x & x \\ \hline 0 & 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & 0 & x & x \end{array}\right]$$

$$= \left[\begin{array}{ccc|cccc} x & x & x & x & x & x & x \\ x & x & x & x & x & x & x \\ 0 & x & x & x & x & x & x \\ \hline 0 & 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & 0 & x & x \end{array}\right].$$

Moreover the bottom block is rank 3 only and one ought to be able to extract a zero eigenvalue. We now show how a sequence of Givens transformations can be generated to obtain a deflated and decomposed form of the type :

$$\ldots \left[\begin{array}{ccc|ccc} x & x & x & x & x & x & x \\ 0 & x & x & x & x & x & x \\ 0 & 0 & x & x & x & x & x \\ \hline 0 & 0 & 0 & 0 & x & x & x \\ \hline 0 & 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & 0 & x \end{array}\right] \left[\begin{array}{ccc|ccc} x & x & x & x & x & x & x \\ 0 & x & x & x & x & x & x \\ 0 & 0 & x & x & x & x & x \\ \hline 0 & 0 & 0 & x & x & x & x \\ \hline 0 & 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & 0 & x \end{array}\right] \left[\begin{array}{ccc|ccc} x & x & x & x & x & x & x \\ x & x & x & x & x & x & x \\ 0 & x & x & x & x & x & x \\ \hline 0 & 0 & 0 & x & x & x & x \\ \hline 0 & 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & 0 & x & x \end{array}\right]$$

We first apply the row transformation $Z_1^* = G_3.G_2.G_1$ to A_1, where the Givens transformations $G_1 \in \mathcal{G}(1,2)$, $G_2 \in \mathcal{G}(2,3)$ and $G_3 \in \mathcal{G}(3,4)$ are chosen to annihilate the elements 0_1, 0_2 and 0_3, respectively, as given below. Propagating these through the intermediate triangular matrices (here only B_1) this results in the column transformation $Q_2 = G_3.G_4.G_5$ applied to A_2, where the Givens transformations $G_4 \in \mathcal{G}(1,2)$ and $G_5 \in \mathcal{G}(2,3)$ respectively create the nonzero elements x_4 and x_5 ($G_6 \in \mathcal{G}(3,4)$ does not create any element) :

$$\ldots \left[\begin{array}{ccc|ccc} x & x & x & x & x & x & x \\ x_4 & x & x & x & x & x & x \\ 0 & x_5 & x & x & x & x & x \\ \hline 0 & 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & 0 & x \end{array}\right] \left[\begin{array}{ccc|ccc} x & x & x & x & x & x & x \\ 0 & x & x & x & x & x & x \\ 0 & 0 & x & x & x & x & x \\ 0 & 0 & 0 & x & x & x & x \\ \hline 0 & 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & 0 & x \end{array}\right] \left[\begin{array}{ccc|ccc} x & x & x & x & x & x & x \\ 0_1 & x & x & x & x & x & x \\ 0 & 0_2 & x & x & x & x & x \\ 0 & 0 & 0_3 & x & x & x & x \\ \hline 0 & 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & 0 & x & x \end{array}\right]$$

Then the two elements x_4 and x_5 are *annihilated again by Givens transformations* $G_7 \in \mathcal{G}(1,2)$ and $G_8 \in \mathcal{G}(2,3)$ as part of the row transformation $Z_2^* = G_8.G_7$ acting on A_2 (this yields 0_7 and 0_8, respectively). Propagating these through the intermediate triangular matrices left of A_2 and then back to A_1, this results in the column transformation $Q_1 = G_9.G_{10}$ acting on A_1. Here the Givens transformations $G_9 \in \mathcal{G}(1,2)$ and $G_{10} \in \mathcal{G}(2,3)$ create the elements x_9 and x_{10}, respectively :

$$\ldots \left[\begin{array}{ccc|ccc} x & x & x & x & x & x & x \\ 0_7 & x & x & x & x & x & x \\ 0 & 0_8 & x & x & x & x & x \\ \hline 0 & 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & 0 & x \end{array}\right] \left[\begin{array}{ccc|ccc} x & x & x & x & x & x & x \\ 0 & x & x & x & x & x & x \\ 0 & 0 & x & x & x & x & x \\ \hline 0 & 0 & 0 & x & x & x & x \\ \hline 0 & 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & 0 & x \end{array}\right] \left[\begin{array}{ccc|ccc} x & x & x & x & x & x & x \\ x_9 & x & x & x & x & x & x \\ 0 & x_{10} & x & x & x & x & x \\ \hline 0 & 0 & 0 & x & x & x & x \\ \hline 0 & 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & 0 & x & x \end{array}\right]$$

This subsequence of matrices is now already closer to the desired result. The next steps are dual to the ones above and are just indicated below by the sequence of annihilated and created elements. Just as above, everything is done via appropriate Givens rotations :

$$\ldots \left[\begin{array}{ccc|ccc} x & x & x & x & x & x & x \\ 0 & x & x & x & x & x & x \\ 0 & 0 & x & x & x & x & x \\ \hline 0 & 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x_{15} & x & x \\ 0 & 0 & 0 & 0 & 0 & x_{14} & x \end{array}\right] \left[\begin{array}{ccc|ccc} x & x & x & x & x & x & x \\ 0 & x & x & x & x & x & x \\ 0 & 0 & x & x & x & x & x \\ \hline 0 & 0 & 0 & x & x & x & x \\ \hline 0 & 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & 0 & x \end{array}\right] \left[\begin{array}{ccc|ccc} x & x & x & x & x & x & x \\ x & x & x & x & x & x & x \\ 0 & x & x & x & x & x & x \\ \hline 0 & 0 & 0 & x & x & x & x \\ \hline 0 & 0 & 0 & 0_{13} & x & x & x \\ 0 & 0 & 0 & 0 & 0_{12} & x & x \\ 0 & 0 & 0 & 0 & 0 & 0_{11} & x \end{array}\right] .$$

and finally :

$$\left[\begin{array}{ccc|c|ccc}
x & x & x & x & x & x & x \\
0 & x & x & x & x & x & x \\
0 & 0 & x & x & x & x & x \\
\hline
0 & 0 & 0 & 0 & x & x & x \\
\hline
0 & 0 & 0 & 0 & x & x & x \\
0 & 0 & 0 & 0 & 0_{18} & x & x \\
0 & 0 & 0 & 0 & 0 & 0_{17} & x
\end{array}\right]
\left[\begin{array}{ccc|c|ccc}
x & x & x & x & x & x & x \\
0 & x & x & x & x & x & x \\
0 & 0 & x & x & x & x & x \\
\hline
0 & 0 & 0 & x & x & x & x \\
\hline
0 & 0 & 0 & 0 & x & x & x \\
0 & 0 & 0 & 0 & 0 & x & x \\
0 & 0 & 0 & 0 & 0 & 0 & x
\end{array}\right]
\left[\begin{array}{ccc|c|ccc}
x & x & x & x & x & x & x \\
x & x & x & x & x & x & x \\
0 & x & x & x & x & x & x \\
\hline
0 & 0 & 0 & x & x & x & x \\
\hline
0 & 0 & 0 & 0 & x & x & x \\
0 & 0 & 0 & 0 & x_{20} & x & x \\
0 & 0 & 0 & 0 & 0 & x_{19} & x
\end{array}\right]$$

which is precisely the desired form. Notice that all this requested about n Givens rotations on each side of each condensed matrix. As a result a zero eigenvalue was deflated and moreover a block reduction was obtained as the same time (see section 5.1 for more details on the operation count).

Case 3. We now consider the case where the closest matrix with a zero diagonal element occurs in a matrix B_i. Without loss of generality we may assume that it is the matrix B_1, since we can always associate the subproduct $A_i B_{i-1}^{-1} A_{i-1} \ldots B_1^{-1} A_1$ with the matrix A_1 (this subproduct indeed exists and is unreduced Hessenberg). Below we thus take the example $\ldots B_1, A_1$ where B_1 has a zero diagonal in position (4,4) :

$$\ldots \left[\begin{array}{cccccc}
x & x & x & x & x & x \\
0 & x & x & x & x & x \\
0 & 0 & x & x & x & x \\
0 & 0 & 0 & 0 & x & x \\
0 & 0 & 0 & 0 & x & x \\
0 & 0 & 0 & 0 & 0 & x
\end{array}\right]
\left[\begin{array}{cccccc}
x & x & x & x & x & x \\
x & x & x & x & x & x \\
0 & x & x & x & x & x \\
0 & 0 & x & x & x & x \\
0 & 0 & 0 & x & x & x \\
0 & 0 & 0 & 0 & x & x
\end{array}\right]$$

We first perform a row transformation $Z_1^* = G_1$ on both B_1 and A_1 where $G_1 \in \mathcal{G}(4,5)$ is chosen to annihilate the element 0_1 in B_1. At the same time a nonzero element x_1 is created in A_1 :

$$\ldots \left[\begin{array}{cccccc}
x & x & x & x & x & x \\
0 & x & x & x & x & x \\
0 & 0 & x & x & x & x \\
0 & 0 & 0 & 0 & x & x \\
0 & 0 & 0 & 0 & 0_1 & x \\
0 & 0 & 0 & 0 & 0 & x
\end{array}\right]
\left[\begin{array}{cccccc}
x & x & x & x & x & x \\
x & x & x & x & x & x \\
0 & x & x & x & x & x \\
0 & 0 & x & x & x & x \\
0 & 0 & x_1 & x & x & x \\
0 & 0 & 0 & 0 & x & x
\end{array}\right]$$

Then a column transformation $Q_1 = G_2$ with $G_2 \in \mathcal{G}(3,4)$ is applied to A_1 to annihilate the element x_1 again (yielding 0_2). Propagating this over all triangular matrices back to B_1 yields a column transformation $Q_2 \in \mathcal{G}(3,4)$ that does not create any fill in :

$$\ldots \left[\begin{array}{cccccc}
x & x & x & x & x & x \\
0 & x & x & x & x & x \\
0 & 0 & x & x & x & x \\
0 & 0 & 0 & 0 & x & x \\
0 & 0 & 0 & 0 & 0 & x \\
0 & 0 & 0 & 0 & 0 & x
\end{array}\right]
\left[\begin{array}{cccccc}
x & x & x & x & x & x \\
x & x & x & x & x & x \\
0 & x & x & x & x & x \\
0 & 0 & x & x & x & x \\
0 & 0 & 0_2 & x & x & x \\
0 & 0 & 0 & 0 & x & x
\end{array}\right]$$

After this step the B_1 matrix has two consecutive zero diagonal elements. The next pair of steps move these zero diagonals one elements down while keeping A_1 Hessenberg. First apply a row transformation

$Z_1^* = G_3$ on both B_1 and A_1 where $G_3 \in \mathcal{G}(5,6)$ annihilates 0_3 in B_1 and creates x_3 in A_1 :

$$\cdots \begin{bmatrix} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & x \\ 0 & 0 & 0 & 0 & 0 & 0_3 \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & x_3 & x & x \end{bmatrix}$$

Then apply the column transformation $Q_1 = G_4$ with $G_4 \in \mathcal{G}(4,5)$ on A_1 to annihilate the element x_3 again (yielding 0_4). Propagating this over all triangular matrices back to B_1 yields a column transformation $Q_2 \in \mathcal{G}(4,5)$ that creates the element x_4 :

$$\cdots \begin{bmatrix} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & x_4 & x & x \\ 0 & 0 & 0 & 0 & 0 & x \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0_4 & x & x \end{bmatrix}$$

With the two consecutive zero diagonals now at the bottom of B_1, we finally apply a column transformation $Q_1 = G_5$ with $G_5 \in \mathcal{G}(5,6)$ on A_1 to annihilate its bottom off diagonal element (yielding 0_5). Propagating this back to B_1 yields a column transformation $Q_2 \in \mathcal{G}(5,6)$ that creates the element x_5 :

$$\cdots \left[\begin{array}{ccccc|c} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x_5 & x \\ \hline 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right] \left[\begin{array}{ccccc|c} x & x & x & x & x & x \\ x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & x & x & x \\ \hline 0 & 0 & 0 & 0 & 0_5 & x \end{array}\right]$$

The above form can now be deflated as indicated above. Notice that again the number of Givens transformations applied to each matrix is at most of the order of n for one deflated eigenvalue at ∞.

Summary. The above three cases indicate that any zero pivot element can be deflated with $O(n)$ Givens transformations per matrix, until a (set of) lower dimensional problem(s) is obtained where now all triangular matrices are invertible and A_1 is unreduced Hessenberg. In the proof of Theorem 1 and Corollary 1 the general case can thus be "pretreated" by the Hessenberg-triangular reduction followed by the direct deflation described above. Theorem 1 and Corollary 1 can then be applied to these "nonsingular" cases, which implicitly yields a proof of these theorems for the general case where any B_i or A_i may be singular. Moreover, since the above procedure allows us to reduce the general problem to the nonsingular case, we only need to consider this simpler case in the sequel.

3.3 Shift calculation and QR step construction

Since we have now a Hessenberg-triangular form with all lower order matrices invertible and unreduced, the corresponding products $B_K^{-1}A_K \ldots B_2^{-1}A_2B_1^{-1}A_1$ exist and are unreduced Hessenberg. In the QR algorithm applied to an unreduced Hessenberg matrix, the shift is typically computed from the bottom 2×2 submatrix. For the above sequence, this is of the form

$$\begin{bmatrix} b_{n-1,n-1}^{(K)} & b_{n-1,n}^{(K)} \\ 0 & b_{n,n}^{(K)} \end{bmatrix}^{-1} \begin{bmatrix} a_{n-1,n-1}^{(K)} & a_{n-1,n}^{(K)} \\ 0 & a_{n,n}^{(K)} \end{bmatrix} \cdots \begin{bmatrix} b_{n-1,n-1}^{(1)} & b_{n-1,n}^{(1)} \\ 0 & b_{n,n}^{(1)} \end{bmatrix}^{-1} \begin{bmatrix} a_{n-1,n-1}^{(1)} & a_{n-1,n}^{(1)} \\ a_{n,n-1}^{(1)} & a_{n,n}^{(1)} \end{bmatrix} \tag{12}$$

Notice that the triangular 2×2 inverses can be replaced by their adjoints up to a scalar factor. The eigenvalues of this 2×2 matrix are thus easily computed and are used for calculating the shift of the QR-step.

The transformation Q_1 of the QR step applied to the Hessenberg matrix

$$B_K^{-1} A_K \ldots B_2^{-1} A_2 B_1^{-1} A_1$$

is now completely defined by its first column. In the case of a single shift λ, this first column has only two nonzero elements, corresponding to the normalized version of the 2-vector :

$$\begin{bmatrix} b_{1,1}^{(K)} & b_{1,2}^{(K)} \\ 0 & b_{2,2}^{(K)} \end{bmatrix}^{-1} \begin{bmatrix} a_{1,1}^{(K)} & a_{1,2}^{(K)} \\ 0 & a_{2,2}^{(K)} \end{bmatrix} \cdots \begin{bmatrix} b_{1,1}^{(1)} & b_{1,2}^{(1)} \\ 0 & b_{2,2}^{(1)} \end{bmatrix}^{-1} \begin{bmatrix} a_{1,1}^{(1)} \\ a_{2,1}^{(1)} \end{bmatrix} - \begin{bmatrix} \lambda \\ 0 \end{bmatrix}$$

Since the matrices Q_i and Z_i are all defined by one another through the constraint that updates on B_i, $i = 1 \ldots, K$ and A_i, $i = 2, \ldots, K$ must be upper triangular, one could as well compute any other matrix than Q_1. It turns out that the simplest one to contruct is Z_1. It performs a QR step on the unreduced Hessenberg matrix

$$A_H \doteq A_1 B_K^{-1} A_K \ldots B_2^{-1} A_2 B_1^{-1}$$

and is again defined by its first column, consisting of only two nonzero elements. Now this 2-vector is the normalized version of :

$$\begin{bmatrix} a_{1,1}^{(1)} \\ a_{2,1}^{(1)} \end{bmatrix} - \begin{bmatrix} \lambda \\ 0 \end{bmatrix} \frac{b_{1,1}^{(1)} \cdot \ldots \cdot b_{1,1}^{(K)}}{a_{1,1}^{(2)} \cdot \ldots \cdot a_{1,1}^{(K)}}$$

which involves much less computations.

In the implicit double shift one determines the first column of the real matrix $(A_H - \lambda_1)(A_H - \lambda_2)$ where λ_1 and λ_2 are the two eigenvalues of (12). In order to avoid complex arithmetic when λ_i, $i = 1, 2$ are complex conjugate one constructs the first column of $A_H^2 - s \cdot A_H + p \cdot I$ where $s = (\lambda_1 + \lambda_2)$ and $p = \lambda_1 . \lambda_2$ are *real*. This vector has only three nonzero elements and is up to a constant :

$$\begin{bmatrix} a_{1,1}^{(1)} & a_{1,2}^{(1)} \\ a_{2,1}^{(1)} & a_{2,2}^{(1)} \\ 0 & a_{3,2}^{(1)} \end{bmatrix} \begin{bmatrix} b_{1,1}^{(K)} & b_{1,2}^{(K)} \\ 0 & b_{2,2}^{(K)} \end{bmatrix}^{-1} \begin{bmatrix} a_{1,1}^{(K)} & a_{1,2}^{(K)} \\ 0 & a_{2,2}^{(K)} \end{bmatrix} \cdots \begin{bmatrix} b_{1,1}^{(1)} & b_{1,2}^{(1)} \\ 0 & b_{2,2}^{(1)} \end{bmatrix}^{-1} \begin{bmatrix} a_{1,1}^{(1)} \\ a_{2,1}^{(1)} \end{bmatrix}$$

$$-s \begin{bmatrix} a_{1,1}^{(1)} \\ a_{2,1}^{(1)} \\ 0 \end{bmatrix} + \begin{bmatrix} p \\ 0 \\ 0 \end{bmatrix} \frac{b_{1,1}^{(1)} \cdot \ldots \cdot b_{1,1}^{(K)}}{a_{1,1}^{(2)} \cdot \ldots \cdot a_{1,1}^{(K)}}.$$

3.4 Periodic QR step

Again for simplicity we only consider the product of four matrices $B_2^{-1} A_2 B_1^{-1} A_1$ and the case of a single shift in order to explain the general idea. The first three matrices are upper triangular. The last matrix A_1 is upper Hessenberg.

$$\begin{bmatrix} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x \end{bmatrix}$$

Apply first $Z_1^* \in \mathcal{G}(1,2)$ to annihilate the bottom element in the 2-vector determined above. Applying this to the rows of B_1 and A_1 yields :

$$\begin{bmatrix} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ x_1 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x \end{bmatrix}$$

Then construct the column transformation $Q_2 \in \mathcal{G}(1,2)$ to annihilate again x_1 in B_1 but also apply this transformation to the columns of A_2, creating x_2 :

$$\begin{bmatrix} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ x_2 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ 0_2 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x \end{bmatrix}$$

Then apply the row transformation $Z_2^* \in \mathcal{G}(1,2)$ to B_2 and A_2 annihilating x_2 but creating x_3 :

$$\begin{bmatrix} x & x & x & x & x & x \\ x_3 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ 0_3 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ x & x & x & x & x & x \\ x & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x \end{bmatrix}$$

Finally close the loop with the column transformation $Q_2 \in \mathcal{G}(1,2)$ applied to B_2 and A_1 to annihilate again x_3 but creating a "bulge" x_4 in A_1 :

$$\begin{bmatrix} x & x & x & x & x & x \\ 0_4 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ x & x & x & x & x & x \\ x_4 & x & x & x & x & x \\ 0 & 0 & x & x & x & x \\ 0 & 0 & 0 & x & x & x \\ 0 & 0 & 0 & 0 & x & x \end{bmatrix}$$

Repeating this process chases the bulge one step down at each sequence of Givens transformations, untill it finally dissapears at the bottom of the Hessenberg matrix A_1. Basically the same procedure applies to the implicit double shift for real matrices except that then the bulge chasing transformations are 3×3 unitary matrices, realized by a product of Householder transformations or Givens transformations.

3.5 Reordering eigenvalues

We assume now that an upper triangular decomposition was obtained upon convergence of the above QR steps (blow there is only one 2×2 block in A_1). Then we want to permute the two (real) eigenvalues corresponding to the diagonal elements x_1 and x_2 :

$$\begin{bmatrix} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x_1 & x & x & x \\ 0 & 0 & 0 & x_2 & x & x \\ 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x_1 & x & x & x \\ 0 & 0 & 0 & x_2 & x & x \\ 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ 0 & x & x & x & x & x \\ 0 & 0 & x_1 & x & x & x \\ 0 & 0 & 0 & x_2 & x & x \\ 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & x \end{bmatrix} \begin{bmatrix} x & x & x & x & x & x \\ x & x & x & x & x & x \\ 0 & 0 & x_1 & x & x & x \\ 0 & 0 & 0 & x_2 & x & x \\ 0 & 0 & 0 & 0 & x & x \\ 0 & 0 & 0 & 0 & 0 & x \end{bmatrix}$$

One then computes the product of the corresponding 2×2 matrices and computes from there the requested updating Givens transformations that will perform the swapping. Care has to be taken to implement this in a numerically stable manner as was e.g. the case for the QZ reordering in [14]. This especially applies to the swapping of two 2×2 blocks which is a much more delicate problem.

4 Applications of the periodic Schur form

4.1 Periodic control systems

The application of this decomposition to control theory is apparent. Periodic discrete time systems naturally arise when performing multirate sampling of continuous time systems [1]. In optimal control of such a periodic system one considers the problem :

$$\begin{aligned} \text{Minimize } J &= \textstyle\sum_{i=1}^{\infty} z_i^T Q_i z_i + u_i^T R_i u_i \\ \text{subject to } E_i z_{i+1} &= F_i z_i + G_i u_i \end{aligned} \qquad (13)$$

where the matrices Q_i, R_i, E_i, F_i, G_i are periodic with period K. The Hamiltonian equations are periodic homogenous systems of difference equations (2) in the state z_i and co-state λ_i of the system. The correspondences with (2) are :

$$x_i \doteq \begin{bmatrix} \lambda_i \\ z_i \end{bmatrix}, B_i \doteq \begin{bmatrix} -G_i R_i^{-1} G_i^T & E_i \\ F_i^T & 0 \end{bmatrix}, A_i \doteq \begin{bmatrix} 0 & F_i \\ E_i^T & Q_i \end{bmatrix}. \qquad (14)$$

For finding the periodic solutions to the underlying periodic Riccati equation one has to find the stable invariant subspaces of matrices $S^{(i)}$ as above, which happen to be simplectic in the discrete time case (one has to assume here that E_i, F_i and R_i are invertible and eliminate implicitly E_i [7]). Clearly the Schur form is useful here as well as the reordering of eigenvalues [10], [14].

In pole placement of periodic systems [9], again the periodic Schur form and reordering is useful when one wants to extend Varga's pole placement algorithm [17] to periodic systems. Consider the system

$$\begin{aligned} B_i z_{i+1} &= A_i z_i + D_i u_i \\ \text{with state feedback } u_i &= F_i z_i + v_i \end{aligned} \qquad (15)$$

where the matrices A_i, B_i, D_i, F_i are periodic with period K. This results in the closed loop system

$$B_i z_{i+1} = (A_i + D_i F_i) z_i + D_i v_i \qquad (16)$$

of which the underlying time invariant eigenvalues are those of the matrix :

$$S_F^{(1)} \doteq B_K^{-1}(A_K + D_K F_K) \cdots B_2^{-1}(A_2 + D_2 F_2) B_1^{-1}(A_1 + D_1 F_1). \qquad (17)$$

In the above equation it is not apparent at all how to choose the matrices F_i to assign particular eigenvalues of $S_F^{(1)}$. Yet when the matrices A_i, B_i are in the triangular form (6), one can choose the F_i matrices to have only nonzero elements in the last column. This will preserve the triangular form of the matrices $A_i + D_i F_i$ and it is then trivial to choose e.g. one such column vector to assign one eigenvalue. In order to assign the other eigenvalues one needs to *reorder* the diagonal elements in the periodic Schur form and each time assign another eigenvalue with the same technique. This algorithm will of course fail when the periodic system is not controllable, but this very procedure can in fact be adapted to precisely construct the controllable subspace of the periodic system.

4.2 K-cyclic matrix problems

Here we consider the following pencils of matrices :

$$\lambda \mathcal{B} - \mathcal{A} \doteq \lambda \begin{bmatrix} B_K & 0 & \cdots & \cdots & 0 \\ 0 & B_1 & 0 & \cdots & 0 \\ \vdots & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & B_{K-2} & 0 \\ 0 & 0 & \cdots & 0 & B_{K-1} \end{bmatrix} - \begin{bmatrix} 0 & 0 & \cdots & 0 & A_K \\ A_1 & 0 & 0 & \cdots & 0 \\ \vdots & A_2 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & 0 & 0 \\ 0 & 0 & \cdots & A_{K-1} & 0 \end{bmatrix}. \tag{18}$$

If the B_i matrices here are invertible one can divide them out by columns transformation, yielding :

$$\lambda I_{nK} - \mathcal{B}^{-1}\mathcal{A} \doteq \lambda I_{nK} - \mathcal{S} \doteq \lambda \begin{bmatrix} I_n & 0 & \cdots & \cdots & 0 \\ 0 & I_n & 0 & \cdots & 0 \\ \vdots & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & I_n & 0 \\ 0 & 0 & \cdots & 0 & I_n \end{bmatrix} - \begin{bmatrix} 0 & 0 & \cdots & 0 & S_K \\ S_1 & 0 & 0 & \cdots & 0 \\ \vdots & S_2 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & 0 & 0 \\ 0 & 0 & \cdots & S_{K-1} & 0 \end{bmatrix}$$

where the matrices $S_i = B_i^{-1} A_i$ are as defined earlier. The matrix $\mathcal{S}$ is now known as a K-cyclic matrix, and by extension we will call $\lambda\mathcal{B} - \mathcal{A}$ a K-cyclic pencil. It is well-known that the eigenvalues of $\mathcal{S}$ are the K-th roots of those of the matrix $\mathcal{S}^K$, but the latter is easily checked to be block diagonal :

$$\mathcal{S}^K \doteq \begin{bmatrix} S^{(1)} & 0 & \cdots & \cdots & 0 \\ 0 & S^{(2)} & 0 & \cdots & 0 \\ \vdots & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & S^{(K-1)} & 0 \\ 0 & 0 & \cdots & 0 & S^{(K)} \end{bmatrix}$$

where again the matrices $S^{(i)}$ are as defined earlier. This shows the relation between the two problems. We now show that the decomposition (6) actually yields a block Schur decomposition of the above pencil as well. Indeed the orthogonal transformations $\mathcal{Z} \doteq \text{diag}\{Z_K, Z_1, \ldots Z_{K-1}\}$ and $\mathcal{Q} \doteq \text{diag}\{Q_1, \ldots Q_{K-1}, Q_K\}$ yield a pencil $\mathcal{Z}^* \cdot (\lambda\mathcal{B} - \mathcal{A}) \cdot \mathcal{Q}$ which after appropriate reordering becomes *upper block triangular* with on diagonal pencils of the type :

$$\lambda \begin{bmatrix} b^{(*)}_{K,K} & 0 & \cdots & & 0 \\ 0 & b^{(*)}_{1,1} & 0 & \cdots & 0 \\ \vdots & 0 & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & b^{(*)}_{K-2,K-2} & 0 \\ 0 & 0 & \cdots & 0 & b^{(*)}_{K-1,K-1} \end{bmatrix} - \begin{bmatrix} 0 & 0 & \cdots & 0 & a^{(*)}_{K,K} \\ a^{(*)}_{1,1} & 0 & 0 & \cdots & 0 \\ \vdots & a^{(*)}_{2,2} & \ddots & \ddots & \vdots \\ \vdots & \vdots & \ddots & 0 & 0 \\ 0 & 0 & \cdots & a^{(*)}_{K-1,K-1} & 0 \end{bmatrix}$$

where $^{(i)}$ indicates that the element belongs to the triangular matrices $\hat{A}_i$ or $\hat{B}_i$. For this reason the pencil $\lambda\mathcal{B} - \mathcal{A}$ is nonsingular iff $a^{(i)}_{j,j} b^{(i\pm1)}_{j,j} \neq 0$, i.e. iff there are no zero by zero divides in two *consecutive* elements (in a periodic sense) on the diagonals of the decomposition (6).

4.3 Two point boundary value problems

In the solution of two point boundary problems (not necessarily periodic), one encounters inversions of matrices of cyclic type $(\mathcal{B} + \mathcal{A})x = u$ where $\mathcal{A}$ and $\mathcal{B}$ are as above (18). Again we can apply the orthogonal transformations $\mathcal{Z}^*$ and $\mathcal{Q}$ to obtain the system of equations $\mathcal{Z}^*(\mathcal{B} + \mathcal{A})\mathcal{Q}(\mathcal{Q}^* x) = \mathcal{Z}^* u$ which essentially decomposes in n *scalar* TPBV problems. The big advantage of this is that increasing and decreasing solution in the TPBV problem have been decoupled. The periodic Schur form in fact "aligns" stable and unstable solutions at each step. The decomposition could also be computed at a coarse mesh and then "extrapolated" at finer meshes in order to avoid too much work. This is still under investigation.

5 Numerical aspects

The use of Householder and Givens transformations for all operations in the periodic QR algorithm guarentees that the obtained matrices $\hat{A}_i$ and $\hat{B}_i$ in fact correspond to slightly perturbed data as follows (indices are taken modulo K) :

$$\hat{A}_i = \overline{Z}_i^*(A_i + \delta A_i)\overline{Q}_i, \quad \hat{B}_i = \overline{Z}_i^*(B_i + \delta B_i)\overline{Q}_{i+1},$$

where $\overline{Q}_i$ and $\overline{Z}_i$ are *exactly unitary* matrices and where $\|\overline{Q}_i - Q_i\|$, $\|\overline{Z}_i - Z_i\|$, $\|\delta A_i\|/\|A_i\|$ and $\|\delta B_i\|/\|B_i\|$ are all of the order of the machine precision ϵ. This is obvious for the Hessenberg-triangular reduction and the direct deflation since each element transformed to zero can indeed be put equal to zero without affecting the ϵ bound (see [18], [8]). Things are different with the QR steps, since there one puts off-diagonal elements in A_1 equal to zero *only when these elements have converged to sufficiently small elements*. Convergence of the QR process is thus needed to guarentee stability as well. Finally, for the reordering one needs to *prove* that the swapping transformations indeed result in strictly upper triangular matrices with reversed order of eigenvalues. This is the subject of another report.

6 Concluding remarks

The above decomposition has clearly many applications and we expect that additional ones will be found in the future (e.g. in robust control of periodic systems). The above decomposition is also related to [4] which computes the Jordan chains of sequences as considered here. This *generalized QR decomposition* in fact plays the role of the rank determination (via QR or SVD) needed to reconstruct the Jordan/Kronecker structure of pencils of the type (18). This could be used as a preprocessing to eliminate the chains at $\lambda = 0$ or $\lambda = \infty$ and extract in this manner a set of smaller but invertible matrices A_i, B_i as was also done in section 3.2 via direct deflation. The advantage of this new approach is that it also identifies the

structural indices at these two eigenvalues. Moreover, the generalized QR decomposition allows for *non-square matrices* as well, and one can thus consider systems of the type (2) with $m \times n$ matrices A_i and B_i.

Similar unpublished ideas are being pursued by John Hench, UC Santa Barbara (personal communication), who arrives at the same decomposition (6) with a different algorithm. His condensed form essentially consists of all A_i matrices in Hessenberg form and all B_i matrices in triangular form. We feel that the connection with the QR algorithm then fails to go through, although he reports a good convergence of that algorithm as well. Possible application to periodic continuous control systems are also being considered by him.

The present report is a more extended version of the paper [3] presented at the SPIE conference held in San Diego in July 1992.

Acknowledgements

Part of this research was performed while the authors were visiting the Institute of Mathematics and Applications of the University of Minnesota, Minneapolis, during the summer quarter of the Applied Linear Algebra Year organized there. We greatly appreciated the hospitality and the productive atmosphere of that institute. Bojanczyk was partially supported by the Joint Services Electronics Program (Grant F49620 90 C-0039 monitored by AFOSR). G. Golub was partially supported by the Army Research (Grant DAAL03 90-G-0105) and ARGOSystems (Grant 59613 Dept. Air Force). P. Van Dooren was partially supported by the Research Board of the University of Illinois at Urbana-Champaign (Grant P 1-2-68114) and by the National Science Foundation (Grant CCR 9209349).

References

[1] B. Francis, T. Georgiou, Stability theory for linear time-invariant plants with periodic digital controllers, *IEEE Trans. Aut. Contr.* **33** (1988) 820-832.

[2] S. Bittanti, P. Colaneri, G. de Nicolao, The difference periodic Riccati equation for the periodic prediction problem, *IEEE Trans. Aut. Contr.* **33** (1988) 706-712.

[3] A. Bojanczyk, G. Golub, P. Van Dooren, The periodic Schur form. Algorithms and Applications, SPIE Conference, San Diego, July 1992.

[4] B. De Moor, P. Van Dooren, Generalizations of the singular value and QR decomposition, *SIAM Matr. Anal. & Applic.* **13** (1992).

[5] J. Doyle, B. Francis and A. Tannenbaum, *Feedback Control Theory*, McMillan, 1992.

[6] D. Flamm, A new shift-invariant representation for periodic linear systems, *Proceedings American Control Conference*, May 1990, San Diego CA, 1510-1515.

[7] J. Gardiner, A. Laub, A generalization of the matrix-sign-function solution to the algebraic Riccati equations, *Int. Journal Control* **44** (1986) 823-832.

[8] G. Golub, C. Van Loan, *Matrix Computations* 2nd edition, The Johns Hopkins University Press, Baltimore, Maryland, 1989.

[9] O. Grasselli, S. Longhi, The geometric approach for linear periodic discrete-time systems, *Lin. Algebra & Applic.* **158** (1991) 27-60.

[10] A. Laub, Invariant subspace methods for the numerical solution of Riccati equations, in *The Riccati equation*, Eds. Bittanti, Laub, Willems, Springer Verlag, 1990.

[11] C. Moler, G. Stewart, An algorithm for the generalized matrix eigenvalue problem, *SIAM Numer. Anal.* **10** (1973) 241-256.

[12] A. Sage, C. White, *Optimum Systems Control*, 2nd Ed., Prentice-Hall, New Jersey, 1977.

[13] P. Van Dooren, The generalized eigenstructure problem in linear system theory, *IEEE Trans. Aut. Contr.* **26** (1981) 111-129.

[14] P. Van Dooren, A generalized eigenvalue approach for solving Riccati equations, *SIAM Sci. & Stat Comp.* **2** (1981) 121-135.

[15] P. Van Dooren, M. Verhaegen, On the use of unitary state-space transformations, in *Special Issue of Contemporary Mathematics on Linear Algebra and its Role in Linear System Theory*, AMS, 1985.

[16] P. Van Dooren, Numerical aspects of system and control algorithms, *Journal A* **30** (1989) 25-32.

[17] A. Varga, A Schur method for pole assignment, *IEEE Trans. Aut. Contr.* **26** (1981) 517-519.

[18] J. H. Wilkinson, *The algebraic eigenvalue problem*, Clarendon press, Oxford, 1965.

TASK 6 FAULT TOLERANT BEAMFORMING ALGORITHMS

F. T. Luk

Computing the PSVD of two 2 × 2 triangular matrices

Gary E. Adams and Adam W. Bojanczyk
School of Electrical Engineering
Cornell University
Ithaca, NY 14853, USA

Franklin T. Luk
Department of Computer Science
Rensselaer Polytechnic Institute
Troy, NY 12180, USA

Abstract

In this paper, we propose a method for computing the SVD of a product of two 2 × 2 triangular matrices. We show that our method is numerically desirable in that all relevant residual elements will be numerically small.

1. Introduction

The problem of computing the singular value decomposition (SVD) of a product of two matrices has many applications; see, e.g., [4] and [5]. The problem is also closely related to finding a generalized SVD of two matrices (cf. [6]). A crucial step in either the product SVD (PSVD) or the generalized SVD (GSVD) problem is the accurate computation of the PSVD of two 2 × 2 triangular matrices.

We wish to achieve two objectives: first, to ensure that the transformations applied to the triangular matrices must leave the matrices triangular and, second, to ensure that the SVD of the product is computed accurately. As discussed in a recent paper by Bai and Demmel [1], these two properties are essential to guarantee the stability of the GSVD method [6]. Several strategies have been proposed to preserve these two properties. In [1], examples are presented where these strategies can fail and a new method that overcomes the exposed drawbacks is then proposed.

In this paper we propose an alternative approach. Our new method, which we will call a *half-recursive* method, is a slight variation of the *fully-recursive* method proposed in [2] for computing the SVD of a product of several matrices. We show that our algorithm is *simpler* to implement and enjoys the same nice numerical properties as the method in [1].

Our paper is organized as follows. In Section 2 we describe the PSVD of two 2 × 2 upper triangular matrices. A criterion for numerical stability is given in Section 3. We present our new algorithm in Section 4, and an error analysis in Section 5. Finally, some detailed proofs can be found in Appendices A and B, and a numerical example in Appendix C.

0-8194-0943-X/92/$4.00

2. Problem Definition

Given two upper triangular matrices:

$$A_1 = \begin{pmatrix} a_1 & b_1 \\ 0 & d_1 \end{pmatrix} \quad \text{and} \quad A_2 = \begin{pmatrix} a_2 & b_2 \\ 0 & d_2 \end{pmatrix},$$

we call the product A:

$$A = A_1 A_2,$$

and let

$$A = \begin{pmatrix} a & b \\ 0 & d \end{pmatrix}$$

Our objective is to find three orthogonal matrices Q_1, Q_2, Q_3 such that

$$A' = Q_1 A Q_3^T = \begin{pmatrix} a' & 0 \\ 0 & d' \end{pmatrix} \tag{2.1}$$

and

$$A_i' = Q_i A_i Q_{i+1}^T = \begin{pmatrix} a_i' & b_i' \\ 0 & d' \end{pmatrix}, \tag{2.2}$$

for $i = 1, 2$. The two equations (2.1) and (2.2) imply that

$$A' = A_1' A_2'.$$

In other words, we would like to find *three* transformations Q_1, Q_2 and Q_3 to zero out *four* elements, namely, the off-diagonal elements of A and the sub-diagonal elements of A_1 and A_2. The extra requirement, although mathematically feasible, may cause numerical difficulty if not treated with care; see examples in [1] and [2]. Our goal is to develop an algorithm so that properties (2.1) and (2.2) will be satisfied except for very small numerical errors. In this paper, we use the vector and matrix 2-norms:

$$\|x\|_2 = \left(\sum_i |x_i|^2\right)^{1/2},$$

2.1. Relationship with GSVD

The basic step in a GSVD of two 2×2 triangular matrices A_1 and A_2 is to compute the SVD of the product $A_1 \cdot \mathrm{adj}(A_2)$, where adj denotes the adjoint of a matrix. We have

$$\mathrm{adj}(A_2) = \begin{pmatrix} d_2 & -b_2 \\ 0 & a_2 \end{pmatrix}.$$

It is therefore obvious that our two-by-two PSVD method can also be applied to the two-by-two GSVD problem.

3. Criterion for Numerical Stability

Recall that A_1', A_2' and A' denote the three matrices A_1, A_2 and A, respectively, after the equivalence transformations as defined in (2.1) and (2.2) have been performed. Let $\tilde{A}$ be the computed A, and let $\tilde{Q}_1$, $\tilde{Q}_2$, and $\tilde{Q}_3$ be the computed transformations. Define

$$\tilde{A}' := \tilde{Q}_1 \tilde{A} \tilde{Q}_3^T = \begin{pmatrix} \tilde{a}' & \tilde{b}' \\ \tilde{c}' & \tilde{d}' \end{pmatrix} \tag{3.1}$$

and

$$\tilde{A}'_i = Q_i A_i Q_{i+1}^T = \begin{pmatrix} \tilde{a}'_i & \tilde{b}'_i \\ \tilde{c}'_i & \tilde{d}'_i \end{pmatrix} \tag{3.2}$$

Let ϵ denote the relative machine precision. The best that we can aim for is to compute $\tilde{A}'_i$ such that

$$\| \tilde{A}'_i - A'_i \| = O(\epsilon) \tag{3.3}$$

The relation (3.3) implies that the (2,1) element $\tilde{c}'_i$ of $\tilde{A}'_i$ will satisfy

$$|\tilde{c}'_i| = O(\epsilon \| A_i \|), \tag{3.4}$$

for $i = 1, 2$. Condition (3.4) implies that $\tilde{c}'_i$ may be safely truncated to zero. Thus, $\tilde{c}'$ is also forced to zero.

We prove in Section 5 that by using our new method, the computed matrices $\tilde{A}'_1$ and $\tilde{A}'_2$ will satisfy condition (3.4) and $\tilde{A}'$ will satisfy the conditions that

$$|\tilde{b}'| = O(\epsilon \| A \|) \tag{3.5a}$$

and

$$|\tilde{c}'| = O(\epsilon \| A \|). \tag{3.5a}$$

The conditions proposed in [1] for computing the GSVD of two matrices, A_1 and adj(A_2), follow from (3.4), (3.5), and the similar construction of the two algorithms.

4. New Algorithm

In this section, we propose a new algorithm for the PSVD problem. Our algorithm is a modification of the algorithm presented in [2] for a product of several matrices. The tool we use is a transformation discussed in Charlier et al. [3]:

$$Q = \begin{pmatrix} s & c \\ -c & s \end{pmatrix}. \tag{4.1}$$

where $c^2 + s^2 = 1$. We may regard the transformation as a permuted reflection:

$$Q = \begin{pmatrix} c & s \\ s & -c \end{pmatrix} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.$$

The reason behind using permuted reflections is that we actually deal with an $n \times n$ problem. The permutation that is incorporated into Q corresponds to the so called odd-even order of eliminations in one sweep of a Jacobi-SVD procedure.

While each transformation Q_i is defined by the cosine-sine pair:

$$c_i = \cos\theta_i \quad \text{and} \quad s_i = \sin\theta_i,$$

we also associate Q_i with the tangent

$$t_i = \tan\theta_i.$$

Given t_i, we can easily recover c_i and s_i using the relations

$$c_i = \frac{1}{\sqrt{1 + t_i^2}} \quad \text{and} \quad s_i = t_i c_i. \tag{4.2}$$

Following the exposition in [2], we consider the result of applying the left and right transformations Q_l and Q_r to a 2×2 upper triangular matrix A

$$A' = Q_l A Q_r^T = \begin{pmatrix} a' & b' \\ \epsilon' & d' \end{pmatrix} = \begin{pmatrix} s_l & c_l \\ -c_l & s_l \end{pmatrix} \begin{pmatrix} a & b \\ 0 & d \end{pmatrix} \begin{pmatrix} s_r & c_r \\ -c_r & s_r \end{pmatrix}^T . \tag{4.3}$$

We can derive from (4.3) these four relations:

$$\epsilon' = c_l c_r (-a t_r + d t_l - b) , \tag{4.4a}$$

$$b' = c_l c_r (-a t_l + d t_r + b t_l t_r) , \tag{4.4b}$$

$$a' = c_l c_r (b t_l + d + a t_l t_r) , \tag{4.4c}$$

$$d' = c_l c_r (a - b t_r + d t_l t_r) , \tag{4.4d}$$

where $t_l = \tan\theta_l$ and $t_r = \tan\theta_r$. The postulates that both ϵ' and b' be zeros define two conditions on t_l and t_r, so that (4.3) represents an SVD of A. The postulate that ϵ' be zero defines a condition relating θ_l to θ_r, so that if one is known the other can be computed in order to reduce A' to an upper triangular form. For ease of exposition, assume for now that $abd \neq 0$. This condition will be removed in Section 5.2. It implies that $c_l c_r \neq 0$, and so the postulate that $\epsilon' = 0$ in (4.4a) becomes

$$-a t_r + d t_l - b = 0 . \tag{4.4e}$$

The consequence of (4.4e) is that (4.4c) and (4.4d) simplify to

$$a' = c_l c_r (t_l^2 + 1) d \tag{4.4f}$$

and

$$d' = c_l c_r (t_r^2 + 1) a , \tag{4.4g}$$

respectively. The relations (4.4f) and (4.4g) imply that

$$a' d' = a d .$$

For the SVD problem, both ϵ' and b' are zeros and we can use (4.4e) to reduce (4.4b) either to an equation in t_l:

$$b' = c_l c_r \left(\frac{bd}{a}\right) \left(t_l^2 + 2 t_l \sigma_l - 1\right) , \tag{4.5a}$$

where

$$\sigma_l = \frac{1}{2d} \left(\frac{d^2 - a^2}{b} - b\right) ,$$

or to an equation in t_r:

$$b' = c_l c_r \left(\frac{ab}{d}\right) \left(t_r^2 + 2 t_r \sigma_r - 1\right) , \tag{4.5b}$$

where

$$\sigma_r = \frac{1}{2a} \left(\frac{d^2 - a^2}{b} + b\right) .$$

From (4.5a) we get a quadratic equation by setting b' to zero:

$$t_l^2 + 2 \sigma_l t_l - 1 = 0 \tag{4.5c}$$

and from (4.5b) we get

$$t_r^2 + 2\sigma_r t_r - 1 = 0 \tag{4.5d}$$

The two equations (4.5c) and (4.5d) are solved by the formulas given in [2]:

$$r = \frac{(d-a)(d+a)}{b} , \tag{4.6a}$$

$$\sigma_l = \frac{r-b}{2d} , \tag{4.6b}$$

$$\sigma_r = \frac{r+b}{2a} , \tag{4.6c}$$

$$t_l = \frac{1}{\sigma_l + \mathrm{sign}(\sigma_l)\sqrt{\sigma_l^2 + 1}} , \tag{4.6d}$$

$$t_r = \frac{1}{\sigma_r + \mathrm{sign}(\sigma_r)\sqrt{\sigma_r^2 + 1}} . \tag{4.6e}$$

In finite-precision arithmetic, either one of t_l and t_r can be computed with a higher relative precision. In particular, if

$$\mathrm{sign}(r) = -\mathrm{sign}(b) ,$$

then (4.6d) will produce a very accurate t_l; whereas if

$$\mathrm{sign}(r) = \mathrm{sign}(b) ,$$

then (4.6e) will produce a very precise t_r. If $r = 0$, then both t_l and t_r will be computed with the same relative accuracy.

Now, let $r \neq 0$. We first present a lemma relating the sizes of t_l and t_r to those of a and d.

Lemma 4.1. Let $abdr \neq 0$. If $|a| > |d|$, then $|\sigma_l| > |\sigma_r|$ and $|t_l| < |t_r|$. Conversely, if $|a| < |d|$, then $|\sigma_l| < |\sigma_r|$ and $|t_l| > |t_r|$.

Proof. See [2]. □

We are ready to present an algorithm for computing the three orthogonal matrices Q_1, Q_2 and Q_3, such that (2.1) and (2.2) are satisfied. The algorithm proceeds in two stages. In the first stage, we calculate the product A explicitly:

$$a = a_1 a_2 , \tag{4.7a}$$

$$b = a_1 b_2 + b_1 d_2 , \tag{4.7b}$$

$$d = d_1 d_2 . \tag{4.7c}$$

We use (4.6a) to calculate r, and then compute either σ_l or σ_r so that the corresponding tangent defines the smaller angular rotation. Hence we obtain either t_1 or t_3. In the second stage, we use the relation (4.4e) with t_1 or t_3 as the reference tangent to compute the remaining transformations. Suppose that t_1 is known, then t_2 and t_3 are generated by the forward substitutions:

$$t_2 = \frac{d_1 t_1 - b_1}{a_1} , \tag{4.8a}$$

$$t_3 = \frac{dt_1 - b}{a} \quad . \tag{4.8b}$$

On the other hand, if t_3 is known, then t_2 and t_1 are generated by the backward substitutions:

$$t_2 = \frac{a_2 t_3 + b_2}{d_2} \quad , \tag{4.8c}$$

$$t_1 = \frac{at_3 + b}{d} \quad . \tag{4.8d}$$

If t_1 is computed first as the reference tangent, then (4.8a) will guarantee that A_1' will be numerically upper triangular and (4.8b) will guarantee that A' will be numerically diagonal. As will be shown later, these two properties will guarantee that A_2' will be numerically upper triangular and hence both (3.4) and (3.5) will be satisfied.

We refer to the method defined by (4.8a)-(4.8b) or (4.8c)-(4.8d) as *half-recursive*, to differentiate it from the *fully-recursive* method proposed in [2] for computing the PSVD of several matrices. The fully-recursive method also picks the smaller outer angular rotation as the starting point for the recursion, from which all remaining rotations are computed. However in [2], the other outer rotation is computed from the previous rotation in the sequence. For example, in the case of a product of two matrices, the tangent t_3 in (4.8b) would be computed from t_2 using (4.4e):

$$t_3 = \frac{d_2 t_2 - b_2}{a_2} \quad . \tag{4.9}$$

Note how (4.8b) uses the product A whereas (4.9) uses the matrix A_2. It was shown in [1] that the fully-recursive method may fail to satisfy (3.5) and thus is not recommended for the GSVD problem. On the other hand, the fully-recursive method easily extends to any number of factors in the product. It is not clear what is an appropriate extension of the half-recursive method for the case of a product of more than two matrices.

Our half-recursive method is equivalent to the method proposed by Bai and Demmel in [1] in the sense that it also computes a very accurate PSVD of $A_1 A_2$, and that it uses essentially the same criterion in choosing whether to compute the middle transformation Q_2 from Q_1 or from Q_3. A proof that the two methods use the same condition for computing Q_2 is given in Appendix B.

5. Backward Error Analysis

In this section, we present a backward error analysis of our computation. The function $\mathrm{fl}(a)$ will be used to denote the floating point approximation of a. For the purpose of this analysis, a "bar" denotes a computed quantity which is perturbed as the result of inexact arithmetic. For example, instead of a, b and d, we have the perturbed values $\bar{a}$, $\bar{b}$ and $\bar{d}$ which result from floating point computation of $A_1 A_2$. We assume that exact arithmetic may be performed using these perturbed values. The "tilde" symbol is used to denote conceptual values computed exactly from perturbed data. For example, $\tilde{r}$ denotes the result of using formula (4.6a) in exact arithmetic with the perturbed data $\bar{a}$, $\bar{b}$ and $\bar{d}$.

In our error analysis, we adopt a convention that involves a liberal use of Greek letters. For example, by α we mean a relative perturbation of an absolute magnitude not greater than ϵ, where ϵ denotes the machine precision. All terms of order ϵ^2 or higher will be ignored in this first-order analysis.

We start our procedure by computing elements of the product matrix A:

$$\bar{a} := \mathrm{fl}(a_1 a_2) = a_1 a_2(1+\alpha)\,, \tag{5.1a}$$

$$\bar{d} := \mathrm{fl}(d_1 d_2) = d_1 d_2(1+\delta)\,, \tag{5.1b}$$

$$\bar{b} := \mathrm{fl}(a_1 b_2 + b_1 d_2) = a_1 b_2(1+2\beta_1) + b_1 d_2(1+2\beta_2)\,, \tag{5.1c}$$

where, according to our convention, the parameters α_1, δ_1, β_1, β_2, and β_3 are all quantities whose absolute values are bounded by ϵ. From (5.1) it follows that

$$\bar{A} = (A_1 + \delta A_1)(A_2 + \delta A_2)\,,$$

with $\| \delta A_i \| \leq \epsilon \| A_i \|$. This property, which in general does not hold for a product of more than two 2×2 upper triangular matrices, will allow us to prove backward error type assertions on the half-recursive method.

Our analysis is divided into two parts. In Section 5.1, we consider a regular case where all elements of the computed matrix product are numerically significant with respect to the maximal-in-magnitude element; i.e.,

$$\min(|\bar{a}|, |\bar{b}|, |\bar{d}|) > \epsilon \max(|a|, |b|, |d|)\,. \tag{5.2}$$

In Section 5.2, we consider special cases where at least one element of the computed A is numerically insignificant.

5.1. Regular Case

Without loss of generality, we assume that $rb < 0$; i.e., $\mathrm{sign}(r) = -\mathrm{sign}(b)$. Thus we compute t_1 first as the reference tangent from which t_2 and t_3 will be next determined via (4.8a) and (4.8b), respectively. We recall several lemmas from [2].

Lemma 5.1. Let $\bar{t}_1$ and $\tilde{t}_1$ be the exact and computed solutions, respectively, of equation (4.6d) with data $\bar{a}, \bar{b}, \bar{d}$. Moreover, let $\bar{c}_1, \bar{s}_1$ and $\tilde{c}_1, \tilde{s}_1$ be the exact and computed cosines and sines using (4.2) with the tangent value $\tilde{t}_1$. Then

$$\tilde{t}_1 = \bar{t}_1(1 + 10\epsilon_1)\,, \tag{5.3a}$$

$$\tilde{c}_1 = \bar{c}_1(1 + 3\mu_1)\,, \tag{5.3b}$$

$$\tilde{s}_1 = \bar{s}_1(1 + 4\nu_1)\,, \tag{5.3c}$$

where $|\epsilon_1| < \epsilon$, $|\mu_1| < \epsilon$, and $|\nu_1| < \epsilon$.

Proof. See [2]. □

In other words, Lemma 5.1 states that the procedure (4.6a)–(4.6e) for solving (4.5c) is numerically stable in the forward sense. Two lemmas follow, leading to our main result of Theorem 5.1.

Lemma 5.2. The recurrences (4.8a) and (4.8b) yield $\tilde{t}_2$ and $\tilde{t}_3$ such that

$$\bar{a}_1 \tilde{t}_2 - \bar{d}_1 \tilde{t}_1 + b_1 = 0\,, \tag{5.4a}$$

$$\tilde{a}\tilde{t}_3 - \tilde{d}\tilde{t}_1 + \tilde{b} = 0 \,. \tag{5.4b}$$

with

$$\tilde{a}_1 = a_1(1 + 2\epsilon_1) \,,\ \tilde{d}_1 = d_1(1 + \epsilon_1) \,. \tag{5.4c}$$

$$\tilde{a} = \tilde{a}(1 + 2\epsilon) \,,\ \tilde{d} = \tilde{d}(1 + \epsilon) \,. \tag{5.4d}$$

Proof. The proof easily follows from (4.8a) and (4.8b). □

Lemma 5.3. The recurrence (4.8b) yields $\tilde{t}_3$ such that $\tilde{t}_3 = \tilde{t}_3(1 + 13\gamma)$.

Proof. From (4.8b)

$$\tilde{t}_3 = \left(\frac{\tilde{d}\tilde{t}_1(1 + 11\epsilon) - \tilde{b}}{\tilde{a}}\right)(1 + 2\gamma_1) = \left(\frac{\tilde{d}\tilde{t}_1 - \tilde{b}}{\tilde{a}} + \frac{11\epsilon\tilde{d}\tilde{t}_1}{\tilde{a}}\right)(1 + 2\gamma_1) = \left(\tilde{t}_3 + 11\epsilon\tilde{t}_3\frac{\tilde{d}}{\tilde{a}}\frac{\tilde{t}_1}{\tilde{t}_3}\right)(1 + 2\gamma_1) \,.$$

Since $|\tilde{d}/\tilde{a}| \leq 1$ and $|\tilde{t}_1/\tilde{t}_3| \leq 1$, we get $\tilde{t}_3 = \tilde{t}_3(1 + 13\gamma)$. □

We now show that $\tilde{a}'$ and $\tilde{d}'$ are computed with high relative precision.

Theorem 5.1. Let $\tilde{a}'$ and $\tilde{d}'$ be the exact singular values of the computed product $\tilde{A}$. If $\tilde{a}'$ and $\tilde{d}'$ are computed via relations (4.4c) and (4.4d), then the computed singular values a' and d' satisfy the following relations

$$a' = \tilde{a}'(1 + \alpha_4) \,,\ d' = \tilde{d}'(1 + \delta_4) \,. \tag{5.5}$$

Proof. From (4.4f) and (4.4g), we get

$$\tilde{a}' = \tilde{d}(\tilde{t}_1^2 + 1)\tilde{c}_1\tilde{c}_3 \quad \text{and} \quad \tilde{d}' = \tilde{a}(\tilde{t}_3^2 + 1)\tilde{c}_1\tilde{c}_3 \,,$$

where $\tilde{t}_1$ and $\tilde{t}_3$ are the exact tangents corresponding to the data a, b and d and $\tilde{t}_i = \tilde{s}_i/\tilde{c}_i$. Thus, the lemma follows from Lemmas 5.1 and 5.3. □

Theorem 5.2. Suppose that the computed tangent values are $\tilde{t}_1$ and $\tilde{t}_3$. Let $\tilde{c}_1$, $\tilde{s}_1$, $\tilde{c}_3$ and $\tilde{s}_3$ be the corresponding exact cosine and sine values. Let

$$\tilde{\epsilon}' := \tilde{c}_1\tilde{c}_3[-\tilde{a}\tilde{t}_3 + \tilde{d}\tilde{t}_1 - \tilde{b}] \,. \tag{5.6}$$

$$\tilde{b}' := \tilde{c}_1\tilde{c}_3[-\tilde{a}\tilde{t}_1 + \tilde{d}\tilde{t}_3 + b\tilde{t}_1\tilde{t}_3] \,. \tag{5.7}$$

That is, $\tilde{\epsilon}'$ and $\tilde{b}'$ are the exact values of ϵ' and b', respectively, corresponding to the computed data $\tilde{a}$, $\tilde{b}$, $\tilde{d}$, $\tilde{t}_1$ and $\tilde{t}_3$. Then

$$|\tilde{\epsilon}'| \leq K_1\epsilon \ \|\tilde{A}\| \,. \tag{5.8}$$

$$|\tilde{b}'| \leq K_2\epsilon \ \|\tilde{A}\| \,. \tag{5.9}$$

where K_1 and K_2 are some positive constants.

Proof. See Appendix A. □

Theorems 5.1 and 5.2 together state that the SVD of the upper triangular matrix $\tilde{A}$ is computed very accurately. We now justify why the (2,1) element in the computed matrix A_i' can be set to

zero by showing that $|\tilde{e}'_i|$ corresponds to a relative and elementwise perturbation of A'_i of the order of ϵ. Let the cosine and sine pairs $\tilde{c}_i$ and $\tilde{s}_i$ satisfy $t_i = \tilde{s}_i/\tilde{c}_i$, for $i = 1, 2, 3$. From (4.2) we can derive that

$$\tilde{c}_i := \mathrm{fl}(\tilde{c}_i) = \tilde{c}_i(1 + 3\mu_i) , \tag{5.10a}$$

$$\tilde{s}_i := \mathrm{fl}(\tilde{s}_i) = \tilde{s}_i(1 + 4\nu_i) . \tag{5.10b}$$

Let $\tilde{A}'_i$ denote the exact updated matrix derived from A_i, $\tilde{c}_i$, $\tilde{s}_i$, $\tilde{c}_{i+1}$, and $\tilde{s}_{i+1}$. Our next results provide a bound on the element $\tilde{e}'_i$, $i = 1, 2$, defined by the relation

$$\tilde{e}'_i := -\tilde{c}_i\tilde{s}_{i+1}a_i + \tilde{s}_i\tilde{c}_{i+1}d_i - \tilde{c}_i\tilde{c}_{i+1}b_i . \tag{5.11}$$

Theorem 5.3. The matrices $\tilde{A}'_1$ and $\tilde{A}'_2$ are almost upper triangular in that their (2,1) elements $\tilde{e}'_1$ and $\tilde{e}'_1$ satisfy the inequalities:

$$|\tilde{e}'_1| \leq 3\epsilon \, \|A_1\| \tag{5.12a}$$

and

$$|\tilde{e}'_2| \leq K_3\epsilon \, \|A_2\| . \tag{5.12b}$$

Proof. Note that $\tilde{A}'_1$ is the same for both fully-recursive and half-recursive methods. The proof that $\tilde{A}'_1$ is almost upper triangular in the sense that (5.12a) holds can be found in [2].

To prove the second part of the theorem from (5.4a)-(5.4d) and (5.1a)-(5.1c), we get the following two relations to first order of the machine precision:

$$a_1(1 + 2\upsilon_1)\tilde{t}_2 - d_1(1 + \phi_1)\tilde{t}_1 + b_1 = 0 , \tag{5.13a}$$

$$a_1a_2(1 + \alpha + 2\upsilon)\tilde{t}_3 - d_1d_2(1 + \delta + \phi)\tilde{t}_1 + a_1b_2(1 + 2\beta_1) + b_1d_2(1 + 2\beta_2) = 0 . \tag{5.13b}$$

By multiplying both sides of (5.13a) by $d_2(1 + 2\beta_2)$ and subtracting from (5.13b) we obtain

$$a_1\{a_2(1 + \alpha + 2\upsilon)\tilde{t}_3 - \left(\frac{d_1d_2}{a_1}\right)(\delta + \phi - \phi_1 + 2\beta_2)\tilde{t}_1 + b_2(1 + 2\beta_1) - d_2(1 + 2\beta_2 + 2\upsilon_1)\tilde{t}_2\} = 0 ,$$

or, since $a_1 \neq 0$,

$$a_2(1 + \alpha + 2\upsilon)t_3 - \left(\frac{d_1d_2}{a_1}\right)(\delta + \phi - \phi_1 + 2\beta_2)\tilde{t}_1 + b_2(1 + 2\beta_1) - d_2(1 + 2\beta_2 + 2\upsilon_1)\tilde{t}_2$$

$$= a_2\tilde{t}_3 - d_2\tilde{t}_2 + b_2 + \Delta = 0 ,$$

where

$$\Delta = a_2(\alpha + 2\upsilon)t_3 - \left(\frac{d}{a}\right)a_2(\delta + \phi - \phi_1 + 2\beta_2)\tilde{t}_1 + b_2\beta_1 - d_2(2\beta_2 + 2\upsilon_1)\tilde{t}_2 .$$

Thus, we can rewrite (5.11) for $i = 2$ as

$$\tilde{e}'_2 = -\tilde{c}_2\tilde{s}_3a_2 + \tilde{s}_2\tilde{c}_3d_2 - \tilde{c}_2\tilde{c}_3b_2 + \tilde{c}_3\tilde{c}_2(a_2\tilde{t}_3 - d_2\tilde{t}_2 + b_2 + \Delta) . \tag{5.13c}$$

Now, as we start the half-recursive method from t_1, it means that $|\tilde{t}_1| \leq |\tilde{t}_3|$ and $|\tilde{d}| \leq |\tilde{a}|$. Hence from (5.10a), (5.10b) and (5.13c), we derive the inequality:

$$|\tilde{e}'_2| \leq |\tilde{s}_3\tilde{c}_2a_2(\alpha + 2\upsilon)| + |\tilde{c}_3\tilde{c}_2a_2(\delta + \phi - \phi_1 + 2\beta_2)| + |\tilde{c}_3\tilde{c}_2b_2\beta_2| + |\tilde{c}_3\tilde{s}_2d_2(2\beta_2 + 2\psi_1)|$$

$$\leq K_3 \epsilon \|A_2\| ,$$

completing the proof. □

In summary, we have proved two results using backward error analysis. First, the transformed matrix $\bar{A}'$ is almost diagonal in that inequalities (5.8) and (5.9) both hold. Second, we can safely set each computed matrix $\bar{A}_i'$, $i = 1,2$, to a triangular form because (5.12a) and (5.12b) are valid. As a final note, even though we have assumed that $rb < 0$, we can easily prove similar results for the case where $rb \geq 0$.

5.2. Special Cases

In this subsection, we assume that inequality (5.2) is violated. To be specific, define

$$\gamma := \min(|\bar{a}|, |\bar{b}|, |\bar{d}|) \tag{5.14}$$

and

$$\Gamma := \max(|\bar{a}|, |\bar{b}|, |\bar{d}|) . \tag{5.15}$$

Now,

$$\gamma \leq \epsilon \Gamma ; \tag{5.16}$$

i.e., one of the elements of $\bar{A}$ is numerically insignificant. This situation requires modifications to our algorithm, since the proposed formulas may break down. In particular, we do not solve a quadratic equation to determine either $\bar{t}_1$ or $\bar{t}_3$. Instead, we set one of the two tangents to zero and attempt to compute all the other tangents from the recurrences. We divide the special cases into three groups, first,

$$|\bar{a}| + |\bar{d}| \neq 0 \quad \text{and} \quad |\bar{b}| \neq 0 . \tag{5.17}$$

second,

$$|\bar{a}| + |\bar{d}| = 0 \quad \text{and} \quad |\bar{b}| \neq 0 . \tag{5.18}$$

and third,

$$|\bar{b}| = 0 . \tag{5.19}$$

First, assume that (5.17) holds. Hence at least one, but not all, of the following three conditions hold:

$$\gamma = \bar{b}, \quad \gamma = \bar{a} \quad \text{or} \quad \gamma = \bar{d} .$$

We set $\bar{t}_1$ to zero if

$$|\bar{a}| > |\bar{d}| , \tag{5.20}$$

and set $\bar{t}_3$ to zero if

$$|\bar{a}| \leq |\bar{d}| . \tag{5.21}$$

Thus, the sizes of the diagonal elements of $\bar{A}$ will be compared to determine which one of $\bar{t}_1$ or $\bar{t}_3$ should be zeroed. Without loss of generality, assume that (5.20) holds; hence, $\bar{t}_1$ becomes the reference angle. So, $\bar{t}_2$ and $\bar{t}_3$ are computed from recurrence (4.8a) and (4.8b). Further, since $\bar{t}_1 = 0$ it follows that $\bar{t}_3 = -\bar{b}/\bar{a}$. Substituting these values into (5.6) and (5.7), we can verify that Theorem 5.2 holds. Similarly, Theorem 5.3 follows from (5.11). We note that it is very important to decide which reference angle to choose, even for the case when $\bar{b}$ is numerically zero. At first, the choice of the reference angle may seem arbitrary for a "small" $\bar{b}$, since either $\bar{t}_1$ or $\bar{t}_3$ can be set to zero. However, an unnecessarily large error may occur unless we pay special care.

Second, assume that (5.18) holds. Then, at least one of the a_i's equals zero and at least one of the d_j's also equals zero, for $i, j = 1, 2$. A solution is to permute either the rows or the columns in order to ensure that the transformed product is diagonal and that the data are reordered. Hence for this case, we may set the two extreme tangents $\{\tilde{t}_1, \tilde{t}_3\}$ to $\{0, \infty\}$, resulting in the transformations being rotations of negative ninety and zero degrees, respectively. To be specific, consider the case where one or more a_i's equal zero. If $a_1 = 0$, set $\tilde{t}_1 = 0$ and $\tilde{t}_2 = \tilde{t}_3 = \infty$. If $a_1 \neq 0$ and $a_2 = 0$, set $\tilde{t}_1 = 0$, compute $\tilde{t}_2$ from the forward recurrence and set $\tilde{t}_3 = \infty$. Note that we may also choose to determine the tangents using the values of the d_j's.

Third, assume that (5.19) holds. We need to account for the fact that we are really solving an $n \times n$ problem. Although the 2×2 subproblem is already numerically diagonal, it is not sufficient to set $\tilde{t}_1 = \tilde{t}_3 = \infty$, which will leave the 2×2 product unchanged. The $n \times n$ data need to be reordered, calling for $\tilde{t}_1 = \tilde{t}_3 = 0$; i.e., the affected rows and columns will be permuted. Unfortunately, while applying the symmetric permutation, the triangular structures of both $\tilde{A}_1$ and $\tilde{A}_2$ are destroyed. Therefore, $\tilde{t}_2$ is determined from the recurrence.

6. Concluding Remark

In this paper we have presented a simple and accurate way to calculate the PSVD or GSVD of two 2×2 upper triangular matrices. In Appendix C we present an example which shows that our half-recursive method produces identical numerical results as the method in [1]. A significant issue in the design of PSVD algorithms is how to compute the middle transformation. The method used in our half-recursive algorithm is computationally more efficient than the method in [1] and yields identical results. The following table lists the number of floating point operations to compute the three transformations, Q_1, Q_2, and Q_3, for the three different algorithms in the regular case. The column labeled "Simplified Direct" lists the operation count for the Bai and Demmel algorithm if our simplified method is substituted for their method of computing the middle transformation.

Floating-point Operation Counts

	Direct	Simplified Direct	Half-recursive
Addition	29	23	26
Multiplication	57	45	41
Division	13	11	8
Square Root	4	4	4

The Half-recursive method is less expensive than the Direct method and similar in cost to the Simplified Direct algorithm. In addition, the upper-triangular structure of the 2×2 matrices is maintained by the Half-recursive method. Application of the 2×2 Half-recursive algorithm to $n \times n$ problems is a topic for further investigaion.

7. Acknowledgements

G. E. Adams and F. T. Luk were supported in part by the Army Research Office under grant DAAL03-90-G-0104 and the Joint Services Electronics Program under contract F49620-90-C-0039 at Cornell University, and A. W. Bojanczyk by the Army Research Office under grant DAAL03-90-G-0092.

Appendices

A Proof of Theorem 5.2

We first present a lemma.

Lemma A.1. Let $\tilde{\sigma}_1$ and $\tilde{t}_1$ be the exact values corresponding to the given data $\tilde{a}$, $\tilde{b}$ and $\tilde{d}$, and let $\bar{t}_1$ be the computed value of $\tilde{t}_1$. Define a residual r_1 by

$$r_1 := \frac{\bar{b}\bar{d}}{\tilde{a}}(\bar{t}_1^2 + 2\tilde{\sigma}_1\bar{t}_1 - 1) . \tag{A.1}$$

Then

$$| r_1 | \leq K_4\epsilon|\tilde{b}| , \tag{A.2}$$

where K_4 is a positive constant.

Proof. See the proof of Lemma 5.2 in [2]. □

We now have the necessary tools for proving the theorem.

Proof (of Theorem 5.2). First, from Lemma 5.2 and relation (5.4b) we get

$$\tilde{\epsilon}' = \tilde{c}_1\tilde{c}_3[(-\tilde{a}\bar{t}_3 + \tilde{d}\bar{t}_1 - \tilde{b}) + (\tilde{a}\bar{t}_3 - \tilde{d}\bar{t}_1 + \tilde{b})] = (\tilde{a} - \bar{a})\tilde{c}_1\tilde{s}_3 - (\tilde{d} - \bar{d})\tilde{s}_1\tilde{c}_3 .$$

Using (5.1a)-(5.1b) and (5.4d) we prove the inequality:

$$| \tilde{c}' | \leq K\epsilon (| a | + | d |) \leq K_1\epsilon \| \tilde{A} \| . \tag{A.3}$$

Second, rewrite (A.1) as

$$r_1 = \frac{1}{\tilde{a}}[\bar{d}\bar{b}\bar{t}_1^2 + \bar{t}_1(\tilde{d}^2 - \tilde{a}^2 - \tilde{b}^2) - \bar{d}\bar{b}] = \frac{1}{\tilde{a}}[(\bar{d}\bar{t}_1 - \tilde{b})(\bar{b}\bar{t}_1 + \tilde{d}) - \bar{t}_1\tilde{a}^2] . \tag{A.4}$$

From (5.6) we obtain

$$\frac{1}{\tilde{a}}(\bar{d}\bar{t}_1 - \tilde{b}) = \bar{t}_3 + \frac{\tilde{\epsilon}'}{\tilde{c}_1\tilde{c}_3\tilde{a}} . \tag{A.5}$$

Substituting (A.5) into (A.4) and rearranging terms, we get

$$-\tilde{a}\bar{t}_1 + \tilde{d}\bar{t}_3 + \tilde{b}\bar{t}_1\bar{t}_3 = r_1 - \frac{\tilde{\epsilon}'(\tilde{b}\bar{t}_1 + \tilde{d})}{\tilde{c}_1\tilde{c}_3\tilde{a}} ,$$

and so

$$\tilde{b}' = \tilde{c}_1\tilde{c}_3 r_1 - \frac{\tilde{\epsilon}'(\tilde{b}\bar{t}_1 + \tilde{d})}{\tilde{a}} . \tag{A.6}$$

From (4.6d) we derive

$$| \bar{t}_1\tilde{\sigma}_1 | \leq \frac{1}{2} ,$$

and from (4.6b) we get

$$| \tilde{\sigma}_1 | = | \frac{\tilde{r} - \tilde{b}}{2\tilde{d}} | \geq | \frac{\tilde{b}}{2\tilde{d}} | .$$

It follows that

$$| \tilde{t}_1 | \leq | \frac{\tilde{d}}{\tilde{b}} | < | \frac{a}{b} | . \tag{A.7}$$

since we have assumed that $| \tilde{d} | < | \tilde{a} |$. Finally, recall from (5.3) that $t_1 = \tilde{t}_1(1 + 10\epsilon_3)$, and use (A.6), Lemma A.1 and (A.5) to obtain

$$| \tilde{b}' | \leq \tilde{c}_1 \tilde{c}_2 | r_1 | + 2 | \tilde{\epsilon}' | \leq K_2 \epsilon \| A \| , \tag{A.8}$$

thus completing the proof. □

B How to Compute the Middle Transformation

As pointed out by Bai and Demmel in [1], a critical issue concerns how the middle transformation should be computed. They proposed the following scheme for its computation after both end transformations have been determined. In order to relate the test for computing Q_2 in [1] to the test in the half-recursive method, we first translate our setting to that in [1]. Let

$$U^T \equiv \begin{pmatrix} c_1 & -s_1 \\ s_1 & c_1 \end{pmatrix} , \quad Q^T \equiv \begin{pmatrix} c_2 & -s_2 \\ s_2 & c_2 \end{pmatrix} \quad \text{and} \quad V^T \equiv \begin{pmatrix} c_3 & -s_3 \\ s_3 & c_3 \end{pmatrix} .$$

Note that the relation, given by

$$Q_1 A_1 = \begin{pmatrix} s_1 & c_1 \\ -c_1 & s_1 \end{pmatrix} \begin{pmatrix} a_1 & b_1 \\ 0 & d_1 \end{pmatrix} = \begin{pmatrix} s_1 a_1 & s_1 b_1 + c_1 d_1 \\ -c_1 a_1 & -c_1 b_1 + s_1 d_1 \end{pmatrix} \tag{B.1a}$$

upon permuting rows and changing the signs of the top row, is equivalent to

$$U^T A_1 = \begin{pmatrix} c_1 & -s_1 \\ s_1 & c_1 \end{pmatrix} \begin{pmatrix} a_1 & b_1 \\ 0 & d_1 \end{pmatrix} = \begin{pmatrix} c_1 a_1 & c_1 b_1 - s_1 d_1 \\ s_1 a_1 & s_1 b_1 + c_1 d_1 \end{pmatrix} \equiv G . \tag{B.1b}$$

Similarly,

$$A_2 Q_3^T = \begin{pmatrix} a_2 & b_2 \\ 0 & d_2 \end{pmatrix} \begin{pmatrix} s_3 & -c_3 \\ c_3 & s_3 \end{pmatrix} = \begin{pmatrix} s_3 a_2 + c_3 b_2 & -c_3 a_2 + s_3 b_2 \\ c_3 d_2 & s_3 d_2 \end{pmatrix} . \tag{B.2a}$$

By changing the sign of the second columns and permuting columns, we obtain

$$V^T adj(A_2) = \begin{pmatrix} c_3 & -s_3 \\ s_3 & c_3 \end{pmatrix} \begin{pmatrix} d_2 & -b_2 \\ 0 & a_2 \end{pmatrix} = \begin{pmatrix} c_3 d_2 & -c_3 b_2 - s_3 a_2 \\ s_3 d_2 & -s_3 b_2 + c_3 a_2 \end{pmatrix} \equiv H . \tag{B.2b}$$

In [1], Bai and Demmel used (B.1b) and (B.2b) as a starting point for computing Q_2. Their argument is as follows. After postmultiplications of both (B.1b) and (B.2b) by Q_2, the (1,2) elements of G and H should become zeros. Now, one should compute Q_2 from the one product, either G or H, for which the computed element in the (1,2) position has a smaller error relative to the norm of the row in which it resides. The magnitude of that error can be only bounded and hence the test for the choice is based on the bounds of the errors. It is easy to see that the bound g for the relative error in the (1,2) element of the computed G is

$$g = \frac{|c_1 b_1| + |s_1 d_1|}{|c_1 a_1| + |c_1 b_1 - s_1 d_1|} . \tag{B.3a}$$

while the bound h for the relative error in the (1,2) element of the computed H is

$$h = \frac{|c_3 b_2| + |s_3 a_2|}{|c_3 d_2| + |c_3 b_2 + s_3 a_2|} . \tag{B.3b}$$

Now if $g \leq h$, then Bai and Demmel compute Q_2 from $U^T A$ and otherwise from $V^T B$. The next lemma shows that the conditions specifying how Q_2 is computed by Bai and Demmel and by the half-recursive method are essentially equivalent.

Lemma B.1. In exact arithmetic, the condition

$$g \leq h \tag{B.4a}$$

where g is defined by (B.3a) and h is defined by (B.3b) is equivalent to the condition

$$|a| \geq |d|. \tag{B.4b}$$

Proof. First note that (B.3a) and (B.3b) can be simplified to

$$g = \frac{|b_1| + |t_1 d_1|}{|a_1| + |t_1 d_1 - b_1|} \tag{B.5a}$$

and

$$h = \frac{|b_2| + |t_3 a_2|}{|d_2| + |t_3 a_2 + b_2|} , \tag{B.5b}$$

respectively. Through (4.8a) and (4.8c) the relations (B.3a) and (B.3b) simplify further to

$$g = \frac{|b_1| + |t_1 d_1|}{|a_1|(1 + |t_2|)} \tag{B.6a}$$

and

$$h = \frac{|b_2| + |t_3 a_2|}{|d_2|(1 + |t_2|)} , \tag{B.6b}$$

respectively. Hence (B.4a) is equivalent to

$$|b_1 d_2| + |t_1 d| \leq |a_1 b_2| + |a t_3| . \tag{B.7}$$

We now prove that (B.4b) implies (B.4a). The proof that $|a| < |d|$ implies that $g < h$ is analogous and is omitted. Our proof is elementary but tedious as it requires us to consider a large number of cases. Assume that $|a| \geq |b|$. Then Lemma 4.1 implies that $t_3 \geq t_1$. From (4.8b) we see that

$$|a t_3 + b| = |d t_1|$$

and as $|a t_3| \geq |a t_1|$ we conclude that

$$\operatorname{sign}(a t_3) = -\operatorname{sign}(b) = -\operatorname{sign}(a_1 b_2 + b_1 d_2) , \tag{B.8}$$

as from (4.7b) $b = a_1 b_2 + b_1 d_2$. Substituting (4.8b) into (B.7) and using (4.7b) again we get that (B.7) is equivalent to the following inequality:

$$|b_1 d_2| + |a t_3 + a_1 b_2 + b_1 d_2| \leq |a_1 b_2| + |a t_3| . \tag{B.9}$$

Case 1. $-|b| \geq |b_1 d_2| - |a_1 b_2|$.

Then

$$|at_3| \geq |dt_1| - |b| \geq |dt_1| + |b_1 d_2| - |a_1 b_2| \,.$$

establishing (B.7).

Case 2a. $-|b| > |b_1 d_2| - |a_1 b_2|$ and $|at_3| > |b|$.

Then $|a_1 b_2| > |b_1 d_2|$ and using (B.8) we obtain that

$$|b_1 d_2| + |dt_1| = |b_1 d_2| + |at_3 + a_1 b_2 + b_1 d_2| = |at_3| + 2\,|b_1 d_2| - |a_1 b_2| \,.$$

from which (B.7) follows.

Case 2b. $-|b| > |b_1 d_2| - |a_1 b_2|$ and $|at_3| \leq |b|$.

Then again $|a_1 b_2| > |b_1 d_2|$. Now from (B.8)

$$|b_1 d_2| + |dt_1| = |b_1 d_2| + |at_3 + a_1 b_2 + b_1 d_2|$$

$$= |b_1 d_2| - |at_3| + |a_1 b_2| - |b_1 d_2| = |a_1 b_2| - |at_3| \,.$$

from which (B.7) again follows.

Remark. Note that there might be a slight difference in using (B.4a) or (B.4b) as the lemma holds only in exact arithmetic. In finite precision computation, the relations (B.4a) and (B.4b) may not always be equivalent. However, we have not been able to find any numerical example where these two conditions are not equivalent. Moreover, as shown in this paper, the consequences of numerical non-equivalence are numerically insignificant.

C Numerical Example

It has been proved in Appendix B that the half-recursive procedure computes essentially the same numerical results as the direct method of [1]. For both methods, the end transformations are computed explicitly from the product $A = A_1 A_2$, and the middle transformation is computed from the same direction. The greatest difference between the fully-recursive method and the other two occurs when there is cancellation in forming the product $A = A_1 A_2$. In the following PSVD example, A_1 and A_2 each has an $O(1)$ norm, but the product $A_1 A_2$ has an $O(10^{-5})$ norm. Hence errors which are small relative to the initial matrices may be large relative to the product.

$$A_1 = \begin{pmatrix} 2.316797292247488e+00 & -1.437687878748196e-01 \\ 0 & -5.208536329107726e-06 \end{pmatrix}$$

$$A_2 = \begin{pmatrix} 2.472499811756353e-05 & 2.624474233535929e-01 \\ 0 & 4.229273187671001e+00 \end{pmatrix}$$

$$A_1 A_2 = \begin{pmatrix} 5.728280868959543e-05 & -1.110223024625157e-16 \\ 0 & -2.202832304370565e-05 \end{pmatrix}$$

The three methods all compute the left transformation from the explicit product and calculate the middle transformation from A_1. We use the subscripts dir, hr, and fr to distinguish between results computed via the direct, half-recursive, and fully-recursive methods, respectively. The computed values of $A'_{1,dir}$, $A'_{1,hr}$, and $A'_{1,fr}$ are *numerically identical* in that the corresponding entries are numerically equal:

$$A'_{1\,dir} = \begin{pmatrix} 2.321253790033e780e+00 & 2.775557561502891e-17 \\ 3.225938007089203e-07 & -5.198536633811768e-06 \end{pmatrix}$$

$$A'_{1\,hr} = \begin{pmatrix} -5.198536633811768e-06 & -3.225938007089203e-07 \\ -2.775557561502891e-17 & 2.321253790033078e+00 \end{pmatrix}$$

$$A'_{1\,fr} = \begin{pmatrix} -5.198536633811768e-06 & -3.225938007089203e-07 \\ -2.775557561502891e-17 & 2.321253790033078e+00 \end{pmatrix}$$

The computed matrices $A'_{2\,dir}$, $A'_{2\,hr}$, and $A'_{2\,fr}$ are numerically triangular, but now the (1,2) element of $A'_{2\,fr}$ is significantly different from the corresponding elements in $A'_{1\,dir}$ and $A'_{1\,hr}$.

$$A'_{2\,dir} = \begin{pmatrix} 2.467752941777026e-05 & 5.551115123125783e-17 \\ 1.531353724707708e-06 & 4.237408440201395e+00 \end{pmatrix}$$

$$A'_{2\,hr} = \begin{pmatrix} 4.237408440201395e+00 & -1.531353724707708e-06 \\ -5.551115123125783e-17 & 2.467752941777026e-05 \end{pmatrix}$$

$$A'_{2\,fr} = \begin{pmatrix} 4.237408440201395e+00 & -1.531363802644679e-06 \\ 0 & 2.467752941777026e-05 \end{pmatrix}$$

To maintain triangularity, A'_1 and A'_2 are truncated by setting the appropriate elements to zero. Let A''_1 and A''_2 denote the truncated matrices. The product $A'' = A''_1 \cdot A''_2$ should be diagonal.

$$A''_{dir} = \begin{pmatrix} 5.728280868950542e-05 & 0 \\ 1.615587133892632e-27 & -2.202832304370564e-05 \end{pmatrix}$$

$$A''_{hr} = \begin{pmatrix} -2.202832304370564e-05 & -1.615587133892632e-27 \\ 0 & 5.728280868950542e-05 \end{pmatrix}$$

$$A''_{fr} = \begin{pmatrix} -2.202832304370564e-05 & 5.010342801562901e-17 \\ 0 & 5.728280868950542e-05 \end{pmatrix}$$

Clearly, A''_{hr} and A''_{dir} are numerically diagonal, but A''_{fr} fails the criterion of diagonality. Forcing A''_{fr} to be a diagonal matrix requires a truncation of $O(10^{-17})$, which is significant with respect to $\|A''\|$. The matrices A''_{dir} and A''_{hr} require only insignificant truncations to obtain diagonality, but we have previously made $O(10^{-17})$ truncations during their computation to force $A''_{2\,dir}$ and $A''_{2\,hr}$ to triangular forms. Thus, equal amounts of absolute truncation errors have been committed by all three methods. The only difference is that the relative truncation error is largest for the fully-recursive method.

It is interesting to note that if triangularity is not enforced and the factors A'_1 and A'_2 are multiplied, then none of the products can be considered diagonal. One may say that the numerical diagonality of A''_{hr} and A''_{dir} is a consequence of the truncation to triangular forms.

$$A'_{1\,dir} \cdot A'_{2\,dir} = \begin{pmatrix} 5.728280868950542e-05 & 2.464671807471544e-16 \\ 1.615587133892632e-27 & -2.202832304370564e-05 \end{pmatrix}$$

$$A'_{1\,hr} \cdot A'_{2\,hr} = \begin{pmatrix} -2.202832304370564e-05 & -1.615587133892632e-27 \\ -2.464671807471544e-16 & 5.728280868950542e-05 \end{pmatrix}$$

$$A'_{1\,fr} \cdot A'_{2\,fr} = \begin{pmatrix} -2.202832304370564e-05 & 5.010342801562901e-17 \\ -1.176117105626251e-16 & 5.728280868950542e-05 \end{pmatrix}$$

In conclusion, our example shows that the half-recursive and direct methods produce numerically identical results, while the fully-recursive method fails to meet the diagonality criterion.

8. References

[1] Z. Bai and J.W. Demmel, "Computing the generalized singular value decomposition," Report No UCB/CSD 91/645, Computer Science Division, University of California, Berkeley, August 1991.

[2] A.W. Bojanczyk, L.M. Ewerbring, F.T. Luk and P. Van Dooren, "An accurate product SVD algorithm," *Signal Processing*, 25 (1991), pp.189–201.

[3] J. P. Charlier, M. Vanbegin and P. Van Dooren, "On efficient implementations of Kogbetliantz's algorithm for computing the singular value decomposition," *Numer. Math.*, 52 (1988), pp 279–300.

[4] K. V. Fernando and S. J. Hammarling, "A product induced singular value decomposition for two matrices and balanced realisation," in *Linear Algebra in Signals, Systems and Control*, B. N. Datta et al., Eds., SIAM, Philadelphia, 1988, pp. 128–140.

[5] M. T. Heath, A. J. Laub, C. C. Paige, and R. C. Ward, "Computing the SVD of a product of two matrices," *SIAM J. Sci. Statist. Comput.*, 7 (1986), pp. 1147–1159.

[6] C. C. Paige, "Computing the generalized singular value decomposition," *SIAM J. Sci. Statist. Comput.*, 7 (1986), pp. 1126–1146.

Analysis of a Linearly Constrained Least Squares Algorithm for Adaptive Beamforming

Franklin T. Luk
Computer Science Department, Rensselaer Polytechnic Institute
Troy, New York 12180

Sanzheng Qiao
Communications Research Laboratory, McMaster University
Hamilton, Ontario L8S 4K1 Canada

ABSTRACT

The problem of linearly constrained least squares has many applicati ns in signal processing. In this paper, we present a perturbation analysis of a linearly constrained least squares algorithm for adaptive beamforming. The perturbation bounds for the solution as well as for the latest residual element are derived. We also propose an error estimation scheme for the residual element, which can be incorporated into a systolic array implementation of the algorithm.

1. INTRODUCTION

The least squares problem with linear equality constraints has important applications in signal processing, e.g., adaptive beamforming. To solve this problem, McWhirter and Shepherd [5] proposed a systolic algorithm and architecture. In this paper, we present a perturbation analysis of the problem and propose an error estimation scheme for the McWhirter-Shepherd (MS) algorithm [5]. This paper is organized as follows. The least squares problem is defined in Section 2 and error bounds are derived in Section 3. An error estimation algorithm is given in Section 4, and in Section 5 a numerical example is presented to illustrate how well our new algorithm works.

2. PROBLEM DEFINITION

Given an $n \times q$ complex data matrix $X(n)$, the least squares problem with linear equality constraints is to find a q-element complex vector $w(n)$ such that

$$\|X(n)w(n)\| = \min \tag{2.1a}$$

subject to the linear constraints

$$Sw(n) = b, \tag{2.1b}$$

where S is a $k \times q$ $(k < q)$ complex matrix and b is a k-element complex vector. Throughout this paper, we use the 2-norm:

$$\|\cdot\| = \|\cdot\|_2.$$

In signal processing, new data arrives continuously. Define the data matrix $X(n)$ recursively by

$$X(n) \equiv \begin{pmatrix} X(n-1) \\ x(n)^T \end{pmatrix},$$

i.e., the nth row $x(n)^T$ represents a snapshot at time n. Our goal is to compute the n-th residual element

$$r_n = x(n)^T w(n). \tag{2.2}$$

0-8194-0943-X/92/$4.00

Is the solution vector $w(n)$ unique? Define a $(k+n) \times q$ matrix $S_X(n)$ by

$$S_X(n) \equiv \begin{pmatrix} S \\ X(n) \end{pmatrix}.$$

We assume that $k + n \geq q$. The solution is unique if and only if the matrix $S_X(n)$ has full column rank, that is, the overdetermined matrix equation

$$S_X(n)w(n) = 0 \tag{2.3}$$

has a unique solution $w(n) = 0$.

Next, we wish to transform (2.1) into a familiar unconstrained problem; see [3] and [4]. Let

$$p = q - k$$

and partition the matrix S as

$$S = (\, S_1 \quad S_2 \,),$$

where S_1 is $k \times k$ and S_2 is $k \times p$. For simplicity, we assume that S_1 is nonsingular and upper triangular: for example, S_1 may be the result of an initial QR decomposition of S. Accordingly, we also partition $X(n)$ as

$$X(n) = (\, X_1(n) \quad X_2(n) \,),$$

so that X_1 is $n \times k$ and X_2 is $n \times p$. Then (2.3) becomes

$$\begin{pmatrix} S_1 & S_2 \\ X_1(n) & X_2(n) \end{pmatrix} w(n) = 0,$$

which is equivalent to

$$\begin{pmatrix} S_1 & S_2 \\ 0 & C(n) \end{pmatrix} w(n) = 0,$$

where

$$C(n) \equiv X_2(n) - X_1(n)S_1^{-1}S_2.$$

The matrix $C(n)$ is called the Schur complement of S_1 in S_X. The equation (2.3) has the trivial solution if and only if $C(n)$ has full column rank. We proceed to eliminate the constraints. Let

$$w(n) = \begin{pmatrix} w_1(n) \\ w_2(n) \end{pmatrix},$$

so that $w_1(n)$ is $k \times 1$ and $w_2(n)$ is $p \times 1$. Since

$$S_1 w_1(n) + S_2 w_2(n) = b,$$

we get

$$w_1(n) = S_1^{-1}b - S_1^{-1}S_2 w_2(n). \tag{2.4}$$

Let

$$v(n) = -X_1(n)S_1^{-1}b.$$

We derive

$$\|C(n)w_2(n) - v(n)\| = \min, \tag{2.5}$$

an unconstrained problem analyzed in [3], [4]. Now, what about the residual element r_n? Define the Schur complement matrix $C(n)$ recursively by

$$C(n) \equiv \begin{pmatrix} C(n-1) \\ c(n)^T \end{pmatrix}.$$

Partition the row vector $x(n)^T$ so that

$$x(n)^T = (x_1(n)^T \quad x_2(n)^T),$$

where $x_1(n)^T$ is $1 \times k$ and $x_2(n)^T$ is $1 \times p$. We get

$$c(n)^T = x_2(n)^T - x_1(n)^T S_1^{-1} S_2.$$

Let v_n denote the n-th element of $v(n)$. The last residual element of (2.5) is then

$$c(n)^T w_2(n) - v_n = x_2(n)^T w_2(n) + x_1(n)^T w_1(n) + v_n - v_n = r_n,$$

i.e., the same residual element as desired by the constrained problem (2.1).

How do we calculate r_n recursively? Suppose that we have available a QR decomposition of the $(n-1) \times p$ matrix $C(n-1)$:

$$C(n-1) = Q(n-1)R(n-1),$$

where $Q(n-1)$ is $(n-1) \times p$ with orthonormal columns and the matrix $R(n-1)$ is $p \times p$ upper triangular. The problem (2.5) is reduced to

$$\left\| \begin{pmatrix} R(n-1) \\ c(n)^T \end{pmatrix} w_2(n) - \begin{pmatrix} u(n-1) \\ v_n \end{pmatrix} \right\| = \min,$$

where $u(n-1) = Q(n-1)^H v(n-1)$. We triangularize the coefficient matrix by a unitary matrix P. Then

$$P^H \begin{pmatrix} R(n-1) & u(n-1) \\ c(n)^T & v_n \end{pmatrix} = \begin{pmatrix} R(n) & u(n) \\ 0^T & \gamma \end{pmatrix},$$

so that $R(n)$ is $p \times p$ upper triangular. The matrix P consists of p Givens matrices. From P and $Q(n-1)$ we can construct an $n \times p$ orthonormal matrix $Q(n)$ such that $C(n) = Q(n)R(n)$ and $u(n) = Q(n)^H v(n)$. The desired element r_n is given by

$$r_n = -(c_1 \ldots c_p)\gamma,$$

where $c_1, \ldots, c_p$ denote cosines of the p rotations that make up P.

3. PERTURBATION ANALYSIS

Eldén [1] presented a perturbation analysis of the linearly constrained least squares problem. Since his theory is general, it involves weighted pseudoinverses and their corresponding condition numbers. In this section, we derive simpler perturbation bounds for the solution $w(n)$ as well as for the residual element r_n. To simplify our presentation, we will drop the argument (n) for the matrices and vectors, and let $\kappa(M)$ denote the condition number of a matrix M with respect to the 2-norm.

Let $\hat{w}$ solve the perturbed least squares problem

$$\| (X_1 + \epsilon E_{X_1} \quad X_2 + \epsilon E_{X_2}) \hat{w} \| = \min \tag{3.1a}$$

subject to the perturbed linear equality constraints

$$(S_1 + \epsilon E_{S_1} \quad S_2 + \epsilon E_{S_2}) \hat{w} = b + \epsilon f_b. \tag{3.1b}$$

Suppose that $t \geq 0$ is a real variable and let

$$C + tE_C = (X_2 + tE_{X_2}) - (X_1 + tE_{X_1})(S_1 + tE_{S_1})^{-1}(S_2 + tE_{S_2})$$

and

$$v + tf_v = -(X_1 + tE_{X_1})(S_1 + tE_{S_1})^{-1}(b + tf_b).$$

Recall that S_1 is nonsingular and that C has full column rank. Suppose ϵ is sufficiently small so that for $t \in [0, \epsilon]$ we have $S_1 + tE_{S_1}$ is nonsingular and $C + tE_C$ has full column rank. Let $w(t)$ solve the matrix equation

$$\begin{pmatrix} S_1 + tE_{S_1} & S_2 + tE_{S_2} \\ 0 & (C + tE_C)^H (C + tE_C) \end{pmatrix} w(t) = \begin{pmatrix} b + tf_b \\ (C + tE_C)^H (v + tf_v) \end{pmatrix}. \tag{3.2}$$

Then $w(0)$ and $w(\epsilon)$ are solutions to problems (2.1) and (3.1), respectively. Define $w \equiv w(0)$ and $\hat{w} \equiv w(\epsilon)$. Then

$$\hat{w} = w(0) + \epsilon \dot{w}(0) + O(\epsilon^2).$$

Differentiate (3.2) with respect to t and set $t = 0$. We get

$$\begin{pmatrix} E_{S_1} & E_{S_2} \\ 0 & E_C^H C + C^H E_C \end{pmatrix} w(0) + \begin{pmatrix} S_1 & S_2 \\ 0 & C^H C \end{pmatrix} \dot{w}(0) = \begin{pmatrix} f_b \\ E_C^H v + C^H f_v \end{pmatrix}. \tag{3.3}$$

Let

$$\tilde{S} \equiv \begin{pmatrix} S_1 & S_2 \\ 0 & I \end{pmatrix}, \quad \tilde{C} \equiv \begin{pmatrix} I & 0 \\ 0 & C \end{pmatrix}, \quad d \equiv \begin{pmatrix} b \\ v \end{pmatrix} \quad \text{and} \quad f_d \equiv \begin{pmatrix} f_b \\ f_v \end{pmatrix}.$$

Then

$$\begin{pmatrix} S_1 & S_2 \\ 0 & C^H C \end{pmatrix}^{-1} = \tilde{S}^{-1} (\tilde{C}^H \tilde{C})^{-1} \quad \text{and} \quad \|C\| \le \|\tilde{C}\|.$$

Solving for $\dot{w}(0)$ in (3.3), we obtain

$$\begin{aligned} \dot{w}(0) &= \begin{pmatrix} S_1 & S_2 \\ 0 & C^H C \end{pmatrix}^{-1} \left[\begin{pmatrix} f_b \\ C^H f_v \end{pmatrix} - \begin{pmatrix} E_{S_1} & E_{S_2} \\ 0 & C^H E_C \end{pmatrix} w + \begin{pmatrix} 0 \\ E_C^H v \end{pmatrix} - \begin{pmatrix} 0 & 0 \\ 0 & E_C^H C \end{pmatrix} w \right] \\ &= \tilde{S}^{-1} (\tilde{C}^H \tilde{C})^{-1} \tilde{C}^H \left[f_d - \begin{pmatrix} E_{S_1} & E_{S_2} \\ 0 & E_C \end{pmatrix} w \right] - \tilde{S}^{-1} (\tilde{C}^H \tilde{C})^{-1} \begin{pmatrix} 0 \\ E_C^H r \end{pmatrix}, \end{aligned} \tag{3.4}$$

where $r \equiv Cw_2 - v$ denotes the residual vector. Furthermore, by assuming

$$\|f_b\| \le \|b\|, \quad \|f_v\| \le \|v\|, \quad \|E_C\| \le \|C\| \tag{3.5a}$$

and

$$\left\| \begin{pmatrix} E_{S_1} & E_{S_2} \\ 0 & E_C \end{pmatrix} \right\| \le \left\| \begin{pmatrix} S_1 & S_2 \\ 0 & C \end{pmatrix} \right\| \le \|\tilde{S}\| \, \|\tilde{C}\|, \tag{3.5b}$$

we derive the inequality

$$\|\dot{w}(0)\| \le \|\tilde{S}^{-1}\| \, \|(\tilde{C}^H \tilde{C})^{-1} \tilde{C}^H\| \left[\|d\| + \|\tilde{C}\| \, \|\tilde{S}\| \, \|w\| \right] + \|\tilde{S}^{-1}\| \, \|(\tilde{C}^H \tilde{C})^{-1}\| \, \|\tilde{C}\| \, \|r\|.$$

Consequently, we obtain the following perturbation result.

Lemma. *Using the notations defined above and assuming that ϵ in (3.1) is sufficiently small so that the inequalities (3.5) are satisfied,* we get

$$\frac{\|\hat{w} - w\|}{\|w\|} \le \epsilon \left\{ \kappa(\tilde{S}) \kappa(\tilde{C}) \left(\frac{\|d\|}{\|\tilde{C}\| \, \|\tilde{S}\| \, \|w\|} + 1 \right) + \kappa(\tilde{S}) \kappa(\tilde{C})^2 \frac{\|r\|}{\|\tilde{C}\| \, \|\tilde{S}\| \, \|w\|} \right\} + O(\epsilon^2). \ \square \tag{3.6}$$

To illustrate the effect of $\kappa(\tilde{S})$ on the solution of (2.1), consider a simple example in which $S = (S_1 \quad 0)$ and $X = (I_n \quad I_n)$, where I_n is an $n \times n$ identity matrix and $n \le k < 2n$. By observation, $w_1 = S_1^{-1} b$ and $w_2 = -w_1$. Since $\kappa(\tilde{S}) = \kappa(S_1)$ in this example, we see why the presence of $\kappa(\tilde{S})$ is necessary in (3.6).

We proceed to derive a bound for the error in the residual. Let

$$\begin{pmatrix} 0 \\ r(t) \end{pmatrix} = \begin{pmatrix} S_1 + tE_{S_1} & S_2 + tE_{S_2} \\ 0 & C + tE_C \end{pmatrix} w(t) - \begin{pmatrix} b + tf_b \\ v + tf_v \end{pmatrix},$$

differentiate the equation, and then set $t = 0$. Using (3.4) to substitute for $\dot{w}(0)$, we get

$$\begin{aligned} \begin{pmatrix} 0 \\ \dot{r}(0) \end{pmatrix} &= \begin{pmatrix} E_{S_1} & E_{S_2} \\ 0 & E_C \end{pmatrix} w + \begin{pmatrix} S_1 & S_2 \\ 0 & C \end{pmatrix} \dot{w}(0) - f_d \\ &= (I - \tilde{C}\tilde{C}^\dagger) \left[\begin{pmatrix} E_{S_1} & E_{S_2} \\ 0 & E_C \end{pmatrix} w - f_d \right] - \tilde{C}(\tilde{C}^H \tilde{C})^{-1} \begin{pmatrix} 0 \\ E_C^H r \end{pmatrix}, \end{aligned}$$

where $\tilde{C}^\dagger = (\tilde{C}^H \tilde{C})^{-1} \tilde{C}^H$. Consequently,

$$\dot{r}(0) = (I - CC^\dagger)(E_C w_2 - f_v) - C(C^H C)^{-1} E_C^H r.$$

As for the residual element we have $r_n = e_n^T r$, where $e_n \equiv (0, \ldots, 0, 1)$ denotes the n-th unit coordinate vector. Using the assumptions (3.5a) and noticing that $w_2 = C^\dagger v$ and $r = (I - CC^\dagger)v$, we derive our major result.

Theorem. *Under the same conditions as in the Lemma, we get*

$$\frac{\|\hat{r} - r\|}{\|v\|} \leq \epsilon \left[\|I - CC^\dagger\| (2\kappa(C) + 1) \right] + O(\epsilon^2) \tag{3.7}$$

and

$$\frac{|\hat{r}_n - r_n|}{\|v\|} \leq \epsilon \left[\|I - CC^\dagger\| (\kappa(C) + \|C\| \, \|C^\dagger e_n\| + 1) \right] + O(\epsilon^2). \quad \square \tag{3.8}$$

Here are some additional remarks. If we set $\tilde{S} = I$ and $b = 0$, then (3.6) leads to a perturbation bound for the standard least squares problem [2]. We also note that $\|\tilde{C}\tilde{S}w\|^2 + \|r\|^2 = \|d\|^2$. Thus, we can define

$$\cos\theta = \|\tilde{C}\tilde{S}w\| / \|d\|$$

and use $(1/\cos\theta)$ and $\tan\theta$ in (3.6). The bound (3.7) is similar to a result derived in [2]. The inequality (3.8) indicates that $|\hat{r}_n - r_n|$ depends on $\kappa(C)$ as well as on $\|v\|$. Both (3.7) and (3.8) can be simplified by using the relation that $\|I - CC^\dagger\| = \min\{1, \, n - p\}$.

4. ERROR ESTIMATION

Although the error bound (3.8) is simple, it requires $C^\dagger e_n$, whose computation involves at least a back-solve. In this section, we present an error estimation scheme for the desired residual element. When the new data vector $x(n)^T$ arrives, it is first processed by S so that $x_1(n)^T$ is annihilated. In particular, let

$$(z_1^{(0)} \ \ldots \ z_q^{(0)}) = x(n)^T \quad \text{and} \quad u^{(0)} = 0.$$

Then the preprocessing proceeds as follows:

$$\begin{pmatrix} s_{l,l} & s_{l,l+1} & \cdots & s_{l,q} \\ 0 & z_{l+1}^{(l)} & \cdots & z_q^{(l)} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ -g_l & 1 \end{pmatrix} \begin{pmatrix} s_{l,l} & s_{l,l+1} & \cdots & s_{l,q} \\ z_l^{(l-1)} & z_{l+1}^{(l-1)} & \cdots & z_q^{(l-1)} \end{pmatrix}$$

and

$$\begin{pmatrix} b_l \\ u^{(l)} \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ -g_l & 1 \end{pmatrix} \begin{pmatrix} b_l \\ u^{(l-1)} \end{pmatrix},$$

for $l = 1, 2, \ldots, k$, where $g_l = z_l^{(l-1)}/s_{l,l}$. Writing in algorithmic form, we have

for $l = 1, 2, \ldots, k$

begin

$g_l = z_l^{(l-1)}/s_{l,l}$;

for $j = l+1, \ldots, q$

$z_j^{(l)} = z_j^{(l-1)} - g_l s_{l,j}$;

$u^{(l)} = u^{(l-1)} - g_l b_l$

end.

The above process shows that

$$(z_{k+1}^{(k)} \ \ldots \ z_q^{(k)}) = x_2(n)^T - x_1(n)^T S_1^{-1} S_2 \quad \text{and} \quad u^{(k)} = -x_1(n)^T S_1^{-1} b.$$

These two variables are then used for updating the QR decomposition of $C(n-1)$ and computing the residual element. We present below the algorithm derived in [4].

for $l = 1, 2, \ldots, p$

begin

$c_{l,l}^{(n)} = \sqrt{|c_{l,l}^{(n-1)}|^2 + |z_{k+l}^{(k+l-1)}|^2}$;

$\cos\theta_l = c_{l,l}^{(n-1)}/c_{l,l}^{(n)}$;

$\sin\theta_l = z_{k+l}^{(k+l-1)}/c_{l,l}^{(n)}$;

for $j = l+1, \ldots, p$

begin

$c_{l,j}^{(n)} = c_{l,j}^{(n-1)} \cos\theta_l + z_{k+j}^{(k+l-1)} \sin\theta_l$;

$z_{k+j}^{(k+l)} = -c_{l,j}^{(n-1)} \sin\theta_l + z_{k+j}^{(k+l-1)} \cos\theta_l$

end;

$v_l^{(n)} = v_l^{(n-1)} \cos\theta_l + u^{(k+l-1)} \sin\theta_l$;

$u^{(k+l)} = -v_l^{(n-1)} \sin\theta_l + u^{(k+l-1)} \cos\theta_l$

end;

$r_n = u^{(k+p)} \prod_{i=1}^{p} \cos\theta_i$.

In the above, $c_{i,j}^{(k)}$ (for $k = n-1, n$) denotes the (i,j)-element of $C(k)$ and $v_i^{(k)}$ the i-th element of $v(k)$.

Now, we discuss an error estimation scheme for the preprocessing. Let $\hat{}$ denote the corresponding computed value and fl the floating point computation. In the above procedure we calculate

$$\hat{g}_l = fl(\hat{z}_l^{(l-1)}/\hat{s}_{l,l}),$$
$$\hat{z}_j^{(l)} = fl(\hat{z}_j^{(l-1)} - fl(\hat{g}_l \hat{s}_{l,j})),$$
$$\hat{u}^{(l)} = fl(\hat{u}^{(l-1)} - fl(\hat{g}_l \hat{m}_l)).$$

Define the relations between the exact and computed quantities as follows:

$$\hat{s}_{i,j} = s_{i,j}(1 + \sigma_{i,j}\phi_{i,j}(\epsilon)),$$
$$\hat{z}_j^{(l)} = z_j^{(l)}(1 + \zeta_j^{(l)}\psi_j^{(l)}(\epsilon)),$$
$$\hat{g}_l = g_l(1 + \alpha_l \xi_l(\epsilon)),$$
$$\hat{u}^{(l)} = u^{(l)}(1 + \eta^{(l)}\theta^{(l)}(\epsilon)),$$
$$\hat{b}_l = b_l(1 + \mu_l \delta_l(\epsilon)),$$

where $|\phi_{i,j}(\epsilon)| = O(\epsilon)$, $|\psi_j^{(l)}(\epsilon)| = O(\epsilon)$, $|\xi_l(\epsilon)| = O(\epsilon)$, $|\theta^{(l)}(\epsilon)| = O(\epsilon)$ and $|\delta_l(\epsilon)| = O(\epsilon)$. The five quantities $\sigma_{i,j}$, $\zeta_j^{(l)}$, α_l, $\eta^{(l)}$ and μ_l are all real and nonnegative. We also assume that the errors such as $\sigma_{i,j}\phi_{i,j}(\epsilon)$ and $\zeta_j^{(l)}\psi_j^{(l)}(\epsilon)$ are so small that higher order terms like $(\sigma_{l,l}\phi_{l,l}(\epsilon))^2$ and $(\sigma_{l,l}\phi_{l,l}(\epsilon))(\zeta_l^{(l-1)}\psi_l^{(l-1)}(\epsilon))$ can be ignored. Using the lemma in [3], we obtain the following algorithm for estimating the errors in preprocessing.

for $l = 1, 2, \ldots, k$

begin

$\alpha_l = \max\{\zeta_l^{(l-1)}, \sigma_{l,l}\}$;

for $j = l+1, \ldots, q$

$$\zeta_j^{(l)} = \frac{|z_j^{(l-1)}\zeta_j^{(l-1)}| + |g_l s_{l,j}| \max\{\alpha_l, \sigma_{l,j}\}|}{|z_j^{(l)}|};$$

$$\eta^{(l)} = \frac{|u^{(l-1)}\eta^{(l-1)}| + |g_l b_l| \max\{\alpha_l, \mu_l\}|}{|u^{(l)}|}$$

end.

As explained in [3], the above estimation scheme can be incorporated with the preprocessing procedure and implemented on the same systolic architecture. Additional time is minimal because the calculations can be carried out during the otherwise idle time of the processors.

The error estimate for r_n can be obtained by the algorithm presented in [3] using $(\zeta_{k+1}^{(k)} \cdots \zeta_q^{(k)})$ and $\eta^{(k)}$ as the error estimates for $(z_{k+1}^{(k)} \cdots z_q^{(k)})$ and $u^{(k)}$, respectively. Again, we list the error estimation algorithm and refer the details to [3]. Define the relations between the exact and computed quantities as follows:

$$\hat{c}_{i,j}^{(n-1)} = c_{i,j}^{(n-1)}(1 + \xi_{i,j}\alpha_{i,j}(\epsilon)), \qquad \hat{c}_{i,j}^{(n)} = c_{i,j}^{(n)}(1 + \sigma_{i,j}\phi_{i,j}(\epsilon)),$$
$$\hat{z}_j^{(k)} = z_j^{(k)}(1 + \zeta_j^{(k)}\psi_j^{(k)}(\epsilon)), \qquad \cos\hat{\theta}_l = \cos\theta_l(1 + \sigma_{l,l}\phi_{l,l}^{(c)}(\epsilon)),$$
$$\hat{u}^{(k)} = u^{(k)}(1 + \eta^{(k)}\theta^{(k)}(\epsilon)), \qquad \sin\hat{\theta}_l = \sin\theta_l(1 + \sigma_{l,l}\phi_{l,l}^{(s)}(\epsilon)),$$
$$\hat{v}_i^{(n-1)} = v_i^{(n-1)}(1 + \xi_{i,p+1}\alpha_{i,p+1}(\epsilon)), \qquad \hat{v}_i^{(n)} = v_i^{(n)}(1 + \sigma_{i,p+1}\phi_{i,p+1}(\epsilon))$$

and

$$\hat{r}_n = r_n(1 + \eta\theta(\epsilon)).$$

The following algorithm estimates the error in the last element of the residual vector:

for $l = 1, 2, \ldots, p$

begin

$\sigma_{l,l} = \max\{\xi_{l,l}, \zeta_{k+l}^{(k+l-1)}\}$;

for $j = l+1, \ldots, p$

begin

$$\sigma_{l,j} = \frac{|c_{l,j}^{(n-1)} \cos\theta_l \max\{\xi_{l,j}, \sigma_{l,l}\}| + |x_{k+j}^{(k+l-1)} \sin\theta_l \max\{\zeta_{k+j}^{(k+l-1)}, \sigma_{l,l}\}|}{|c_{l,j}^{(n-1)} \cos\theta_l + x_{k+j}^{(k+l-1)} \sin\theta_l|};$$

$$\zeta_{k+j}^{(l+k)} = \frac{|c_{l,j}^{(n-1)} \sin\theta_l \max\{\xi_{l,j}, \sigma_{l,l}\}| + |x_{k+j}^{(k+l-1)} \cos\theta_l \max\{\zeta_{k+j}^{(k+l-1)}, \sigma_{l,l}\}|}{|c_{l,j}^{(n-1)} \sin\theta_l - x_{k+j}^{(k+l-1)} \cos\theta_l|}$$

end;

$$\sigma_{l,p+1} = \frac{|v_l^{(n-1)} \cos\theta_l \max\{\xi_{l,p+1}, \sigma_{l,l}\}| + |u^{(k+l-1)} \sin\theta_l \max\{\eta^{(k+l-1)}, \sigma_{l,l}\}|}{|v_l^{(n-1)} \cos\theta_l + u^{(k+l-1)} \sin\theta_l|};$$

$$\eta^{(k+l)} = \frac{|v_l^{(n-1)} \sin\theta_l \max\{\xi_{l,p+1}, \sigma_{l,l}\}| + |u^{(k+l-1)} \cos\theta_l \max\{\eta^{(k+l-1)}, \sigma_{l,l}\}|}{|v_l^{(n-1)} \sin\theta_l - u^{(k+l-1)} \cos\theta_l|}$$

end;

$$\eta = \eta^{(k+p)} \max\{\sigma_{1,1}, \cdots, \sigma_{p,p}\}.$$

5. AN EXAMPLE

The example in this section shows that the computed residual element may be accurate even when the matrix C is ill-conditioned. In this case, the proposed scheme gives a better error estimate than (3.8). Both the MS algorithm and the error estimation algorithm were implemented using MATLAB and run on a VAX 8300 with machine precision $\epsilon = 1.1102 \times 10^{-16}$ in the Communications Research Laboratory at McMaster University.

Example. Suppose the exact constraint matrix and corresponding right side vector are

$$S = \begin{pmatrix} 10^3 & 0 & 0 & 1 & 0 & 0 \\ 0 & 10^{-3} & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad b = \begin{pmatrix} -120000\sqrt{2}/7 \\ \sqrt{10}/700 \\ 6\sqrt{5}/7 \end{pmatrix}.$$

Thus we set the error estimates as $\sigma_{i,j} = \mu_l = 1$ and $\phi_{i,j}(\epsilon) = \delta_l(\epsilon) = \epsilon$. The data matrix at time $n-1$ is

$$X(n-1) = \begin{pmatrix} -1 & -\sqrt{5} & -2\sqrt{10} & 0 & 0 & 0 \\ 0 & -1 & \sqrt{2} & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 0 \end{pmatrix}.$$

Suppose we know the exact $R(n-1)$ and $u(n-1)$:

$$R(n-1) = \begin{pmatrix} 0.001 & 1000\sqrt{5} & 2\sqrt{10} \\ 0 & 1000 & -2\sqrt{2} \\ 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad u(n-1) = \begin{pmatrix} -10\sqrt{2}/7 \\ 4\sqrt{10}/7 \\ 6\sqrt{5}/7 \end{pmatrix}.$$

Similarly, the error estimates of their elements are all initialized as ϵ. Now the new data

$$x(n)^T = (-1 \quad -\sqrt{5} \quad -2\sqrt{10} \quad 0.001 \quad 0 \quad 0) \quad \text{and} \quad u^{(0)} = 0$$

are available and their error estimates are initialized as $\zeta_j^{(0)} = 1$, for $j = 1, \ldots, 6$, and $\eta^{(0)} = 1$, respectively. After preprocessing, we get $c(n)^T = (0.002 \quad 1000\sqrt{5} \quad 2\sqrt{10})$ and $v_n = -10\sqrt{2}/7$. The corresponding error estimates are $\zeta_j^{(3)} = 1$, for $j = 4, 5, 6$, and $\eta^{(3)} = 23$. The QR updating scheme and its error estimation algorithm are then applied to $R(n-1)$, $u(n-1)$, $c(n)$, v_n and their error estimates. The exact residual element $r_n = 6\sqrt{2}/35$. The computed error is

$$|\hat{r}_n - r_n| = 1.11 \times 10^{-16}.$$

The condition number of $C(n)$ is 4.6×10^6 and the error bound as given by (3.8) equals 3.40×10^{-9}. The estimation algorithm gives a much more accurate value of 9.62×10^{-16}.

ACKNOWLEDGEMENTS

The work of F. T. Luk was supported in part by the Rensselaer Polytechnic Institute, and by the Army Research Office under grant DAAL03-90-G-0104 and the Joint Services Electronics Program under contract F49620-90-C-0039 at Cornell University. The work of S. Qiao was supported in part by Natural Sciences and Engineering Research Council of Canada under grant OGP0046301.

REFERENCES

[1] L. Eldén, "Perturbation theory for the least squares problem with linear equality constraints," *SIAM J. Numer. Anal.*, Vol. 17, No. 3, June 1980, pp. 338–350.

[2] G. H. Golub and C. F. Van Loan, *Matrix Computations*, Second Edition, The Johns Hopkins University Press, Baltimore, Maryland, 1989.

[3] F. T. Luk and S. Qiao, "Analysis of a recursive least-squares signal-processing algorithm," *SIAM J. Sci. Stat. Comput.* Vol. 10, No. 3, May 1989, pp. 407–418.

[4] J. G. McWhirter, "Recursive least-squares minimization using a systolic array," *Real Time Signal Processing VI*, K. Bromley, Ed., Proc. SPIE 431, 1983, pp. 105–112.

[5] J. G. McWhirter and T. J. Shepherd, "A systolic array for linearly constrained least-squares problems," *Advanced Algorithms and Architectures for Signal Processing*, J. M. Speiser, Ed., Proc. SPIE 696, 1987, pp. 80–87.

Department of Computer Science

Technical Report

COMPUTING THE SINGULAR VALUE DECOMPOSITION ON A FAT-TREE ARCHITECTURE

TONG J. LEE
School of Electrical Engineering
Cornell University
Ithaca, New York 14853, USA

FRANKLIN T. LUK
Department of Computer Science
Rensselaer Polytechnic Institute
Troy, New York 12180, USA

DANIEL L. BOLEY
Department of Computer Science
University of Minnesota
Minneapolis, Minnesota 55455, USA

Rensselaer Polytechnic Institute
Troy, New York 12180-3590

Report No. 92-33 **November 1992**

COMPUTING THE SINGULAR VALUE DECOMPOSITION ON A FAT-TREE ARCHITECTURE

TONG J. LEE
School of Electrical Engineering
Cornell University
Ithaca, New York 14853, USA
tjlee@ee.cornell.edu

FRANKLIN T. LUK
Department of Computer Science
Rensselaer Polytechnic Institute
Troy, New York 12180, USA
luk@cs.rpi.edu

DANIEL L. BOLEY
Department of Computer Science
University of Minnesota
Minneapolis, Minnesota 55455, USA
boley@cs.umn.edu

ABSTRACT. The Singular Value Decomposition (SVD) is a matrix tool that plays a critical role in many applications; for example, in signal processing, it is often necessary to calculate the SVD in real time. We present here a new technique for computing the SVD on a parallel architecture whose processors are connected via a fat-tree. We tested our idea on the Connection Machine CM-5, and achieved efficiency up to 40% even for moderately sized matrices.

KEYWORDS. Singular value decomposition, parallel Jacobi algorithm, fat-tree, CM-5.

1. Introduction

Let A be a real $m \times n$ matrix. Its singular value decomposition (SVD) is given by

$$A = U\Sigma V^T,$$

where U and V are respectively $m \times m$ and $n \times n$ orthogonal matrices and Σ is an $m \times n$ diagonal matrix. The best approach to parallel SVD computation is apparently one of the Jacobi type; see, e.g., [1], [2], [4], [5], [7], [11], [12]. In this paper, we will discuss the efficient implementation of a Jacobi method on a parallel computer with a fat-tree interconnection network. We will propose a new Jacobi ordering for a fat-tree and analyze its behavior both theoretically and experimentally (on a Connection Machine CM-5).

This paper is organized as follows. In the next two subsections, we present the fat-tree architecture and Jacobi algorithm. Section 2 introduces a new fat-tree ordering, and provides some kernel programs. We analyze communication costs on a fat-tree network in Section 3, and discuss implementation results on the CM-5 in Section 4.

1.1. Fat-Tree Architecture

The fat-tree was introduced by Leiserson [10] as a novel approach to interconnect the processors of a general-purpose parallel supercomputer. This communication structure can also be seen in the distributed computing environment, such as a network of workstations.

The routing network of the Connection Machine CM-5 [14] is based on the fat-tree. This parallel machine consists of up to 544 (= 512 + 32) nodes for the model at the Army High Performance Computing Research Center (AHPCRC) at the University of Minnesota, and 32 nodes at the Northeast Parallel Architectures Center (NPAC) at Syracuse University. Each node of the CM-5 is a SPARC chip which runs at 32 MHz and delivers 22 Mips and 5 Mflops. There is a 64 Kbyte instruction and data cache and a 16 Mbyte memory in each node. All the nodes are synchronized. In October of 1992, two vector units will be installed in each processing node; each vector unit is capable of 64 Mflops peak and 40 Mflops sustained [9]. The control and data networks are connected via a *skinny* fat-tree structure. By *skinny*, we mean that the bandwidth does not increase proportionately to the number of nodes; in particular, the bandwidth is 20Mbyte/sec per node in a group of four processors, 10 Mbyte/sec per node in a group of sixteen, and 5Mbyte/sec overall. So data contention may severely degrade performance when all nodes need to access a large set of data from other nodes through the top level of the tree.

1.2. Jacobi Algorithm

The one-sided Jacobi method [8] generates an orthogonal matrix V such that the columns of the matrix W, given by $W = AV$, are mutually orthogonal. The matrix V can be generated by a sequence of plane rotations $V^{(1)}, V^{(2)}, \ldots$, where each $V^{(k)}$ is an identity matrix except for four entries: $v_{ii}^{(k)} = \cos\theta$, $v_{ij}^{(k)} = -\sin\theta$, $v_{ji}^{(k)} = \sin\theta$ and $v_{jj}^{(k)} = \cos\theta$, where (i, j) represents the index pair of the columns of A that $V^{(k)}$ orthogonalizes. The SVD computation requires $O(mn^2)$ operations for an $m \times n$ matrix A. For a limited

number of processors, i.e., up to $n/2$ processors, an efficient way is to configure them as a linear array along the horizontal dimension. Columns can be distributed either in blocks or in a wraparound fashion. Note from the above derivation that each column-pair can be orthogonalized independently, so that we may transform up to p pairs concurrently, where p denotes the number of processors. This method was used for computing the SVD on special machines, e.g., parallel computers such as the Illiac IV [11] and vector processors such as the CYBER 205 [3]. The one-sided Jacobi method is composed of these major steps:

1. Compute the norm of each column.
2. Compute plane rotations to orthogonalize paired columns.
3. Apply the plane rotations to update the columns and the column norms.
4. Permute the columns in a pre-chosen order to generate the next column pairs, and repeat the process from step 2.

If the column pairs are distributed to different processors, then step 4 requires communication. In the case of a two-dimensional mesh (as in the ILLIAC IV), each column is itself distributed among different processors and step 3 requires that the rotation parameters be transmitted to all the processors containing each given column pair. In the case of a one-dimensional array, each column pair is stored entirely in one processor and significant speedup is possible if vector units are present within each processor.

In this paper, we use the one-dimensional array, with each processor storing two blocks of columns. That is, we use a *block* Jacobi algorithm, in which the column blocks are circulated according to a given ordering to be defined, and the *cyclic-by-rows* ordering [6] is used within each block.

2. New SVD Algorithm

In the past, when the hypercube interconnection topology was in vogue, several Jacobi ordering schemes were proposed [1], [4], [7] to utilize the hypercube structure. Here, for a one-dimensional array of processors with no wraparound, a chess-tournament ordering [2] may be chosen because it does not waste processing power or memory space. However, communication requires a two-way transmission of columns between adjacent processors. An alternative is a ring ordering [4] which uses only one-way transmission, but it requires a wraparound connection. To develop an ideal ordering for a fat-tree, we aim to minimize the total path length by using the extra bandwidth of a fat-tree.

2.1. Fat-Tree Ordering

It is easiest to describe this ordering by an example. In Figure 1 we show the case for sixteen columns and eight processors. For pedagogic reasons, we use a base 8 numbering of the indices and so A=8, B=9, ..., H=15. The XOR (exclusive-or) column is the binary XOR of the column indices: at each step, the XOR value of each index pair is the same.

and from one step to the next this quantity follows the Gray code. The *cost-to-this-step* column denotes the maximum number of levels up the tree the messages must travel to reach their destinations from the previous step. In general, if there are p processors and two columns per processor, then a sweep requires $2p - 2$ steps. We save one step per sweep because the last step of sweep i can be included as the first step for sweep $i + 1$.

Step	Ordering of Index Pairs	XOR	Cost to This Step
0.	(01)(23)(45)(67)(AB)(CD)(EF)(GH)	0001	NA
	Forward Sweep		
1.	(03)(12)(47)(56)(AD)(BC)(EH)(FG)	0011	1
2.	(02)(13)(46)(57)(AC)(BD)(EG)(FH)	0010	1
3.	(06)(17)(24)(35)(AG)(BH)(CE)(DF)	0110	2
4.	(07)(16)(25)(34)(AH)(BG)(CF)(DE)	0111	1
5.	(05)(14)(27)(36)(AF)(BE)(CH)(DG)	0101	2
6.	(04)(15)(26)(37)(AE)(BF)(CG)(DH)	0100	1
7.	(0E)(1F)(2G)(3H)(4A)(5B)(6C)(7D)	1100	3
8.	(0F)(1E)(2H)(3G)(4B)(5A)(6D)(7C)	1101	1
9.	(0H)(1G)(2F)(3E)(4D)(5C)(6B)(7A)	1111	2
10.	(0G)(1H)(2E)(3F)(4C)(5D)(6A)(7B)	1110	1
11.	(0C)(1D)(2A)(3B)(4G)(5H)(6E)(7F)	1010	3
12.	(0D)(1C)(2B)(3A)(4H)(5G)(6F)(7E)	1011	1
13.	(0B)(1A)(2D)(3C)(4F)(5E)(6H)(7G)	1001	2
14.	(0A)(1B)(2C)(3D)(4E)(5F)(6G)(7H)	1000	1
	Backward Sweep		
13.	(0B)(1A)(2D)(3C)(4F)(5E)(6H)(7G)	1001	1
12.	(0D)(1C)(2B)(3A)(4H)(5G)(6F)(7E)	1011	2
11.	(0C)(1D)(2A)(3B)(4G)(5H)(6E)(7F)	1010	1
10.	(0G)(1H)(2E)(3F)(4C)(5D)(6A)(7B)	1110	3
9.	(0H)(1G)(2F)(3E)(4D)(5C)(6B)(7A)	1111	1
8.	(0F)(1E)(2H)(3G)(4B)(5A)(6D)(7C)	1101	2
7.	(0E)(1F)(2G)(3H)(4A)(5B)(6C)(7D)	1100	1
6.	(04)(15)(26)(37)(AE)(BF)(CG)(DH)	0100	3
5.	(05)(14)(27)(36)(AF)(BE)(CH)(DG)	0101	1
4.	(07)(16)(25)(34)(AH)(BG)(CF)(DE)	0111	2
3.	(06)(17)(24)(35)(AG)(BH)(CE)(DF)	0110	1
2.	(02)(13)(46)(57)(AC)(BD)(EG)(FH)	0010	2
1.	(03)(12)(47)(56)(AD)(BC)(EH)(FG)	0011	1
0.	(01)(23)(45)(67)(AB)(CD)(EF)(GH)	0001	1
	Forward Sweep		
1.	(03)(12)(47)(56)(AD)(BC)(EH)(FG)	0011	1

Figure 1. Fat-tree Ordering based on the Gray code (eight processors and sixteen columns).

2.2. Kernel Programs

To see how to write a simple node program to generate the fat-tree ordering, we use the following observations from the example in Figure 1. To simplify the presentation, we consider only the forward sweep. At each step, each processor must communicate with a remote processor whose label differs in one bit. The basis for our kernel presented here is to compute a mask such that the *exclusive-or* of the mask with the current processor label yields the remote processor label. When using the Gray code, this mask can be computed using only the step number - it is independent of the processor label.

We also use the following observations. First, we use the fact that the XOR's follow the Gray code. Second, we observe that during the second half of the forward sweep (steps 7-14), the lower half of the columns (numbers $0,\ldots,7$ in Figure 1) remain fixed in the processor with the same number. Hence the location of the remaining columns is fixed entirely by the Gray code. Third, we observe that the first half of the steps (steps 0-6) amount to doing a Gray code fat-tree ordering on each half of the processor array separately. The only remaining step is the transition from the first half to the second half (step 6 to step 7). Hence we can define the ordering for these steps recursively from the smaller cases.

We can summarize the steps for the forward sweep in the following procedure, in a pseudo-MATLAB notation assuming for the sake of simplicity of the presentation that the sends do not block.

```
% Node program for processor ProcNo for one forward sweep using an array of
% NProcs processors.  Assume Column(1) and Column(2) are the head and tail
% columns, respectively, in the local memory.

Orthogonalize_Individual_Column_Blocks % (within each block);

for StepNo = 1:2*NProcs-2,

   Pairwise_Orthogonalize_Column_Blocks;

   %% for each processor, figure where the data goes to and send it.

   [Mask,ColumnSwitch] = MakeMask(StepNo,ProcNo,NProcs);
   RemoteProcNo = XOR(ProcNo,Mask);

   Send Column(2) to remote processor RemoteProcNo;
   if ColumnSwitch == rotate,
      Column(2) = Column(1);
      Column(1) = receive_from(RemoteProcNo);
   else
      Column(2) = receive_from(RemoteProcNo);
   end;

end;
```

```
function [Mask,ColumnSwitch]=MakeMask(StepNo,ProcNo,NoProcs);

% ColumnSwitch indicates which column of the pair is to be sent/received.

% Mask is the XOR Mask so that RemoteProcNo = XOR(ProcNo,Mask).
% The Mask is computed independent of the processor label ProcNo.

% Handle first 2 steps as special cases to start recursion
if StepNo <= 2,
   Mask=1;
   ColumnSwitch = tail;
   if rem(ProcNo,2) == 1 & StepNo == 1, ColumnSwitch = rotate; end;

% First half of sweep: pretend this is a separate fat tree sweep on each
% half of the processor array.
else if StepNo < NoProcs-1,
   [Mask,ColumnSwitch] = MakeMask(StepNo,rem(ProcNo,NoProcs/2),NoProcs/2);

% Middle of sweep: here is first exchange through top of tree.
else if StepNo == NoProcs-1,
   Mask = NoProcs/2;
   ColumnSwitch = tail;
   if ProcNo >= NoProcs/2, ColumnSwitch = rotate; end;

% Last half of sweep: only tail columns move, figure Mask using Gray codes.
else if StepNo > NoProcs-1,
   Mask = xor(gray(StepNo),gray(StepNo+1));
   ColumnSwitch = tail;

end;
```

2.3. Test of Convergence

For a fat-tree ordering, any consecutive $2p-2$ (or even $2p-1$) steps may not constitute one sweep. We must complete a sweep, either forward or backward, to ensure that all column pairs have been orthogonalized. The convergence test is simple. We maintain a one-bit counter in every processor. The counter is reset at the beginning of every sweep, and is set whenever a column pair needs to be orthogonalized. At the end of the sweep, a global *or* operation is performed and convergence is achieved if no bit has been set.

3. Analysis on a Binary Fat-Tree Network

We consider a binary fat-tree with p processors, and assume that the communication time from one processor to another is determined by the number of links a message has to traverse and the capacity of these links. Our assumption is supported by experimental results reported in [13]. Define a channel to be the communication link between any two adjacent nodes; here a node can be a processor or an internal switching element. The capacity of a channel equals the number of parallel wires in the channel, and thus the maximum number of simultaneous bit-serial messages it can support [10]. Denote the capacity of the channels at the bottom level by γ. Label the levels from bottom up as level 1, 2, ..., so that the capacity of the channels at level l is given by $2^{l-1}\gamma$. Let us ignore start-up and latency costs. Within a single problem, all the messages have the same size and thus we measure the cost of multiple message transmission using *path length*.

For the ring ordering, at each step a message always goes through the top level and the maximum path length equals $2\log p$ (unless otherwise stated, we use base 2 logarithms). Since there is at most one message at each channel, congestion never occurs and it takes $2p-1$ steps to complete one sweep. The total path length equals $(4p-2)\log p$.

The fat-tree ordering does not cause congestion on a fat-tree network. Hence it suffices to count the number of times that each level is used. Denote that count by $c(p,l)$. Consider the forward sweep. We see from Figure 1 that with $p=8$ processors, the top level is used in two transition steps, the middle level in six steps and the bottom in fourteen steps. The first six steps correspond to the fat-tree ordering for the first four processors, and also for the second four processors. In the general case of p processors, there are $2p-2$ steps using $\log p$ levels, of which the first $p-2$ steps amount to the ordering for $p/2$ processors. When the number of processors doubles to $2p$, we add a new top level and the first $2p-2$ steps correspond exactly to the p processor ordering. There are an extra $2p$ steps, of which two use the new top level, four use the next level (the old top level), eight use the following level, etc. Formally, we get the recurrence

$$c(2p,l) = c(p,l) + 4(p/2^l) \qquad \text{for } l = 1,\ldots,\log p.$$

starting with $c(p,\log p) = 2$ and $c(p,l) = 0$ for $l > \log p$. Therefore, $c(p,l) = 4p/2^l - 2$, and the total path length is given by

$$2\sum_{l=1}^{\log p} c(p,l) = 2[(2p-2) + (p-2) + \ldots + 14 + 6 + 2] = 8p - 4\log p - 8.$$

For a large p, the path length ratio of the two orderings grows like $\log p/2$, a very attractive result for our new ordering.

4. Connection Machine CM-5

Although the CM-5 network is a 4-way tree, the analysis on 2-way trees is applicable. We take a 4-way tree and expand every interior node into a binary tree consisting of that node with two new children each connected to two of the four former children. The number of levels as well as the path length are doubled. However, the CM-5 is *skinny* and the capacity only doubles at every level. Hence it becomes a *skinny* 2-way tree in which the capacity goes up by $\sqrt{2}$ at each level.

To simplify our analysis, we concentrate on the 32-processor model. So $p = 32$ and there are three tree levels because $\lceil \log_4 p \rceil = 3$. The dominating communication cost for the CM-5 is the overhead time that is spent on address calculation, buffer space management, and so on. Let t_{or} and t_{of} represent the cost of such overhead in each step for the ring and the fat-tree ordering, respectively. Let t_{cf} be the overhead cost for resolving contention in the channels of the CM-5 network when applying the fat-tree ordering, and let t_e be the time for traversing an edge in the network. We note that $t_e < t_{cf} < t_{oh}$, where $t_{oh} \in \{t_{or}, t_{of}\}$, $t_{cf} \approx t_{oh}$, and $t_e \in (t_{oh}/10^3, t_{oh}/10^2)$. The overheads t_{or} and t_{of} depend on the data size and are of equal magnitude.

We proceed to compute the coefficient for t_e, which we assume to equal the number of messages that traverse the channels in one sweep. For the ring ordering, there is no congestion in the networks. So the coefficient for t_e is $2 \cdot 63 \cdot 3$ (=378), and the total cost equals $63\ t_{or} + 378\ t_e$. For the fat-tree ordering, we observe that level 1 is visited 62 times, level 2 fourteen times, and level 3 two times. We model the resolution of the contention by sending messages in batches. Messages through level 2 must be sent in two batches and messages through level 3 in four batches, in order to avoid contention. Hence we account for the thinness of the CM-5 network by assigning a weight of two to level 2 and a weight of four to level 3. The total path length is $2(62 + 2 \cdot 14 + 4 \cdot 2) = 196$ and the total cost equals $62\ t_{of} + 196\ t_e + t_{cf}$. Thus, on the CM-5 the fat-tree ordering may not outperform the ring ordering because of the extra cost associated with message contention.

4.1. Experimental Results

In Table 1 we present implementation results on a 32-node CM-5 for random $n \times n$ matrices with n ranging from 64 to 1024. The program was written in Fortran and each experiment repeated ten times. We measured the overall and computation (by disabling communication) costs for one sweep, and estimated the communication cost by subtracting the latter from the former. Our results show that, despite the message congestion that it causes on the CM-5, the fat-tree ordering gets more competitive as n grows, justifying our effort to minimize the total message path length (see also [13]). The mflops (million floating-point operations per second) figures in Table 2 are computed based on the count that $8n^3$ flops are required for one sweep. We conjecture that the *compute* performance deteriorates when n gets beyond 512 because the cache is no longer large enough to hold the huge column blocks. Nonetheless, our implementation results shows how, as the message size increases (hence t_e increases [13]), the fat-tree ordering quickly becomes competitive.

	n	64	128	256	512	1024
Overall	Ring	$7.595\,e^{-2}$	$3.229\,e^{-1}$	2.628	$1.794\,e^{1}$	$1.380\,e^{2}$
	Fat-tree	$8.134\,e^{-2}$	$3.481\,e^{-1}$	2.237	$1.795\,e^{1}$	$1.361\,e^{2}$
Compute	Ring	$3.013\,e^{-2}$	$2.320\,e^{-1}$	1.871	$1.493\,e^{1}$	$1.309\,e^{2}$
	Fat-tree	$3.436\,e^{-2}$	$2.420\,e^{-1}$	1.878	$1.493\,e^{1}$	$1.310\,e^{2}$
Communicate	Ring	$4.582\,e^{-2}$	$0.909\,e^{-1}$	0.757	3.010	7.110
	Fat-tree	$4.698\,e^{-2}$	$1.061\,e^{-1}$	0.359	3.020	5.140

Table 1. CPU Time (seconds) of Ring and Fat-Tree Orderings

	n	64	128	256	512	1024
Overall	Ring	27.61	51.96	51.07	59.85	62.25
	Fat-tree	25.78	48.20	60.00	59.82	63.11
Compute	Ring	69.60	72.32	71.74	71.92	65.62
	Fat-tree	61.03	69.33	71.47	71.92	65.57

Table 2. Mflops Rates of Ring and Fat-Tree Orderings

Acknowledgements

The work of T. J. Lee and F. T. Luk was supported in part by the Joint Services Electronics Program under contract F49620-90-C-0039 at Cornell University; F. T. Luk was also supported by start-up funds at the Rensselaer Polytechnic Institute. The authors thank the AHPCRC and NPAC for time on the CM-5, and Richard Brent and Lennart Johnsson for valuable discussions on CM-5 communication and hardware issues.

References

[1] C. H. Bischof, *The two-sided block Jacobi method on a hypercube*, in Hypercube Multiprocessors, M. T. Heath, ed., SIAM, 1988, pp. 612–618.

[2] R. P. Brent and F. T. Luk, *The solution of singular-value and symmetric eigenvalue problems on multiprocessor arrays*, SIAM J. Sci. Statist. Comput., 6 (1985), pp. 69–84.

[3] P. P. M. de Rijk, *A one-sided Jacobi algorithm for computing the singular value decomposition on a vector computer*, SIAM J. Sci. Statist. Comput., 10 (1989), pp. 359–371.

[4] P. J. Eberlein and H. Park, *Efficient implementation of Jacobi algorithms and Jacobi sets on distributed memory architectures*, J. Par. Distrib. Comput., 8 (1990), pp. 358–366.

[5] L. M. Ewerbring and F. T. Luk, *Computing the singular value decomposition on the Connection Machine*, IEEE Trans. Computers, 39 (1990), pp. 152–155.

[6] G. E. Forsythe and P. Henrici, *The cyclic Jacobi method for computing the principal values of a complex matrix*, Trans. Amer. Math. Soc., 94 (1960), pp. 1–23.

[7] G. R. Gao and S. J. Thomas, *An optimal parallel Jacobi-like solution method for the singular value decomposition*, in Internat. Conf. Parallel Proc., 1988, pp. 47–53.

[8] M. R. Hestenes, *Inversion of matrices by biorthogonalization and related results*, J. Soc. Indust. Appl. Math., 6 (1958), pp. 51–90.

[9] S. L. Johnsson. Private communication, September 1992.

[10] C. E. Leiserson, *Fat-trees: Universal networks for hardware-efficient supercomputing*, IEEE Trans. Computers, c-34 (1985), pp. 892–901.

[11] F. T. Luk, *Computing the singular-value decomposition on the ILLIAC IV*, ACM Trans. Math. Softw., 6 (1980), pp. 524–539.

[12] ——, *A triangular processor array for computing singular values*, Lin. Alg. Applics., 77 (1986), pp. 259–273.

[13] R. Ponnusamy, A. Choudhary, and G. Fox, *Communication overhead on CM5: an experimental performance evaluation*, in Frontier 92, Fourth Symp. on the Frontiers of Massively Parallel Computation, IEEE, 1992, pp. 108–115.

[14] Thinking Machines Corporation, *The Connection Machine CM-5 Technical Summary*, October 1991.

TASK 7 INTERRUPT AND BRANCH HANDLING FOR REAL-TIME SIGNAL PROCESSING SYSTEMS

H. C. Torng

Interrupt Handling For Out-of-Order Execution Processors*

H. C. Torng, Cornell University

Martin Day, Bell Information Systems

School of Electrical Engineering
Phillips Hall
Cornell University
Ithaca, NY 14853

Accepted for publication by IEEE Transactions on Computers

ABSTRACT

Processors with multiple functional units, including the superscalars, achieve significant performance enhancement through low-level execution concurrency. In such processors, multiple instructions are often issued and definitely executed concurrently and out-of-order. Consequently, interrupt and exception handling becomes a vexing problem.

We identify factors that must be considered in evaluating the effectiveness of interrupt and exception handling schemes: latency, cost, and performance degradation. We then briefly enumerate proposals and implementations for interrupt and exception handling on out-of-order execution processors.

Next, we present an efficient hardware mechanism, the Instruction Window (IW), and a new approach, which allows for precise, responsive and flexible interrupt and exception handling.

The implementation of the IW is then discussed. The design of an 8-cell IW has been carried out; it can work with a very short machine cycle time.

Finally, we present a comparison of all interrupt and exception handling schemes for out-of-order execution processors.

* The research reported herein has been supported in part by the Joint Services Electronics Program, Contract Number F49620-90-C-0039.

Interrupt Handling For Out-of-Order Execution Processors*

H. C. Torng, Cornell University

Martin Day, Bell Information Systems

School of Electrical Engineering
Phillips Hall
Cornell University
Ithaca, NY 14853

ABSTRACT

Processors with multiple functional units, including the superscalars, achieve significant performance enhancement through low-level execution concurrency. In such processors, multiple instructions are often issued and definitely executed concurrently and out-of-order. Consequently, interrupt and exception handling becomes a vexing problem.

We identify factors that must be considered in evaluating the effectiveness of interrupt and exception handling schemes: latency, cost, and performance degradation. We then briefly enumerate proposals and implementations for interrupt and exception handling on out-of-order execution processors.

Next, we present an efficient hardware mechanism, the Instruction Window (IW), and a new approach, which allows for precise, responsive and flexible interrupt and exception handling.

The implementation of the IW is then discussed. The design of an 8-cell IW has been carried out; it can work with a very short machine cycle time.

Finally, we present a comparison of all interrupt and exception handling schemes for out-of-order execution processors.

* The research reported herein has been supported in part by the Joint Services Electronics Program, Contract Number F49620-90-C-0039.

Interrupt Handling For Out-of-Order Execution Processors

1. Introduction

Processors with multiple functional units issue and execute multiple instructions concurrently and possibly out-of-order; they enhance performance by extracting low-level concurrency from the instruction stream [1, 2, 3]. The CDC 6600, IBM 360/91, and the CRAY machines are forerunners of this class of processors; however, these processors issue at most one instruction per cycle. Due to advances in device technologies, recently announced RISC processors often issue and certainly execute multiple instructions concurrently. However, these processors have not been able to support interrupt and exception[1] handling efficiently and with an acceptable latency.

In this paper, we address the interrupt handling problem, which has hampered the development of processors which execute and may even issue multiple instructions. We propose an efficient hardware mechanism, which supports an interrupt handling scheme with a flexible latency, set specifically for each type of interrupts requested.

The remaining sections are organized as follows: Section 2 presents a discussion of interrupts and exceptions. Factors for evaluating the effectiveness of interrupt handling schemes are presented. Existing proposals and implementations for interrupt handling on out-of-order execution processors are briefly reported in Section 3.

Section 4 presents the Instruction Window (IW), a simple and yet versatile hardware mechanism which supports efficient and flexible interrupt handling. Basic window operations are introduced in Section 5. Section 6 proposes an innovative interrupt handling scheme, which makes us of the IW. In Section 7, we discuss the implementation of the IW. Section 8 gives an evaluation of all interrupt handling schemes.

2. Interrupts and Exceptions

An important and indispensable feature of any processor is its ability to handle properly interrupts and exceptions. An I/O device, a sensor, or a timer may "interrupt" a processor to perform a specific task. An executing

[1] From now on, we will simply use interrupt to stand for interrupt and exception.

instruction may cause a page fault or an overflow/underflow; an "exception" thus results. Finally, one may place an instruction in an instruction stream to call for a "trap", which initiates a pre-planned action. Presentations on interrupts, exceptions and traps can be found from many sources, among them: [4,5,6,7,8]. In this paper, we use the term interrupt to denote an interrupt, an exception or a trap. Our study does not treat the subject of interrupt detection; rather, we investigate how a processor responds to an interrupt request, once it has been received.

When an interrupt request is received, the processor must save its processor state, then load and execute an appropriate interrupt handler. Upon completion of the interrupt handling routine, the saved processor state is restored, and the interrupted process can then be restarted.

A processor state should contain enough and preferably only enough information so that the interrupted process can be restarted at the **precise** point where it was interrupted. To be able to resume an interrupted process, the processor state should consist of the contents of the general purpose registers, the program counter, the condition register, all index registers and the relevant portion of the main memory.

The classical approach to identifying precisely the point where a process is interrupted is to save, among other vital items, the address of a specific instruction, say instruction α, when the processor state is saved. All instructions that precede instruction α have been executed. And instruction α and those that follow it have not. Instruction α thus provides a **precise** interrupt point.

For processors, which execute instructions concurrently and possibly out-of-order, the identification of a precise interrupt point when an interrupt request is made may become very costly.

In order to evaluate interrupt handling schemes, a framework must be established. Three factors have been identified:

1) Latency:

   An interrupt handling approach must be judged by the latency between the receipt of an interrupt request and the completion of saving the processor state. Clearly, any acceptable interrupt handling scheme should yield a latency, that is appropriate for the interrupt request, which may be generated internally or externally.

2) Component Cost:

   The cost of additional hardware and software incurred by the installation of an interrupt handling scheme must be considered.

3) Performance Degradation:

The presence and operation of an interrupt handling scheme may bring about performance degradation; its extent should be critically examined.

There are three sources of degradations:

i. Abort -- In response to an interupt request, some instructions that have already been partially or even completely executed are "aborted";

ii. Execution inhibition -- the need to maintain a "consistent" processor state prevents some instructions which have been executed out-of-order from depositing their results; this in turn inhibits the execution of subsequent instructions which use these results as operands;

iii. Update -- Certain schemes, such as checkpointing, require run-time continuous updating operations, which have to be performed by the processor.

3. Interrupt Handling Schemes

In the past, the trend in the design of processors with multiple functional units has been towards sequential instruction issue, concurrent execution and possibly out-of-order instruction completion.

The CDC 6600 [9] maintains a "SCOREBOARD" to resolve dependency conflicts among instructions in an instruction stream, and allows these instructions to complete out-of-order. The "exchange jump" is the primary interrupt mechanism for the Central Processing Unit (CPU) to handle interrupts. If the exchange jump sequence is requested, the CPU is permitted to issue instructions up to, but not including, the next instruction word. All issued instructions are allowed to run to completion. The CPU registers are then interchanged with the data stored in the exchange package. The CPU is restarted at the location specified by the new contents of the program counter. Since the processor must, on average, wait for two instructions to be issued and completed before the interrupt can be serviced, this approach exacts a penalty in latency.

The IBM 360/91 supports both precise and imprecise interrupt handling [10]. Upon the receipt of a precise interrupt request, or a trap (either precise or imprecise), the instruction decoding is temporarily halted and all issued instructions are allowed to complete, thereby resulting in considerable

latency. If an imprecise interrupt is generated (via internal processing), the state of the system is lost and therefore the interrupted process cannot be restarted precisely.

When an interrupt is received in the CRAY-1 [11, 12], instruction issue is temporarily terminated, and all vector and memory bank references are allowed to complete. The interrupt handler is loaded and executed in a similar manner to that employed by the CDC 6600. The CRAY-1 must, on average, wait for two instructions to complete before the processor state can be saved. However, as the CRAY-1 supports complex vector operations, the latency (in cycles) may be longer than the CDC 6600.

Hwu and Patt [13] proposed a promising approach to handling interrupts. A minimum of two checkpoints and hence two additional states are required. Essentially, the checkpoints, which invariably incur some penalty in processor performance, are used to divide the sequential instruction stream into smaller units to reduce the cost of "repair".

Smith and Pleszkun [14] presented several interesting methods to realize the classical precise interrupt. The simplest is the in-order instruction completion method: an instruction is only allowed to modify the processor state when all its preceding instructions are certain to be allowed to complete. A "reorder buffer" is added so that instructions are permitted to complete out of order; it is used to reorder them before they are permitted to modify the processor state. "History buffer" and "future files" are suggested as alternatives. Result bypass is proposed to reduce concomitant performance degradations, which they quantified with extensive simulations.

Sohi [15] deftly combined the operations of reservations stations and reorder buffers into the "register update unit" to effect precise interrupts. In addition, Smith and Pleszkun [14] presented several very stimulating "architectural solutions"; these include saving the "intermediate state of vector instructions" and saving "a sequence of instructions that must be executed before the saved program counter is precise".

4. The Instruction Window (IW)

In this section, we present a hardware mechanism, which contributes toward precise, responsive interrupt handling for processors with multiple functional units.

The general structure of a processor with multiple functional units is shown in Figure 1. It depicts a General Purpose Register (GPR), or equivalently Load/Store, architecture.

The instruction unit prepares the incoming instructions for execution, and issues the instructions to the appropriate functional units via the interconnection network. The functional units operate on the given operands and produce results which are returned to the appropriate registers.

We propose the installation of a hardware structure named the *Instruction Window* (IW). The IW, shown in Figure 2, consists of a set of registers, to be called cells. One and only one instruction occupies a cell. Such a cell serves as a "staging area" for an instruction. In a conventional processor, an instruction is removed from the staging register as soon as it has been issued to a functional unit. In the proposed mechanism, an instruction remains in its "staging register" after its issuance.

We use a three-operand format for instructions:

$$i: \quad OP \quad S1, \quad S2, \quad D \qquad (1)$$

where *i* denotes the instruction tag, OP the operation, S1, S2 the registers used as sources, and D the destination register.

Each cell contains at least three fields: issue, tag, and instruction. An optional vector element number (VEN) can be added for those processors with vector instructions.

The issue field has one bit, which is used to indicate whether that instruction has been issued to a functional unit. This field will not be shown in later figures.

The tag field contains a tag which uniquely identifies the instruction held in that cell.

The instruction field contains a copy of the instruction as it was fetched from the Instruction Buffer.

The optional vector element number (VEN) field is set to a value, equal to the number of vector elements left to be processed. We assume the availability of a Vector Length Register, which provides the initial value for VEN. If the instruction is a scalar instruction, VEN is set to 1.

Thus, as an example, the following 2-instruction sequence may appear in the IW as shown in Figure 3:

```
1:   ADD    R0,   R1,   R2
2:   ADF    VR0,  VR1,  VR2        (2)
```

Note that instructions 1 and 2 occupy two consecutive cells and the cell for instruction 1 is above that for instruction 2. The issue field is omitted. The

opcode ADD stands for an integer addition and ADF stands for a floating-point addition. Registers are denoted with R's and vector registers with VR's.

5. Window Operations

We present in the following the basic operations for the IW: fill, issue, and remove/update.

Fill

When a fill operation begins, the IW has already "pushed" its remaining instructions to the top. Let instruction i precede instruction j in an instruction stream; then i is always located "higher" in the IW then j. And the empty cells are found at the bottom.

When an instruction is written into the IW, it is always placed at the topmost empty cell with a unique tag. Concurrently, if the VEN field is implemented, it is set to N, the vector length specified by the Vector Length Register. The instructions freshly written into the IW follow the same order seen in the instruction stream.

Due to restrictions imposed by available data paths, the number of instructions that can be moved concurrently from the instruction buffer to the IW has to be limited.

At this point, a reader may justifiably have concerns about the implementation of the fill and other operations to be introduced in this section and their possible impact on the machine cycle time; this will be addressed in Section 7.

We use the sequence of instructions, given in (3), to illustrate the operations of the IW.

```
1:   MUL    R0,   R1,   R0
2:   ADD    R2,   R3,   R2
3:   ADF    VR0,  VR1,  VR0
4:   ADD    R4,   R5,   R4          (3)
5:   ADD    R6,   R7,   R6
6:   ADD    R8,   R9,   R8
7:   ADD    R10,  R11,  R10
```

The opcode MUL stands for an integer multiplication operation. The sequence in (3) is designed simply to present the salient features of the IW; it is not meant to stand for any meaningful computation.

For this example, let the processor issue at most one instruction per cycle. It is further assumed that instructions be written into the IW from the instruction buffer at a rate of one per cycle. Three functional units are available: an integer add unit, a floating-point add unit, and an integer multiply unit. Both the integer and the floating point add operations have a latency of three cycles. Pipelining has been employed so that an add operation may be started every cycle. An integer multiply takes six cycles and the multiply unit is not pipelined.

In cycle 1, instruction 1 is written into the topmost cell of the IW. In cycle 2, instruction 2 is written into the IW. As long as there is room in the IW, in cycle i (i= 3, 4, ...,6), instruction is written into the IW. Assuming an IW with at least 6 cells, the contents of the IW after cycle 6 is depicted in Figure 4.

Issue

For processors which issue at most one instruction per cycle, at the beginning of each cycle, the topmost instruction that has not been issued to a functional unit is examined. If an appropriate functional unit, and the requisite data paths are available, and *no data dependency exits*, the instruction is issued. For possible future extensions to processors which may issue up to k instructions per cycle, where k > 1, at least the topmost k instructions that have not been issued are examined.

For each issued instruction, its opcode, operands and result specifications are passed to the assigned functional unit, and actions are initiated to copy operands from the source registers to the functional unit.

Fill and issue operations, as specified, will be performed concurrently. Implementation issues are discussed in Section 7.

Returning to our example, instruction 1 is issued to the integer multiplier in cycle 2. During cycle 3, instruction 2 is issued to the scalar adder. Instruction 3 is issued to the floating-point adder in cycle 4. Note that each of these instructions does not have any data dependency. For each subsequent cycle, at most one instruction will be dispatched to a functional unit. The processing of these instructions is depicted in Figure 5, where L stands for fill, I for issue, E for execute, and S for deposit.

Let us examine instruction 2, which is issued in cycle 3, as shown in Figure 5. Starting in cycle 4, the pipelined integer add unit operates on its operands. Since it has an execution latency of 3 cycles, it will produce its result at cycle 7 and proceed to deposit it into the destination register, R2.

Remove/Update

At the instant when an interrupt request is received by the processor, an issued instruction may be at an intermediate stage of execution. We have a choice of aborting the execution of that instruction or completing its execution. Aborting an instruction means that the execution already performed is wasted; on the other hand, letting an instruction execution run to its completion may impose a long latency before responding to an interrupt request.

We present a new parameter: *No Return Point* (NRP): the execution point after which the instruction should not be prevented from changing the processor state.

For an interrupt which requires "fast" response, for example an internal "machine check", we can set the NRP to be at the start of the final machine cycle, when the computation result is written into the the destination register. With such an NRP setting, only those executing instructions which are about to deposit their results[2] are allowed to complete; all other instructions are aborted; some of the instruction processing already performed is traded away for short latency.

At the other extreme, the NRP can be set at the start of an instruction execution. In so doing all instructions which have started execution are allowed to complete. As appropriate, the NRP can be set somewhere between the two extreme cases.

The definition of the NRP provides us with a means of achieving flexible responses to various types of interrupts. Each interrupt type has its own NRP setting; a processor reacts differently in response to different types of interrupts.

For illustrative purposes, we set, in this paper, the NRP at the start of the final cycle, when the computation result is written into the destination register. For instruction 2, its NRP is cycle 7.

When an instruction reaches its NRP, it will be allowed to complete and therefore should be removed from the IW upon its completion. To accomplish this, the executing functional unit returns the instruction tag to the IW for identification.

It is quite reasonable to assume that it takes one cycle to transmit a tag from the functional unit to the IW. The functional unit returns the tag of an executing instruction to the IW one cycle before it reaches its NRP, so that it will be identified as soon as it reaches its NRP.

---

[2] Note that a result is always deposited into the register file, not the IW.

More than one instruction may reach its NRP at a given cycle. To accommodate the return of multiple tags, we propose a "1 out of w" code for the tags. As an example, let there be 8 cells in the IW, w=8. Each tag has 8 bits with one and only bit, that has a value of 1. In this case, a path with 8 bits will suffice to return all 8 tags, if necessary.

All cells whose tags match the returned tags are marked. Instructions residing in marked cells can then be updated or removed. An instruction in a marked cell is removed if its VEN value is 1. If the VEN value is greater than 1, then it is decreased by 1.

Instructions are removed so that all empty cells are found at the bottom of the IW.

Remove and update operations follow the tag matching, and take place concurrently.

Since the start of cycle 7 is the NRP of instruction 2, its tag is returned to the IW during cycle 6. In cycle 7, an associative search is performed on the IW using the incoming tag as a key. Since the VEN value for instruction 2 is 1, during the remove/update operations of cycle 7, instruction 2 is eliminated as shown in Figure 6. Note that the remaining instructions are pushed to fill the top of the IW, preserving their order in the instruction stream. Note also that register R2 is updated with the result produced by instruction 2.

To illustrate the IW operations further, let us examine the execution of instruction 3, depicted in Figure 5. The first element of the add instruction, issued in cycle 4, will pass its NRP at the start of cycle 8. Thus, its tag is returned to the IW during cycle 7, and is associatively matched with the tags in the IW during tag matching in cycle 8. Then, since its VEN value is not equal to 1, the vector element number corresponding to instruction 3 is decreased by 1 during the remove/update operations of this cycle, as shown in Figure 7. The presence of 2 in the VEN field indicates that two vector elements remain to be processed.

The timing of these three basic operations: fill, issue, remove/update, is depicted in Figure 8. The implementation details will be presented in Section 7.

Let an interrupt request be received in cycle 8. The processing of the received interrupt request will be described in the following section.

6. Interrupt Handling

Upon the receipt of an interrupt request, the processor responds as follows:

1) At the start of the following cycle, an Interrupt Request Signal is generated, and sent to the executing functional units. This aborts all instructions that have not passed their NRPs. Any instruction which has passed its NRP is allowed to complete its execution.

2) When all instructions that are permitted to proceed complete their execution, the processor state is saved. **The IW is included as a component of the processor state and the contents of the occupied cells of the IW are saved.**

3) The appropriate interrupt handler is then fetched from memory and executed by the processor.

In the proposed scheme, the IW is included as a component of the processor state. The saved contents of the IW provides a **modified** "precise" interrupt point. The IW does not identify one instruction which defines the precise interrupt point; rather, it identifies a group of instructions in the IW, which jointly define the "point", where the interrupted processing should resume.

As a component of the processor state, the saved contents of the IW cannot be modified by any interrupt handler.

If an instruction remains in the IW, its VEN field specifies the number of elements to be processed for the given instruction. This information is used to restart the processor at the completion of the interrupt handling procedure. The introduction of the VEN field obviates the need for the processor to re-execute an incomplete vector instruction from the very beginning when the processing resumes.

Returning to our example, the interrupt request was received during cycle 8. The processor then generates the Abort Signal during cycle 9.

At the start of cycle 9, the second element of the vector operand specified by instruction 3, and instructions 1 and 4, pass their respective NRPs. Thus the instructions bearing tags 1 and 4 are eliminated from the IW during cycle 9, and the VEN value for instruction 3 is decreased by 1. The resulting contents of the IW are shown in Figure 9; it is saved as part of the processor state.

Note that Registers R0, R4, and the second element of the vector register, VR0, are all updated by the execution of instructions 1, 4 and 3 respectively.

Furthermore, in cycle 9, all functional units are flushed thereby eliminating all instructions which have not passed their NRPs. The

maximum number of execution cycles that could be wasted on an instruction by flushing the functional units is the latency of the "slowest" functional unit. This is the greatest number of cycles that may have elapsed between the issuance and one cycle before the instruction passes its NRP.

The processor state is saved during cycle 10. Note that instructions 3, 5, 6 and 7 define jointly the "precise" interrupt point. Only the last element of the vector add remains to be processed when the execution of the interrupted process resumes. Note that instruction 4, which follows instruction 3, does not appear as it has already completed its execution before the interrupt request arrived.

The interrupt handler is then fetched and executed. As a specific example, suppose one of the instructions being executed causes an exception due to overflow; the interrupt handler can execute the program, starting with the instructions remaining in the IW - the modified "precise" interrupt point, one instruction at a time to identify the source of the problem.

As in any processor, after completion of the interrupt handler, the original processor state, of which the IW is a component, is restored. Instruction issuing is then restarted from the top of the IW.

Returning to the example, the IW will be restored as shown in Figure 9 at the completion of the interrupt handler. Let the processor state be restored at cycle X. Thus, instruction 3 is issued again with the VEN field being 1 in cycle X. In cycle X+1, instruction 5 is issued to the integer adder. In cycle X+2, instruction 6 is issued. Finally, instruction issuance completes in cycle X+3, when instruction 7 is issued to the adder.

7. Implementation

The Instruction Window (IW) plays an important part in the proposed interrupt handling scheme; it also serves as the staging registers for instruction decoding. The IW differs significantly from a "reorder buffer" [14] in that the computation results are not deposited into the IW and more importantly instructions can be removed from it "out-of-order". We now examine the implementation issues in more detail and assess its potential impact on machine cycle time.

We propose that, as depicted in Figure 8, in each machine cycle "fill", "issue" and "tag match" take place concurrently. And remove/update follows tag match.

At the beginning of each machine cycle, the remaining instructions in the IW have already been moved to occupy the top cells. And the incoming instructions are placed into the cells adjacent to the occupied ones.

The identification of instructions that can be issued deserves more scrutiny. The first task is to identify instructions, at the top of the IW, that have not yet been issued. Since every cell carries an "issue" bit, simple combinational circuit can be used to accomplish this.

For conventional processors, where at most one instruction is issued per machine cycle and instructions are issued according to their order in the instruction stream, only the top unissued instruction in the IW needs to be examined for possible issuance. This is exactly what is done anyway; no additional control complexity is introduced. For future processors which may issue multiple instructions, more complex circuits to detect and resolve dependencies have to be installed. For an 8-cell IW and a 16-element register set, we find that the "multiple issue" critical path incurs a 16-gate delay. The actual path length is of course determined by the packaging details.

We now discuss the tag match, followed by remove/update. With the "1 out of w" coding scheme, all returning tags from the executing functional units are "or-ed" into one. This one tag is matched with all the cells whose instructions have been issued. Matched cells are marked for removal or update in the second half of the cycle.

A marked cell is removed if its VEN field contains 1; otherwise, the value in the VEN field is decreased by 1. Since we require that all remaining instructions be pushed to the top portion of the IW, combinational circuits are provided to write into each cell every cycle. For an 8-cell IW, we find that the path length is 12-gate long. Keep in mind that the remove/update operation follows an associative memory search for tag matching; the total length matches very nicely with the "issue" critical path.

To summarize: the implementation of these operations for a moderately sized IW produces a critical path, which can work with a very short machine cycle time. We have not yet studied rigorously the size of the IW, which can be much larger that 8 for future processors

8. Evaluation

Now we evaluate the scheme with IW and other schemes reported in Section 3, using the criteria enumerated in Section 2. Table 1 provides a summary. The "Abort", "Execution inhibition" and "Updating" performance degradations are defined in Section 2.

Table 1: Evaluation of interrupt handling schemes

	Latency	Component Cost	Performance Degradation
CDC6600 [9]	On the average, two instructions are to be issued and executed.	Provisions for exchange jump.	None.
360/91 [10]	All issued instructions are allowed to complete.	None	None
CRAY-1 [11,12]	All vector and memory bank reference inst's are allowed to complete.	None	None
HPS [13]	Needs to return to the nearest consistent state.	registers, memory and data paths needed to implement checkpoints.	Abort and update degradations incurred.
In-order Inst. Completion [14]	Relatively short.	Needs a "Result-shift" register file.	Abort, execution inhibition and update degradations incurred.
Reorder(History, Future File) Buffer [14, 15]	Relatively fast.	Needs buffers and data paths.	Abort, execution inhibition and update degradations incurred.
Reorder Buffer with bypass [14,15]	Relatively fast.	Needs buffers and elaborate data paths.	Abort and update degradations incurred.
Instruction Window (IW)	Flexible with adaptive NRP settings.	Needs to implement IW.	No update and execution inhibition degradations. Abort penalty is a function of NRP.

9. References

[1] A. J. Bernstein, "Analysis of Programs for Parallel Processing", IEEE Trans. Electron. Computer., vol EC-15, No 5, pp 757-763, Oct. 1966.

[2] C. C. Foster, E. M. Riseman, "Percolation of Code to Enhance Parallel Dispatching and Execution", IEEE Trans. Computer., vol C-21, no 12, pp. 1411-1415, Dec. 1972.

[3] R. M. Keller, "Look-Ahead Processors", Computer. Surv., vol. 7, pp 63-72, Dec. 1975.

[4] J. L. Hennessy and D. A. Patterson, **Computer Architecture: A Quantitative Approach**, Morgan Kaufmann, 1990.

[5] H. S. Stone, **High-Performance Computer Architecture**, Addison-Wesley, 1990.

[6] J. P. Hayes, **Computer Architecture and Organization**, McGraw-Hill, 1988.

[7] K. Hwang and F. A. Briggs, **Computer Architecture and Parallel Processing**, McGraw-Hill, 1984.

[8] P. M. Kogge, **The Architecture of Pipelined Computers**, McGraw-Hill, 1981.

[9] J. E. Thornton, **Design of a Computer: The Control Data 6600**, Scott, Foresman and Company, 1970.

[10] D. W. Anderson, F. J. Sparacio, F.M. Tomasulo, "The IBM System/360 Model 91: Machine Philosophy and instruction-Handling," IBM J., vol 11, pp 8-24, Jan. 1967.

[11] CRAY-1 Computer Systems Hardware Reference Manual, Publication Number: HR-0808, Revision B, CRAY Research Inc., June 1982.

[12] R. M. Russell, "The CRAY-1 Computer System", Comm. ACM, vol 21, no 1. pp 63-72, Jan. 1978.

[13] W. W. Hwu, Y. N. Patt, "Checkpoint Repair for High Performance Out-of-Order Execution Machines", IEEE Trans. Computer., vol C-36, no. 12, pp 1496-1514, Dec. 1987.

[14] J. E. Smith and A. R. Pleszkun, "Implementing Precise Interrupts in Pipelined Processors", IEEE Trans. Computer., vol C-37, no. 5, pp. 562-573, May 1988.

[15] G. S. Sohi, "Instruction Issue Logic for High-Performance, Interruptible, Multiple Functional Unit, Pipelined Computers", IEEE Trans. Computers, vol 39, no. 3, pp 349-359, March 1990.

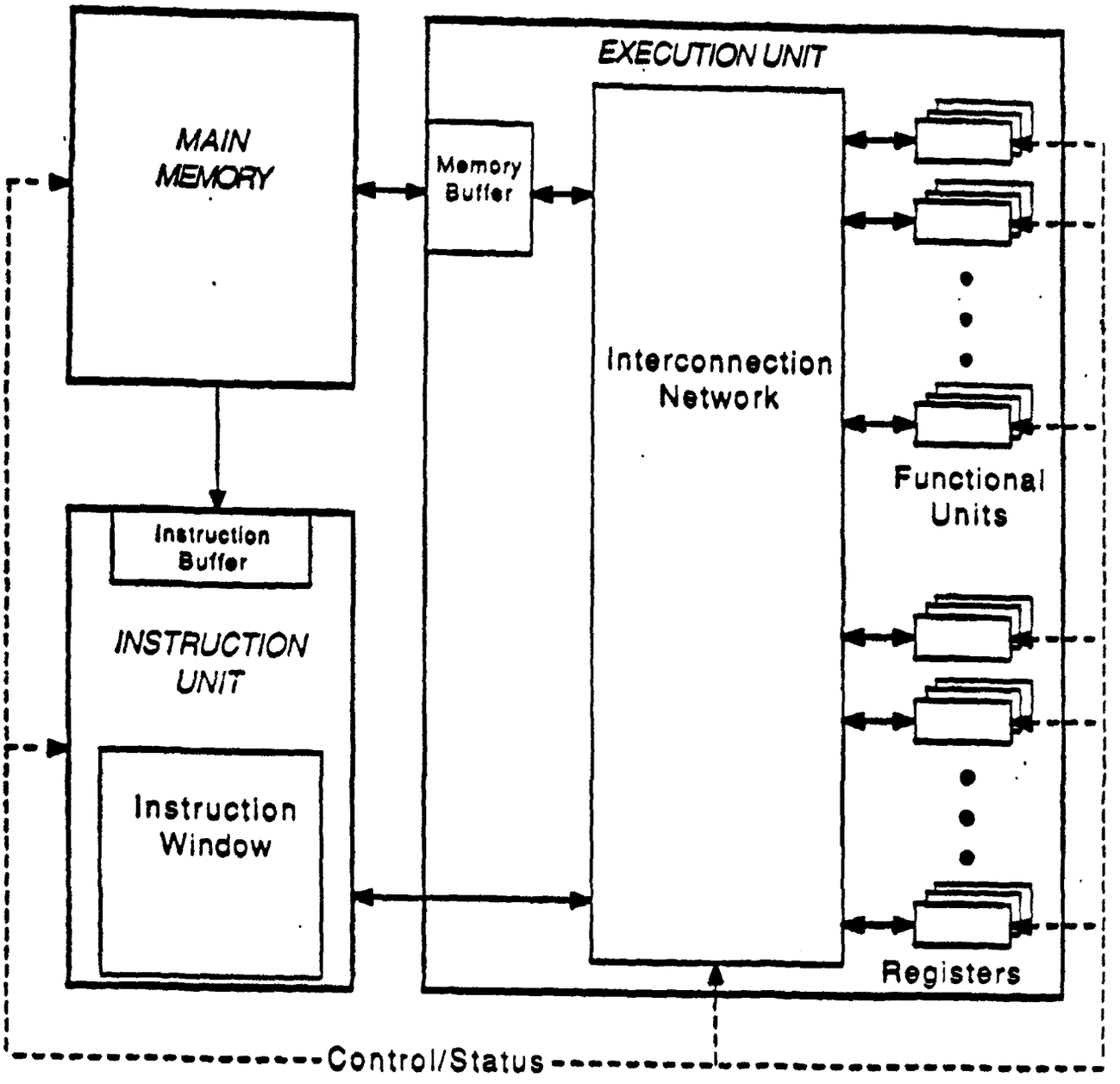


Figure 1: General Purpose Register, Load/Store, Processor Structure

Issue	Tag	Instruction	VEN

Figure 2: The Instruction Window (IW)

1	ADD R0, R1, R2	1
2	ADD VR0, VR1, VR2	2

Figure 3: Occupied Cells in the IW

1	MULT R0, R1, R0	1
2	ADD R2, R3, R2	1
3	ADD VR0, VR1, VR0	3
4	ADD R4, R5, R4	1
5	ADD R6, R7, R6	1
6	ADD R8, R9, R8	1

Figure 4: The IW after Cycle 6

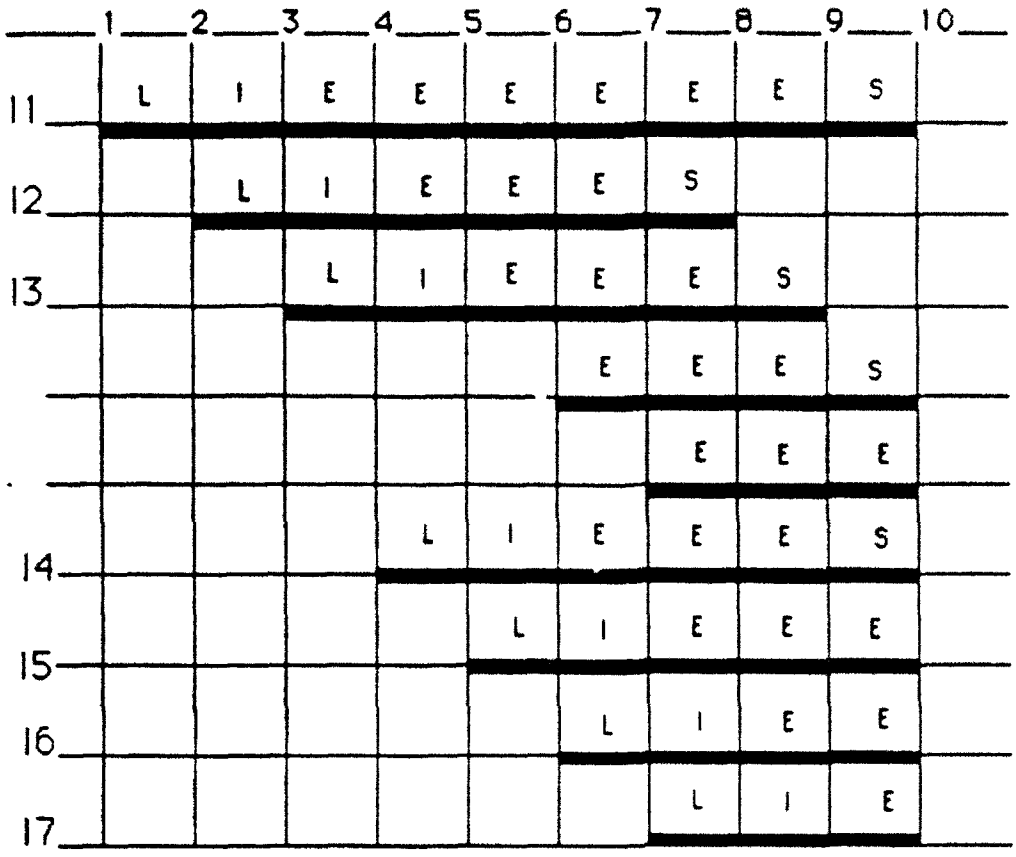


Figure 5. Depiction of Instruction Execution for the first 9 Cycles.

1	MULT R0, R1, R0	1
2	ADD R2, R3, R2	1
3	ADD VR0, VR1, VR0	3
4	ADD R4, R5, R4	1
5	ADD R6, R7, R6	1
6	ADD R8, R9, R8	1

Figure 6: The IW after Cycle 7

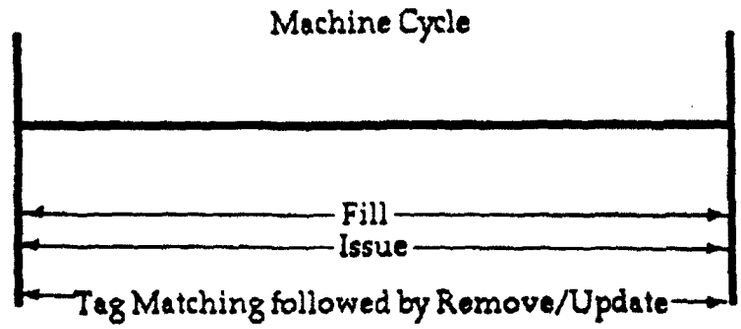


Figure 8: Timing of IW Operations

3	ADD VR0, VR1, VR0	1
5	ADD R6, R7, R6	1
6	ADD R8, R9, R8	1
7	ADD R10, R11, R10	1

Figure 9: The IW after Cycle 9

An Out-of-Order Superscalar Processor with Speculative Execution and Fast, Precise Interrupts

Harry Dwyer
IBM Corporation
9737 Great Hills Trail
Austin, Texas 78759

H.C. Torng
School of Electrical Engineering
Cornell University
Ithaca, N.Y. 14853

Abstract

The achievement of fast, precise interrupts and the implementation of multiple levels of branch predictions are two of the problems associated with the dynamic scheduling of instructions for superscalar processors. Their solution is especially difficult if short cycle time operation is desired. We present solutions to these problems through the development of the Fast Dispatch Stack (FDS) system.

We show that the FDS is capable of scheduling storage, branch, and register-to-register instructions for concurrent and out-of-order executions; the FDS implements fast and precise interrupts in a natural, efficient way; and it facilitates speculative execution -- Instructions preceding and following one or more predicted conditional branch instructions may issue. When necessary, their effects are undone in one machine cycle.

We evaluated the FDS system with extensive simulations.

1. Introduction

Superscalars exploit instruction level parallelism by issuing multiple instructions each cycle to functional units when dependencies allow. Instruction scheduling can be performed during compilation (static scheduling) or during execution (dynamic scheduling), or both. Dynamic scheduling detects instruction dependencies in a segment of the dynamic instruction stream. The most general form of dynamic scheduling, the issue and execution of multiple out-of-order instructions, can significantly enhance system performance [1][4][9]. However, there are problems with this scheme that undermine its usefulness.

The achievement of precise interrupts is difficult, particularly if a fast response time is desired. Interrupts are precise if processor state visible to the operating system and application can be reconstructed to the state a processor would have, had all instructions executed in sequence up to the point of an interrupt; this is costly to implement, particularly if out-of-order stores to memory may occur.

Branches, about 15% to 30% of executed instructions for many applications [6], decrease the effectiveness of multiple issues to functional units if instructions following an undecided branch cannot be issued. Performance may be improved by enabling speculative executions on a predicted path of instructions. If the gains on correct paths outbalance the losses from nullifying execution effects on incorrect paths (squashing), performance improves.

It is imperative that cycle time be considered when investigating new processor structures. Since a processor's performance depends on throughput (instructions issued per cycle) and cycle time, if throughput is increased at the expense of cycle time, a net performance improvement may not occur; performance may, in fact, decrease during the execution of inherently sequential code. Hence, we have developed FDS structures that operate on a short cycle time.

This paper is organized into 5 sections. A dynamic scheduling mechanism that may issue multiple, out-of-order instructions each machine cycle is presented in Section 2. A fast, precise

0-8186-3175-9/92 $3.00 © 1992 IEEE

interrupt handling capability is derived in Section 3 and an instruction squashing capability is presented in Section 4. The performance of the proposed structure is evaluated in Section 5 and tradeoffs are analyzed with simulation. In Section 6, we present conclusions.

2. The Fast Dispatch Stack System

We present an overview of the structure and operation of the Fast Dispatch Stack (FDS) system (Figure 1) in this section. A detailed and comprehensive presentation of the FDS system, a major enhancement of the Dispatch Stack [1], can be found in [2]. The FDS contains a Buffer Unit (BU) and an Issue Unit (IU). The BU supplies instructions to the IU in a form that facilitates fast dependency detection. The IU detects instruction register dependencies and issues instructions with no dependencies to the functional units (FUs) each cycle via an interconnection network.

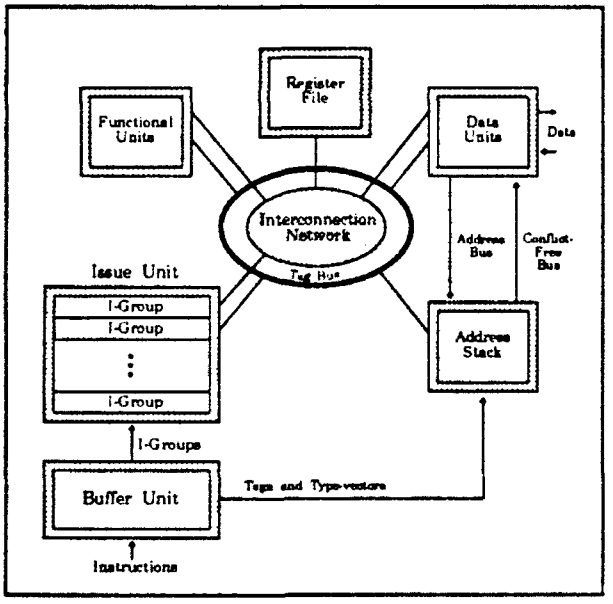


Figure 1: Fast Dispatch Stack system.

The FUs indicate instruction completion by returning tags that are issued with instructions. The FUs read operands from and return results to the Register File. Data Units receive storage instructions from the IU, generate their effective addresses, insert them into the Address Stack where address dependencies are detected, and perform dependency-free memory accesses. A Load/Store instruction set architecture is assumed.

2.1 The Buffer Unit

The BU fetches multiple instructions per instruction cache access (a fetch block) and generates four vectors for each instruction: a tag, a read-vector, a write-vector and a type-vector. Read-vector$_i$ and Write-vector$_i$ are generated from instruction q_i and specify the registers that q_i reads and writes respectively in vectors of binary elements, one element for each register. An element in position j is 1 if register j is accessed, and 0 otherwise. A type-vector specifies an instruction's type in a linear array of elements, one position for each type.

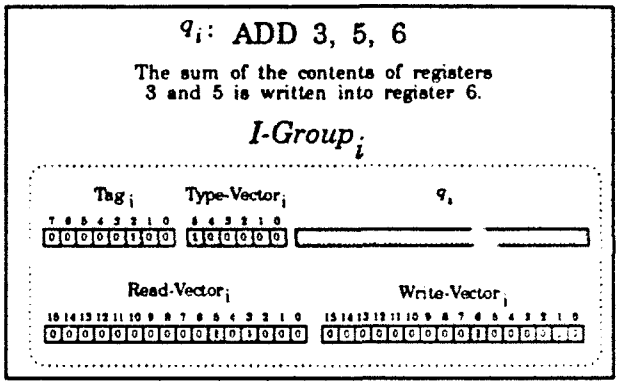


Figure 2: An example I-Group.

A tag is a vector of binary elements with a length equal to the number of available tags. Each tag is unique with one element in each tag set to 1 and the remainder to 0. Instruction q_i's tag is designated Tag_i.

An instruction together with its vectors constitute an I-Group. *I-Group*$_i$ is derived from q_i. Figure 2 shows an ADD instruction's I-Group. Tag_i and *Type-vector*$_i$ have representative assignments. I-Groups are either transferred directly to the IU, or are temporarily buffered in the BU to be forwarded later. The BU also transfers an instruction's tag and type-vector to the Address Stack.

A limited form of register renaming is performed in the BU [2]. An architected register in an instruction is given the name of one of two physical registers that are reserved for its

exclusive use. A write to an architected register causes it to be given a name different from its previous one. This scheme is used in Section 5.

2.2 The Issue Unit

The IU is composed of the Stack and the Dispatcher (Figure 3). The Stack stores I-Groups received from the BU in individual buffers (slots), detects register dependencies between instructions, repositions I-Groups, filling empty slots, and removes completed I-Groups. The Stack determines which slots contain register independent instructions each cycle.

The critical path length in an IU with 8 slots is 14 to 18 gate-delays, depending on implementation details [2]. A maximum gate fan-in of 9 is assumed. If this fan-in cannot be achieved, the critical path length must be increased by a small amount.

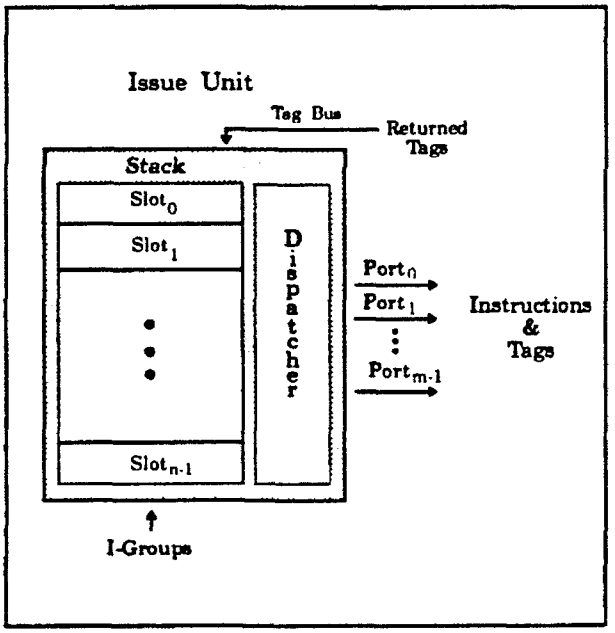


Figure 3: A block diagram of the Issue Unit.

The Dispatcher prioritizes register independent instructions each cycle based on their precedence and transfers a subset of them to output ports. A port is an entry point into the interconnection network for one instruction and its tag. The network may consist of buses, one attached to each port, with one or more FUs on each bus.

A stack of size n is an array of n slots with $Slot_0$ at the top. A slot contains conflict detection logic, tag comparison logic, registers to hold an I-Group, and logic for transferring an I-Group into the slot. The Stack may therefore hold n instructions in n I-Groups. I-Groups occupy positions in the Stack based on precedence, with the instruction of the highest precedence in $Slot_0$. Therefore, an instruction's register usages need only be compared with those of instructions in higher slots. Independent instructions are issued from the IU, contiguous, completed instructions are removed from the Stack, remaining I-Groups are moved upward, and new I-Groups are transferred into the Stack at the end of each cycle. A detailed description of the logic that performs these functions is found in [2].

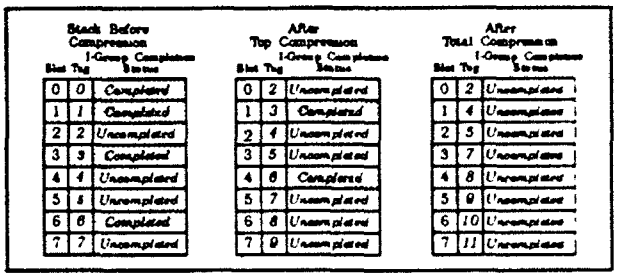


Figure 4: Top compression and total compression.

Stack compression logic selects I-Groups for removal and transfer. We have developed two selection methods: Total Compression which removes completed I-Groups from all slots; Top Compression which removes only a contiguous sequence of completed I-Groups from the top of the Stack (Figure 4).

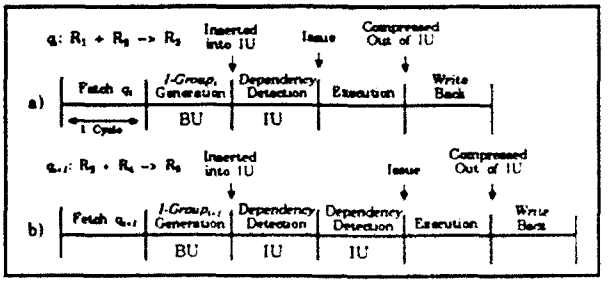


Figure 5: Illustrative pipeline timing of two ADD instructions in a FDS system with no resource conflicts.

The pipeline timing of two ADD instructions, q_i and q_{i+1}, in a FDS system with no resource conflicts is shown in Figure 5. The instructions are fetched together; however q_{i+1} is executed after q_i because it has a data dependency on it.

2.3 The Address Stack

The Address Stack is a linear array of n slots with A-Slot$_0$ at the top. There is a one-to-one positional correspondence between IU stack slots and Address Stack slots. Information on a given instruction is held in identical slot positions in the IU and the Address Stack. This correspondence is maintained with simultaneous compression operations in both stacks. A-Slot$_i$ contains the tag and type, and, if a storage instruction, an effective address when generated, address conflict, and memory access status information on the instruction in IU Slot$_i$. When the Buffer Unit transfers I-Group$_k$ to IU Slot$_j$, a copy of Tag$_k$ and Type-Vector$_k$ is transferred to A-Slot$_j$.

A data unit generates and inserts the effective address of storage instruction, q_S, into the Address Stack slot containing a copy of its tag, Tag_S. The effective address of a storage instruction, q_S, is compared with that of preceding storage instructions in the Address Stack, i.e., with those in higher slots. The tag of an address conflict free storage instruction is asserted and maintained on the Conflict-Free Bus until it completes. This bus, similar to the Tag Bus, simultaneously accommodates multiple tags. It is monitored by one or more data units for address conflict information on multiple storage instructions.

2.4 The Data Unit

A data unit generates the effective addresses of storage instructions, accesses the register file, and performs memory accesses requested by the IU and approved by the Address Stack. It may temporarily buffer data transferred between the cache and the register file to release dependencies of following instructions on storage instructions and to prefetch data. A FDS system contains one or more data units. Each may submit at most one memory access request via a dedicated connection (port) to the Cache each cycle. A request includes a storage instruction's tag. The cache returns the tags of completed accesses on the Memory Tag Bus which is monitored by the Address Stack and data units for access completions.

2.5 Storage Instruction Execution

The IU issues a storage instruction, q_S, to a data unit and then provides it with register conflict information on q_S. Based on this and address conflict information from the Address Stack, the data unit executes q_S in phases, informing the IU when to release dependencies on q_S by deleting register use representations from its I-Group.

A register used in the generation of the effective address for q_S is an address register of q_S. A register whose contents are fetched or stored by q_S is a data register of q_S.

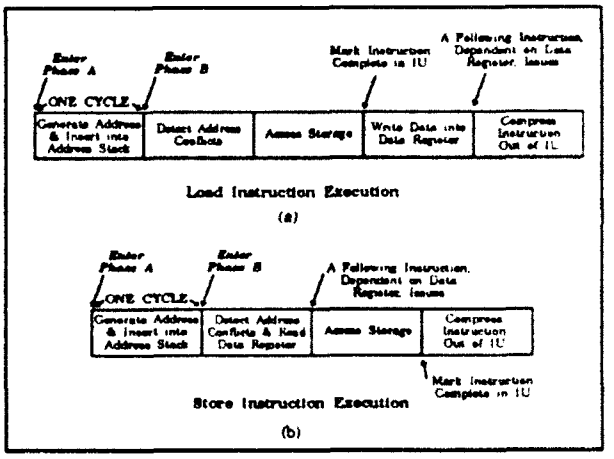


Figure 6: Illustrative load and store instruction timing assuming a 1-cycle data cache access and no register or address conflicts.

Execution proceeds in two phases that are entered in sequence: Phase A and Phase B (see Figure 6). Phase A is initiated by the issuance of q_S to an available data unit when its address registers have no conflicts. The data unit generates and inserts q_S's effective address into the Address Stack slot containing a copy of its tag, Tag_S. Phase B is initiated by the IU, when q_S's data register has no conflicts and Phase A has begun, by placing q_S's tag on the Tag Bus during a specified part of a machine cycle. When a tag in a data unit matches one on the Tag Bus, the unit

may access the data register of the associated instruction.

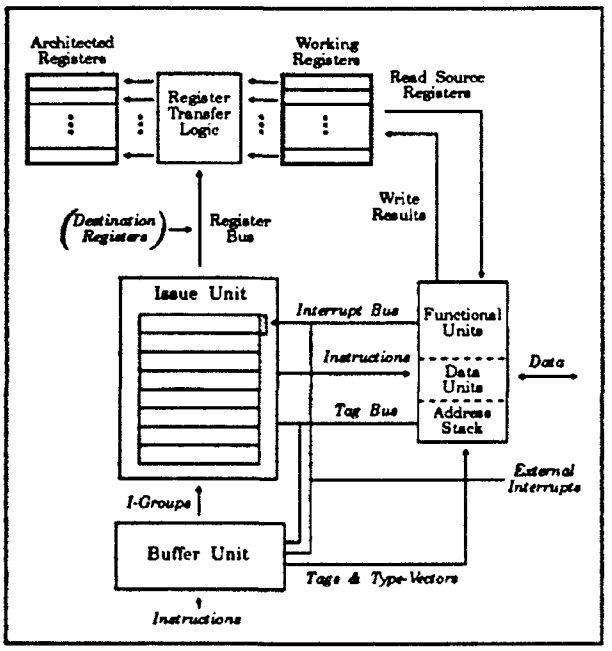


Figure 7: A FDS system with a precise interrupt capability.

3. Precise Interrupt Handling

A precise interrupt may be caused by an instruction, q_j, that is executing out-of-order, i.e., not all instructions preceding q_j have completed. To achieve a processor state that reflects that of a conventional machine that executed instructions up to q_j, instructions that precede q_j must complete execution. If an instruction q_i, which precedes instruction q_j, causes an interrupt while the conventional interrupt point for q_j is being achieved, the saved state is that of a conventional machine that executed instructions up to q_i.

Recall that top compression removes a contiguous sequence of I-Groups whose instructions are complete from the top of the Stack each cycle. Instructions are removed from the IU in the order they entered, i.e., in instruction stream order. This fact is central to the scheme presented.

Figure 7 depicts a FDS system with a precise interrupt capability. FUs and data units read operands from and write results to a set of

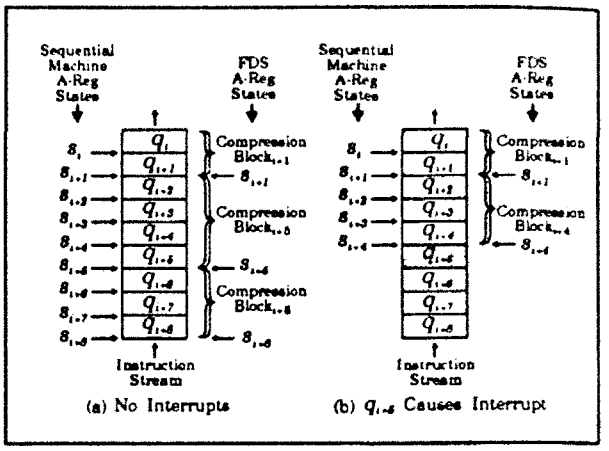


Figure 8: Comparison of A-Reg states in a FDS system and in a conventional machine.

working registers (W-Regs). The W-Regs have a one-to-one correspondence to the architected registers (A-Regs). The IU controls the transfer of none, one, or multiple results from the W-Regs to the A-Regs each cycle, causing the A-Regs to assume states that are consistent with sequential instruction execution.

Recall that the write-vector in an instruction's I-Group specifies its destination register. The contents of *W-Reg*$_i$ are transferred to *A-Reg*$_i$ if *W-Reg*$_i$ is a destination register in the write-vector of an I-Group that is compressed out.

3.1 States Assumed by the Architected Registers

The A-Regs change state only after compression operations in the IU. The A-Regs are in the state they would have in a conventional machine after it executed the last instruction in the most recently compressed out group of I-Groups. A compression-block is a group of I-Groups containing instructions that are concurrently compressed out and is identified by the last instruction in the group. Let instruction q_a be the last instruction in compression-block$_a$. Let s_a be the state of the A-Regs in a conventional machine after executing q_a.

Figure 8 illustrates how the A-Regs in a FDS system change states in two situations: once with no interrupts (Figure 8(a)), and once with an interrupt (Figure 8(b)). When no interrupts occur, the A-Regs in a FDS system experience states s_{i-1}, s_{i+5} and s_{i+8}, assuming the compression blocks

shown. The instruction stream is again processed in Figure 8(b), but this time q_{i+5} causes an interrupt. Since q_{i+5} does not complete, it is not compressed out and preceding instructions are. After instruction q_{i+4} is compressed out, the A-Regs constitute part of the process state of a conventional machine that completed q_{i+4}, s_{i+4}.

As discussed above, one or more I-Groups may be removed from the IU concurrently. The write-vectors of these I-Groups are placed on the Register Bus. Write-vector elements on the bus control the transfer of data from the W-Regs to the A-Regs via Register Transfer Logic. The assertion of a write-vector element specifying destination register R_i on the Register Bus transfers a datum in $W\text{-}Reg_i$ to $A\text{-}Reg_i$. Since each write-vector contains at most one True (1) element, the destination registers of multiple I-Groups may be specified on the Bus causing concurrent transfers.

The FDS must determine when to restore the W-Regs to a conventional machine state. An Interrupt Bus (Figure 5) connects IU $Slot_0$ with system units that can detect an interrupt-causing condition. These units may include the FUs, the cache, and units that detect external interrupt-causing conditions (e.g., I/O or sensor interrupts).

An interrupt is associated with an instruction, q_I, if q_I caused a condition that must be identified with it. The unit detecting the condition asserts q_I's tag on the Interrupt Bus. Since q_I's tag is not asserted on the Tag Bus, it does not complete and is not compressed out of the IU. Logic in IU $Slot_0$ detects the presence of an instruction associated with an interrupt by comparing an instruction's tag with those on the Interrupt Bus each machine cycle.

Let q_{i+5} be associated with an interrupt detected in a FU. Its tag is placed on the Interrupt Bus by the FU. Instruction q_{i+5} will occupy $Slot_0$ after preceding instructions have completed. The instructions preceding q_{i+5} may complete concurrently and out-of-order. The time these instructions take to complete is not lost because they are not re-executed when processing resumes. The match of q_{i+5}'s tag in $Slot_0$ with that on the Interrupt Bus causes the A-Regs to be transferred to the W-Regs, placing them in the state they would have in a conventional machine that executed instructions preceding q_{i+5}. If more than one tag is asserted on the Interrupt Bus, the interrupt taken is the one associated with the instruction of the highest precedence. It will reach $Slot_0$ before other interrupt-causing instructions.

An interrupt may be caused by a condition external to the processor (e.g., an I/O or sensor interrupt). In this case, further instruction processing is not necessary to achieve a conventional interrupt point. For fast operation, the interrupt point saved is the one associated with the instruction in $Slot_0$. The unit detecting the interrupt condition asserts all tags on the Interrupt Bus concurrently by placing all 1s on it. $Slot_0$ detects a tag match and causes a transfer of A-Regs to W-Regs.

A problem is caused by an instruction that overwrites a value in a W-Reg before it is transferred to an A-Reg. To prevent this hazard, an instruction that writes to the destination register of a preceding, completed instruction in the IU is not issued.

3.2 Precise Interrupts and Memory

We outline a scheme that causes main memory to experience states consistent with sequential instruction execution while multiple out-of-order load and store instructions are executed. A comprehensive treatment is found in [2]. If an instruction, q_i, causes an interrupt, memory is left in a state as it would be in a conventional machine that has executed all instructions preceding, but not including, q_i.

A copy-back cache is used; a datum that is stored into a copy-back cache may be transferred to main memory at a later time. We replace each cache line with a *cache line couple* composed of two cache lines. A line of data in main memory, previously mapped to a cache line, is mapped to a cache line couple. Items in one line of a cache line couple have a one-to-one positional correspondence with those in the other line. Two corresponding items form a *data couple*. Items in a data couple share the same address and are given a status of *Current* or *Pending*. At any time, one datum is current and one is pending.

A new cache line that is fetched from main memory is copied into both lines of a cache line couple. Items are marked Current in one line and Pending in the other. A store instruction overwrites the pending datum of a data couple. When a store instruction compresses out of the IU, the datum it stored (marked Pending) is marked Current and the other datum of the data couple (marked Current) is marked Pending. Recall that the Address Stack prevents a store to

the effective address of a preceding completed store instruction in the IU. Therefore, a pending datum can not be overwritten before it becomes current. Data are marked Current as store instructions are compressed out of the IU in instruction stream order, so that current data in the cache is consistent with sequential instruction execution. Only current data is copied back to the main memory.

Let instruction q_i cause an interrupt. Preceding instructions complete and are compressed out of the IU. When a store instruction is compressed out, the datum it wrote is marked current. When q_i reaches $Slot_0$ in the IU, a pending item remaining in the cache is overwritten with the current item in its data couple. Current data may be copied back to the main memory if necessary. Main memory (and the cache) is now in a state consistent with sequential instruction execution up to q_i.

3.3 Comparisons with Previous Work

Smith and Pleszkun have presented solutions (i.e., Reorder Buffer, History Buffer, and Future File) to the precise interrupt problem for systems in which at most one instruction may issue (in-order) and complete (possibly out-of-order) each cycle and store instructions are executed in-order [7]. The FDS does not have these restrictions on issuances and completions. It has, in effect, an integrated reorder buffer that may update architected registers with multiple results each cycle, causing them to "skip" some conventional machine states that are unnecessary for them to assume. We avoid a potential bottleneck in the FDS design by not routing results through instruction issuing logic as in Sohi's RUU [8].

Another approach, supporting a model of execution similar to Smith and Pleszkun's, is checkpoint repair [3]. A minimum of 3 sets of registers are used to save and restore state. A tradeoff must be made between the frequency with which state is saved and the amount of useful results that may be discarded and recalculated upon an exception. The scheme may cause instruction issuing to stall under certain circumstances.

4. Instruction Squashing

As an uncompleted branch instruction is compressed upward in the IU, the number of instructions which can be issued becomes smaller, decreasing throughput. In this section, we present an instruction squashing scheme that facilitates the use of branch prediction techniques in the FDS.

A branch instruction, q_B, transfers control to q_{B+1} or to an out-of-sequence branch target instruction. The branch target is not known until q_B executes. Since q_{B+1} is often fetched before q_B executes, a transfer of control to q_{B+1} usually causes little or no processing delay. Processing is likely to be delayed if control is transferred to an out-of-sequence branch target that is fetched after q_B executes.

Branch prediction schemes attempt to reduce processing delays by predicting and fetching the branch target instruction before the branch is executed. Branch prediction techniques have been presented by others [5][6]. A prediction accuracy of about 80% to 98% is achieved depending on the nature of the computation and the technique employed.

A key issue associated with using branch prediction in a processor that may issue multiple, out-of-order instructions is the expeditious squashing of instructions executed on an incorrectly predicted path. This is more difficult than in a conventional machine because instructions preceding and following a predicted branch instruction may coexist in the issuing mechanism and may execute concurrently and out-of-order before the branch outcome is known.

When the Buffer Unit detects a branch instruction, q_B, in a fetch block, an algorithm is used to predict the outcome of the branch. If the branch is predicted to be taken, the target instruction and instructions following it are fetched and transferred to the IU; otherwise, the fetching of instructions continues on the present path. The BU saves q_B, its tag, and its predicted outcome and forwards instruction q_B to the IU.

When dependencies allow, the IU issues q_B by placing its tag on the Tag Bus (Figure 7). The BU and the data units monitor the Tag Bus at this time. The BU executes q_B when its tag matches that on the Bus and compares its outcome with its predicted outcome. If the predicted outcome is correct, the Buffer Unit places q_B's tag on the Tag Bus just as functional units do for completed instructions. The branch instruction is then

marked complete in the IU. If q_B's predicted outcome is incorrect, the BU places q_B's tag on the Interrupt Bus and invalidates instructions in its buffer.

Assume that q_B has executed and is found to be incorrectly predicted. Since q_B's tag is not asserted on the Tag Bus, it is not marked complete in the IU and so it eventually occupies $Slot_0$ in the IU Stack. Recall that the tag of the instruction in $Slot_0$ is compared with tags on the Interrupt Bus. $Slot_0$ detects a tag match with a branch instruction and knows that this is a *branch prediction interrupt*. The contents of the A-Regs are transferred to the W-Regs and all instructions in the IU are invalidated. After the transfer, the A-Regs and the W-Regs are in the state that they would have in a conventional machine that executed instructions up to but not including q_B.

An instruction that writes to the destination register of a preceding completed instruction in the IU is not issued. This dependency control is part of the precise interrupt scheme adopted (Section 3) and prevents the overwriting of a W-Reg before its contents are transferred to the corresponding A-Reg.

Table 1: Instruction execution times.

Instruction Type	Base Machine	FDS
Store	1 Cycle	2 Cycles
Load	2 Cycles	4 Cycles
Branch	2 Cycles	3 Cycles
Integer	1 Cycle	1 Cycle
Fl. Pt.	1 Cycle	1 Cycle

The instruction squashing capability presented supports the multiple, out-of-order issue of instructions preceding and following multiple predicted conditional branch instructions. Useful work is not undone in the process. The transfer of A-Regs to W-Regs occurs in one cycle when q_B reaches $Slot_0$. The effects of the execution of instructions that followed q_B are thus eliminated in one cycle. Cycles expended while q_B moves to $Slot_0$ are productive because useful instructions preceding q_B are executing. These instructions are issued in multiples and out-of-order as dependencies allow.

5. Measurements

We use 14 Livermore Loops and the Dhrystone benchmarks to study the FDS behavior. Two traces of the 14 Livermore Loop benchmarks are used, LL_16 and LL_32, for 16 and 32 register CPUs respectively. The Dhrystone benchmark trace is for a 32-register CPU.

Throughput comparisons are made with a pipelined "Base Machine", which issues at most one instruction per cycle, in order, to one FU. The 14 Livermore Loops throughput is the harmonic mean of the individual loop throughputs.

Table 2: Benchmark throughputs on FDS systems with register renaming, precise interrupts and branch prediction over a range of branch prediction accuracies (P.A.s).

P.A.	Issue Unit Stack Size				
	4	8	12	16	32
LL_16					
50%	0.91	1.32	1.60	1.73	1.91
85%	0.92	1.36	1.62	1.78	1.95
100%	0.92	1.39	1.63	1.80	1.97
LL_32					
50%	0.91	1.32	1.60	1.73	1.94
85%	0.92	1.36	1.62	1.78	1.98
100%	0.92	1.39	1.63	1.80	2.00
Dhrystone					
50%	0.70	0.91	1.00	1.04	1.05
85%	0.71	1.04	1.15	1.19	1.21
100%	0.72	1.08	1.19	1.22	1.24
Base Machine LL_16: 0.72, LL_32: 0.72, Dhrystone: 0.65					
Base+BP Machine (100% Prediction Accuracy) LL_16: 0.79, LL_32: 0.79, Dhrystone: 0.78					

Instruction execution times of the Base Machine and the FDS are given in Table 1. They include the time necessary to eliminate the dependencies an instruction may inflict on following instructions.

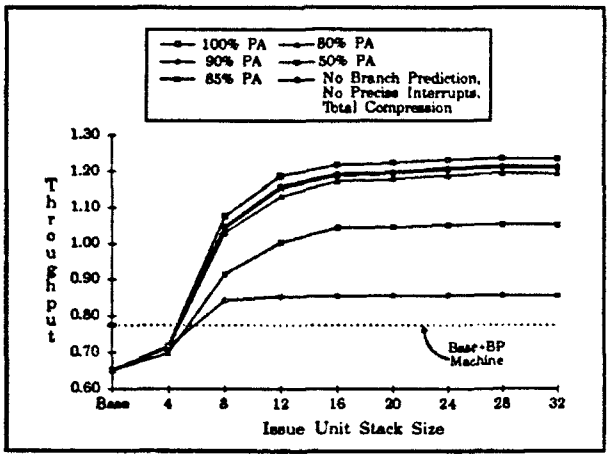


Figure 9: Dhrystone benchmark throughputs on FDS systems with register renaming.

Benchmark throughputs on FDS systems with register renaming, precise interrupts, and branch prediction mechanisms with various prediction accuracies (*PA*s) are given in Table 2. These systems have unlimited numbers of FUs. Included in Table 2 are throughputs measured on the Base Machine (Base) and on the Base Machine with 100% *PA* (Base+BP). These machines are identical except for their branch instruction execution times.

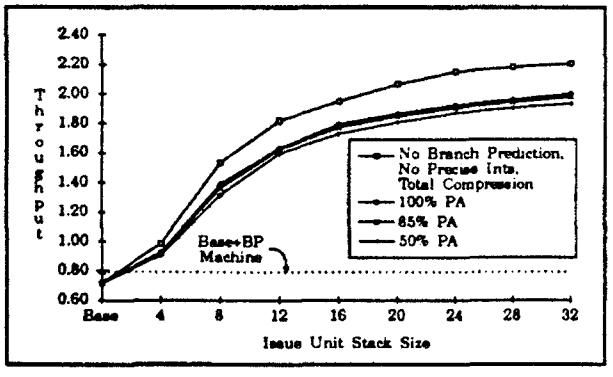


Figure 10: LL_32 Benchmark throughputs on FDS systems with register renaming.

The Base+BP machine performance on the Dhrystone benchmark (0.78) surpasses that on a FDS system with 4 slots and a 100% *PA* (0.72). In this instance, a conventional machine with a shorter branch instruction execution time has higher throughput than the FDS.

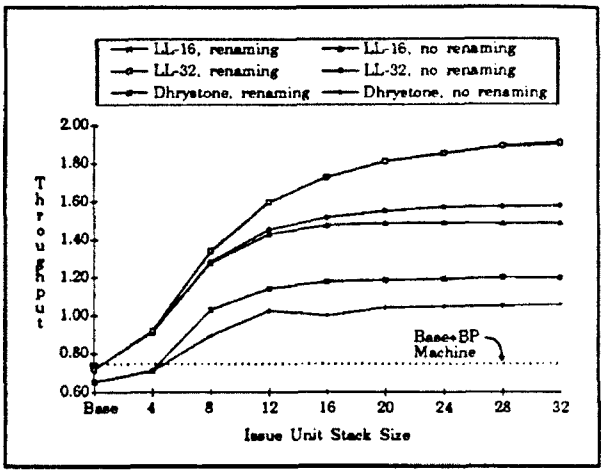


Figure 11: Throughputs on FDS systems with register renaming, 2 integer, 2 Fl. Pt., and 2 data units, a fetch block size of 4, and 4 IU ports with a 85% PA.

Figure 9 includes a plot of a FDS system with total compression and register renaming but without precise interrupts. Recall that total compression removes all completed instructions from the IU each cycle. We see that increases in Dhrystone throughput from speculative executions more than compensate for the use of top compression and the instruction dependency imposed by the squashing capability.

Table 3: Percent decrease in LL_16 throughput on a FDS system with precise interrupts compared to a FDS with total compression. Register renaming is denoted by RR.

	Issue Unit Stack Size				
	4	8	12	16	32
no RR	10.2	15.3	13.2	13.8	14.3
RR	9.2	17.5	14.4	14.4	12.3

LL_32 throughputs on systems with *PA*s of 50%, 85% and 100% are plotted in Figure 10.

Here we plot a FDS system with total compression and register renaming but without precise interrupts. We see that for the LL_32 benchmark, the benefit of speculative executions does not compensate for the negative effects of the squashing scheme adopted.

Throughputs on FDS systems with limited resources are plotted in Figure 11. These systems have 2 integer and 2 Fl. Pt. point FUs, 2 data units, a fetch block size of 4, and 4 IU ports. The benefit of register renaming is apparent in this plot.

The cost of fast, precise interrupts in a FDS system may be expressed as a decrease in throughput compared to a system with total compression. We see (Table 3) that it is less than 15% in systems with 12 or more IU slots.

6. Conclusions

We have presented a mechanism -- Fast Dispatch Stack (FDS), which performs in an integrated fashion the following tasks, indispensable for a superscalar processor:

a. the detection and dispatching of multiple instructions, possibly out-of-order, to available functional units;
b. the implementation of fast, precise interrupts;
c. the implementation of a "squashing" capability so that speculative instruction execution along predicted paths can be undertaken without attendant performance penalty.

We evaluated the design trade-offs and the performance of the resulting superscalar processor with extensive simulations. The results are presented.

We expect that the FDS we have developed can be extended to process established complex instruction sets, such as DEC Vax 780, IBM System 390, and Intel x86. Furthermore, we expect to study the interaction between compilers and the FDS and to study its behavior on other benchmarks.

Acknowledgements

The research reported herein has been supported in part by the Joint Services Electronics Program, Contract Number F49620-90-C-0039.

References

[1] R.D. Acosta, J. Kjelstrup, and H.C. Torng, "An Instruction Issuing Approach to Enhancing Performance in Multiple Functional Unit Processors". *IEEE Transactions on Computers*, Vol. C-35, No. 9, Sept. 1986, pp. 815-828.

[2] H. Dwyer, "A Multiple, Out-of-Order, Instruction Issuing System for Superscalar Processors", Ph.D. Thesis, School of Electrical Engineering, Cornell University, 1991.

[3] W.W. Hwu and Y.N. Patt, "Checkpoint Repair for High-Performance Out-of-Order Execution Machines". *IEEE Transactions on Computers*, Vol. C-36, No. 12, Dec. 1987, pp. 1496-1514.

[4] R.M. Keller, "Look-Ahead Processors". *Computing Surveys*, Vol. 7, No. 4, Dec. 1975, pp. 177-195.

[5] J.K.F. Lee and A.J. Smith, "Branch Prediction Strategies and Branch Target Buffer Design". *IEEE Computer*, Jan. 1984, pp. 6-22.

[6] S. McFarling and J. Hennessy, "Reducing the Cost of Branches". *Proceedings, 13th Annual Symposium on Computer Architecture*, June 1986, pp. 396-404.

[7] J.E. Smith and A.R. Pleszkun, "Implementation of Precise Interrupts in Pipelined Processors". *IEEE Transactions on Computers*, Vol. 37, No. 5, May 1988, pp. 562-573.

[8] G.S. Sohi, "Instruction Issue Logic for High-Performance, Interruptable, Multiple Functional Unit, Pipelined Computers". *IEEE Transactions on Computers*, Vol. 39, No. 3, Mar. 1990, pp. 349-359.

[9] G.S. Tjaden and M.J. Flynn, "Detection and Parallel Execution of Independent Instructions". *IEEE Transactions on Computers*, Vol. C-19, Oct. 1970, pp. 889-895.

An invited paper to be presented at the
1993 International Phoenix Conference on
Computers and Communications

ON INSTRUCTION WINDOWING FOR FINE GRAIN PARALLELISM IN HIGH-PERFORMANCE PROCESSORS

H. C. TORNG, School of Electrical Engineering, Cornell University
Ithaca, New York 14853

H. DWYER, IBM/Austin, and D. MARR, University of Michigan,
formerly at Cornell University

ABSTRACT Fine grain parallelism is an effective approach to enhancing processor performance through multiple and possibly out of order instruction issue and execution. We define, design and evaluate a basic central window, which works with dynamic instruction stream. Several schemes are presented to reduce the window's potential impact on processor cycle time and its hardware cost. Finally, we show that a central window can function effectively as a buffer for speculative execution and for handling interrupts and exceptions.

I. INTRODUCTION

The drive to enhance processor performance and the advances in device technology have led designers to explore various instruction issuing schemes.

Most processors, including 360/91[1] and CRAY machines [2], examine one instruction at a time. If that instruction is free of data and resource dependencies, it is issued; otherwise, the issue process stops until the relevant dependencies have been resolved. Consequently, at most one instruction is issued per cycle.

To enhance processor performance, designers have been pursuing among other things fine grain parallelism in instruction issuances; this entails the following:

1. issue more than one instruction at a machine cycle – multiple instruction issuance;

2. issue instructions out of program sequence – out of order instruction issuance.

In order to realize the intrinsic potential for multiple and out-of-order instruction issuance for a given set of programs, the designers have to endow processors with the capacity to detect execution concurrencies that exist among instructions in an instruction stream. Specifically, instructions in an instruction stream can be issued concurrently and/or out-of-order, if it can be established that there exist no hazards [3, 4] due to:

1. resource conflicts;

2. data dependences;

3. control dependences.

The detection of instructions for multiple and out-of-order issuances is an important task in the design of high-performance processors; we addressed it in this paper.

Section II discusses briefly static means to achieve multiple and out of order instruction issuance. We will introduce the instruction windowing mechanism, which extracts instructions for concurrent issuance in Section III. The basic implementation issues are explored in Section IV. In Section V, we present modifications to the basic realization

The research reported herein has been supported in part by the Joint Services Electronics Program, Contract Number F49620-90-C-0039.

schemes. The use of an instruction window to support speculative execution is outlined in Section VI. Section VII presents two window aided approaches to handling interrupt. Concluding remarks are presented in Section VIII.

II. COMPILERS FOR FINE GRAIN PARELLISM

The extraction of instructions that can be issued concurrently can be performed statically at compile time. It may come under the name of "program restructuring" [5]. In the "very long instruction word" (VLIW) approach, possible concurrent operations are identified and packed into instructions. It is advantageous to put as many operations into an instruction as possible. In order to specify many operations with one instruction, an instruction will have many fields and thus become very long [6, 7].

At compile time, the "scope"-- the number of instructions that can be examined at the same time-- is relatively large. One such technique is Trace Scheduling [8]. A trace represents a path, which may encompass several basic blocks. Instructions in these blocks can be moved and packed into very long instruction words. Since there are many traces possible for a given program, only those with "high" probabilities of occurrence are processed. In executing a trace, we run the risk that it may have to be aborted due to the fact one of the conditional branches does not produce the corresponding path. Provisions, which may be costly, have to be made in the specific trace to ensure semantic correctness for the program.

Various schemes for constructing VLIW have been reported; see for example [9, 10]. It is difficult for machines which rely entirely on compiler technology to extract fine grain parallelism due to variable memory latencies and other variable delays. In addition, dynamic branch prediction is frequently more accurate than static branch prediction. Although it is essential to have good optimizing compilers, it is equally important to have good dynamic instruction scheduling.

III. INSTRUCTION WINDOWING

Due to resource and data considerations, we cannot expect to extract all the concurrencies with static means. Dynamic scheduling must be examined. Some recent machines [11, 12] examine and issue multiple instructions concurrently per machine cycle, limited by the order and mix of instructions in the dynamic instruction stream. Multiple instructions are issued only when 3 or 4 consecutive instructions are of specific types; such restrictions severely limit the utilization of the hardware resources of the processor, and degrade its performance.

Interesting studies on dynamic instruction scheduling have been actively pursued; see for example [13].

In a classical processor, only the instruction at the head of an instruction stream is examined by the instruction unit. It can be said that it constitutes a window with a size of one instruction. In order to extract fine grain paralleism from a dynamic instruction stream, we believe that we have to endow processors with the ability to look at more instructions at any given time; thus the concept of "windowing". A processor, through an instruction window, can extract multiple, and possibly out of order, instructions for issuance.

It can be said that processors have always had an instruction window. We simply propose that the size of the window be increased from one to an integer much greater than one. A processor, being able to see more, should be able to "do" more.

IV. WINDOW IIMPLEMENTATON

The general organization of a processor is shown in Fig. 1 [14]. Note the presence of multiple functional units in a processor. As technology allows us to fabricate with increasing number of transistors per chip, adding multiple functional units becomes increasingly commonplace. A basic instruction window, called a Dispatch Stack (DS), is shown in Fig. 2 [14]; it contains essentially a

stack of registers (slots), with each housing one instruction.

The DS contains instructions which are either waiting to be issued, being executed, or waiting to be removed after execution completion.

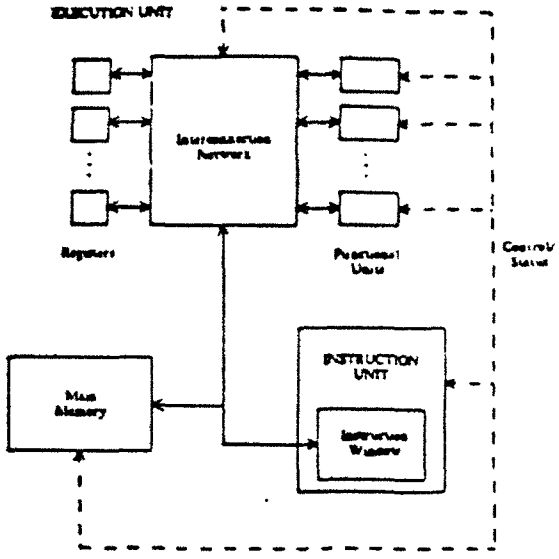


Fig. 1. General Organization of a Processor

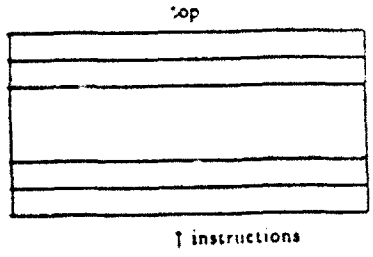


Fig. 2. A Basic Instruction Window

When one completes, the instructions below are moved up to fill the vacated slots, creating space at the bottom for new instructions to be brought in from the an Instruction Cache, or an Instruction Buffer Unit.

Instructions are selected for issue from the DS with the instructions toward the top having higher priority than those below.

In its most elementary implementation, instructions are brought into the DS when conditional branches have been resolved. An unresolved branch may halt the supply of instructions to the DS.

Data dependencies among instructions in the DS are kept with "counters" for each instruction [14]. Combinational logic circuits are used to update the counters. If no dependencies exist – all associated counters contain 0 for an instruction, it is considered independent and can be issued. Several instructions may be issued concurrently, the DS directs the routing of selected instructions to available functional units.

In evaluating such a window, we pay attention to several items:

the implementation cost -- how much chip area would this mechanism consume? This concern has been lessened due to advances in chip technology;

the performance cost -- since the circuits are needed to update the dependence counts, issue independent instructions, remove completed instructions, and finally bring in new instructions, these circuits may have an adverse effect on the clock rate. To put it differently: if we have to increase the processor cycle time, the performance gain due to multiple instruction issue may be nullified.

V. NEW IMPLEMENTATION SCHEMES

To address the concerns on its potential adverse impact on cycle time, we have developed several new schemes: the use of bit vectors [15, 16]; the use of pointers; and finally a block based window [17].

Bit Vector

In using bit vectors, each instruction is represented with an "I-Group", which consists of a tag, a type vector, the instruction itself, a

read-vector, and a write-vector; one such group is shown in Fig. 3.

A tag is a bit vector with one and only one bit set to 1; each tag identifies an instruction uniquely.

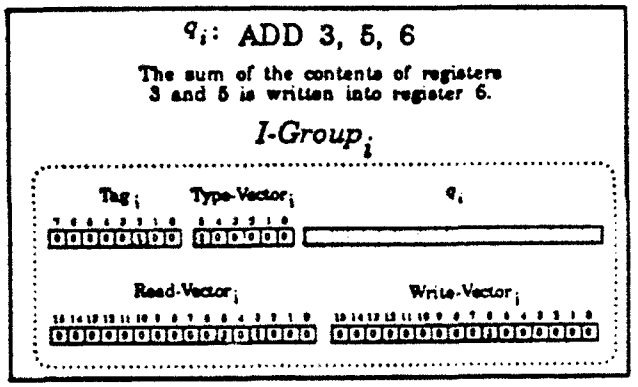


Fig. 3. An Illustrative I-Group

The type vector specifies an instruction's type with one position for each type.

In the read-vector, each bit is assigned exclusively to an architected register. A bit in the vector is set to one if and only if its corresponding register provides an operand for the intended operation. The write-vector can be explained similarly.

The formulation of the I-Groups facilitates the detection of data dependencies among instructions, the removal of completed instructions, and the dispatching of instructions to functional units.

For an 8-slot DS with a set of 16 architected registers and 4 instruction dispatching ports, Dwyer [16] performed a detailed design and found that the critical path imposes a delay of 16 gates.

Pointers

Instead of shifting instructions in the DS to maintain the proper order of appearances among them, Marr [17] proposed that two pointers, head and tail, be used. This brings about considerable reduction in circuit complexity for the DS.

There are however some complications. One of these is that the head pointer can be moved if and only if a contiguous set of instructions, including the top one, have been completed; this is termed "top compression" by Dwyer and the performance degradation due to this restriction is not significant [16]. In addition to accommodating the installation of a head pointer, top compression brings with it additional advantages, which will be discussed in Sections VI and VII.

The tail pointer indicates the last instruction entered into the window. Every time new instructions are placed into the window, the tail pointer is moved down. If the window is full, then the tail pointer must wait until the head pointer moves to give room to place new instructions.

The result of this is that the window can become "fragmented", meaning that although there may be completed instructions in the window, new instructions cannot be placed into the window until those at the head are retired.

Organizing Slots into Blocks

To reduce circuit complexity for the DS even more, Marr [17] proposed that a fixed number, which can be 1, 2, 4, ..., of contiguous slots be organized into blocks. The head pointer is moved only when the instructions in the top block are completed. Similarly, the tail pointer is moved when new instructions can be brought into the window. The advantage of organizing instruction slots into blocks is that being able to do things a "block" at a time enables considerable saving in instruction fetch, dependency evaluation, instruction issue and replacements.

The organizational diagram of a block-based instruction window is shown in Fig. 4. A new window organization, incorporating the pointer and block concepts, can be found in [17].

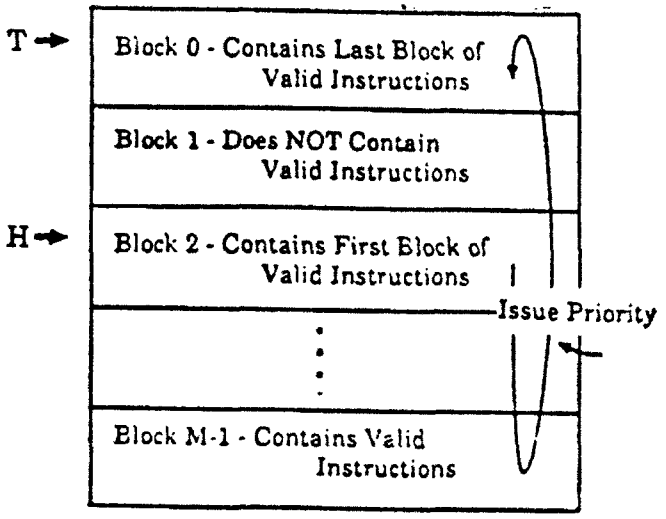


Fig. 4. A Block-based Instruction Window

VI. CONDITIONAL BRANCH HANDLING

One of the vexing problems in processor design is the handling of conditional branches. For processors employing instruction windowing, the simple fact is that we want a large number of instructions in the window to provide more instruction level parallelism. Conditional branches often make it difficult to keep the window full.

Branch prediction techniques have been develcped [18, 19] to fetch and to execute speculatively along a likely path. Even though the achieved prediction accuracy can reach 80% to 98%, some guesses will prove to be wrong. When a branch is incorrectly predicted, changes in machine state need to be removed. Doing so may involve a penalty. Instruction windowing can make an important contribution in this area [15, 16].

Conditional branch instructions along with those instructions on the predicted branch path are brought into the instruction window to be executed speculatively. The execution results are written into an additional set of registers, called the "working registers" for temporary storage and access by subsequent instructions. These results are copied into the "architected registers" once the instructions are retired from the window.

We now require that a new instruction removal mode be instituted: only those instructions at the top of the window may be retired; multiple instructions are retired at once if they form a contiguous sequence of instructions at the top of the window. This instruction retirement mode is called top compression [16].

When the prediction made for a conditional branch is found to be correct, it can be removed from the instruction window when it is included in a contiguous segment of completed instructions, including the top one in the window. Again, note that when an instruction is retired from the window, its result is made permanent by being copied into an architected register.

When the prediction made for a conditional branch is found to be incorrect, it will not be retired from the instruction window. All instructions which follow the branch are removed from the window. When the branch is retired from the window, the contents of the architected registers are copied into the working registers. The instruction window is then filled with instructions from the correct path. Fig. 5 provides a schematic diagram.

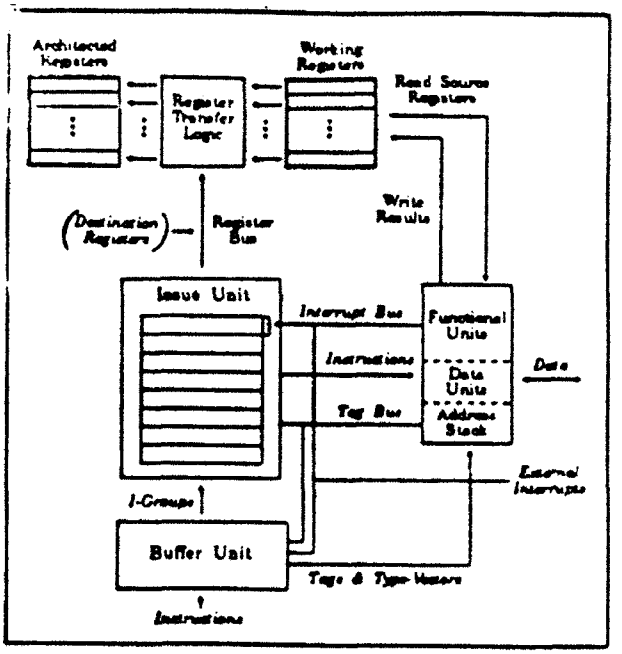


Fig. 5. A Schematic Diagram

Although the instruction window does not improve branch prediction accuracy, it can help to mitigate the performance degradation due to incorrect branch predictions.

VII. INTERRUPT HANDLING

It is imperative that processors, especially high performance processors which execute instructions concurrently and possibly out of order, handle interrupts and exceptions properly and efficiently. In this section, we present two window aided approaches to handling interrupts.

The central point in interrupt handling is the definition of the "interrupt point". The interrupt point is defined to be the instruction α, such that all instructions but not including instruction α are completed. The interrupted program will then resume precisely at instruction α. An instruction window can be used advantageously to implement this view point [16].

One-instruction Interrupt Point

Consider the structure depicted in Fig. 5. The interrupt point, is identified in the window. When it reaches the top slot, the contents of the architected registers are stored as part of the processor state and the window is flushed to received instructions for the process which services the requested interrupt.

It is to be noticed that such a processor executes multiple instructions concurrently and possibly out-of-order. However, when an interrupt is requested, a one-instruction interrupt point can be clearly implemented. Smith and Pleszkun have proposed a specific buffer to perform the same function [20]; here the instruction window, in addition to multiple and out of order instruction dispatching and other services, does it without appreciable extra cost.

Multi-instruction Interrupt Point

In evaluating an interrupt handling scheme, we have to consider three factors: latency, component cost, and performance degradation [21]. One will notice that, in implementing a one-instruction interrupt point, the processor has to wait for the completion of all the instructions that precede the interrupt point to complete; it takes time. Furthermore, all instructions that follow the interrupt point, some of which may have already been completed, have to be discarded.

With an instruction window available, Torng and Day [21] have developed an alternative approach: the instruction window is included as a component of the processor state; the saved contents of the window provides a **modified interrupt point**. A group of instructions in the window jointly define an interrupt point, where the interrupted processing should resume.

VIII. CONCLUDING REMARKS

With the increased and ever increasing device density, it is now feasible to implement an instruction window for high performance processors. The windowing technique will enable multiple and out-of-order instruction issuance; provide indispensable support for speculative execution; and implement precise, responsive, flexible interrupt handling.

Furthermore, these can be achieved without increasing the length of the critical path, which in turn determines a realistic processor clock rate.

IX. REFERENCES

[1] D. W. Anderson, F. J. Sparacio, and R. M. Tomasulo, "The IBM system/360 model 91: Machine philosophy and instruction handling," IBM J., 11, pp. 8-24, Jan. 1967.

[2] R. M. Russell, "The CRAY-1 Computer System," ACM Communications, 21, pp. 63-72, Jan. 1978.

[3] M. Johnson, Superscalar Microprocessor Design, Prentice Hall, 1991, pp. 19 - 21.

[4] H. S. Stone, **High-Performance Computer Architecture**, Addison-Wesley, 2nd edition, 1990, pp. 418 - 424.

[5] D. J. Kuck, Y. Muraoka, and S. C. Chen, "On the Number of Operations Simultaneously Executable in FORTRAN-like Programs and Their Resulting Speed-up", IEEE Trans. on Computers, 21, December 1972, pp. 1293 - 1310.

[6] J. A. Fisher and B. R. Rau, "Instruction - level Parallelism," Science, 253(5025), Sep. 1991, pp. 1232 - 1242.

[7] K. Ebcioglu, "A Compilation Technique for Software pipelining of Loops with Conditional Jumps", Proc. 20th. Workshop Microprogramming, 1987, pp. 69-79.

[8] J. A. Fisher, "Tracing Scheduling: a Technique for Global Microcode Compaction", IEEE Trans. Computers, 30, July 1981, pp. 478 - 490

[9] W. Hwu and P. P. Chang, "Exploiting Parallel Microprocessor Microarchitectures with a Compiler Code Generator", Proceedings 13th Sym. on Computer Architecture, June 1986, pp. 45 - 53.

[10] M. Lam, "Software Pipelining: An Effective Scheduling Technique for VLIW Machines", Proceedings of the SIGPLAN '88 Conf. on Programming Language Design and Implementation, June 1988, pp. 318 - 328.

[11] S. McGeady, "Inside Intel's i960CA Superscalar Processor", Proceedings ComCon, February 1990.

[12] G. F. Grohoski, "Machine Organization of the IBM RISC System/6000 Processor", IBM J. Res. Dev., 34, January 1990, pp. 37 - 58.

[13] A. K. Uht, "A Theory of Reduced and Minimal Procedural Dependencies", IEEE Trans. on Computers, 40, June 1991, pp. 681 - 692.

[14] R. D. Acosta, J. Kjelstrup, and H. C. Torng, "An Instruction Issuing Approach to Enhancing Performance in Multiple Functional Unit Processors", IEEE Trans. on Computers, 35, Sept. 1986, pp. 815 - 828.

[15] H. Dwyer and H. C. Torng, "An Out-of-Order Superscalar with Speculative Execution and Fast, Precise Interrupts", Proceedings 25th Symp. on Microarchitecture, December 1992, pp. 272 - 281.

[16] H. Dwyer, "A Multiple, Out-of-Order, Instruction Issuing System for Superscalar Processors", Ph. D. Thesis, Cornell University, 1991.

[17] D. Marr, "A Block-Based Dispatch Window for Multiple Out-of-Order Instruction Issue", M. S. Thesis, Cornell University, 1992.

[18] J. K. F. Lee and A. J. Smith, "Branch Prediction Strategies and Branch Target Buffer Design", IEEE COMPUTER, 17, Jan. 1984, pp. 6 - 22.

[19] S. McFarling and J. Hennessy, "Reducing the Cost of Branches", Proceedings 13th Sym. on Computer Architecture, June 1986, pp. 396 - 404.

[20] J. E. Smith and A. R. Pleszkun, "Implementation of Precise Interrupts in Pipelined Processors", IEEE Trans. on Computers, 37, May 1988, pp. 562 - 573.

[21] H. C. Torng and M. Day, "Interrupt handling for Out-of-Order Execution Processors", to appear in IEEE Trans. on Computers.