UNITED STATES AIR FORCE

SUMMER RESEARCH PROGRAM -- 1997

SUMMER FACULTY RESEARCH PROGRAM FINAL REPORTS

VOLUME 6

ARNOLD ENGINEERING DEVELOPMENT CENTER

UNITED STATES AIR FORCE ACADEMY

AIR LOGISTIC CENTERS

RESEARCH & DEVELOPMENT LABORATORIES

5800 Uplander Way

Culver City, CA 90230-6608

Program Director, RDL   Program Manager, AFOSR
Gary Moore   Major Linda Steel-Goodwin

Program Manager, RDL   Program Administrator, RDL
Scott Licoscos   Johnetta Thompson

Program Administrator, RDL
Rebecca Kelly-Clemmons

Submitted to:

AIR FORCE OFFICE OF SCIENTIFIC RESEARCH

Bolling Air Force Base

Washington, D.C.

December 1997

AQM01-06-1282

REPORT DOCUMENTATION PAGE

AFRL-SR-BL-TR-00-

0760

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE December, 1997	3.

4. TITLE AND SUBTITLE	5. FUNDING NUMBERS
1997 Summer Research Program (SRP), Summer Faculty Research Program (SFRP), Final Reports, Volume 6, Arnold Eng. Development Center, US Air Force Academy, and Air Logistic Centers	F49620-93-C-0063

6. AUTHOR(S)

Gary Moore

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
Research & Development Laboratories (RDL) 5800 Uplander Way Culver City, CA 90230-6608	

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSORING/MONITORING AGENCY REPORT NUMBER
Air Force Office of Scientific Research (AFOSR) 801 N. Randolph St. Arlington, VA 22203-1977	

11. SUPPLEMENTARY NOTES

12a. DISTRIBUTION AVAILABILITY STATEMENT	12b. DISTRIBUTION CODE
Approved for Public Release	

13. ABSTRACT *(Maximum 200 words)*

The United States Air Force Summer Research Program (USAF-SRP) is designed to introduce university, college, and technical institute faculty members, graduate students, and high school students to Air Force research. This is accomplished by the faculty members (Summer Faculty Research Program, (SFRP)), graduate students (Graduate Student Research Program (GSRP)), and high school students (High School Apprenticeship Program (HSAP)) being selected on a nationally advertised competitive basis during the summer intersession period to perform research at Air Force Research Laboratory (AFRL) Technical Directorates, Air Force Air Logistics Centers (ALC), and other AF Laboratories. This volume consists of a program overview, program management statistics, and the final technical reports from the SFRP participants at the Arnold Engineering Development Center, US Air Force Academy, and Air Logistic Centers.

14. SUBJECT TERMS		15. NUMBER OF PAGES
Air Force Research, Air Force, Engineering, Laboratories, Reports, Summer, Universities, Faculty, Graduate Student, High School Student		
		16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

GENERAL INSTRUCTIONS FOR COMPLETING SF 298

The Report Documentation Page (RDP) is used in announcing and cataloging reports. It is important that this information be consistent with the rest of the report, particularly the cover and title page. Instructions for filling in each block of the form follow. It is important to *stay within the lines* to meet *optical scanning requirements.*

Block 1. Agency Use Only *(Leave blank)*.

Block 2. Report Date. Full publication date including day, month, and year, if available
(e.g. 1 Jan 88). Must cite at least the year.

Block 3. Type of Report and Dates Covered. State whether report is interim, final, etc. If applicable, enter inclusive report dates (e.g. 10 Jun 87 - 30 Jun 88).

Block 4. Title and Subtitle. A title is taken from the part of the report that provides the most meaningful and complete information. When a report is prepared in more than one volume, repeat the primary title, add volume number, and include subtitle for the specific volume. On classified documents enter the title classification in parentheses.

Block 5. Funding Numbers. To include contract and grant numbers; may include program element number(s), project number(s), task number(s), and work unit number(s). Use the following labels:

C - Contract   PR - Project
G - Grant   TA - Task
PE - Program   WU - Work Unit
    Element   Accession No.

Block 6. Author(s). Name(s) of person(s) responsible for writing the report, performing the research, or credited with the content of the report. If editor or compiler, this should follow the name(s).

Block 7. Performing Organization Name(s) and Address(es). Self-explanatory.

Block 8. Performing Organization Report Number. Enter the unique alphanumeric report number(s) assigned by the organization performing the report.

Block 9. Sponsoring/Monitoring Agency Name(s) and Address(es). Self-explanatory.

Block 10. Sponsoring/Monitoring Agency Report Number. *(If known)*

Block 11. Supplementary Notes. Enter information not included elsewhere such as: Prepared in cooperation with....; Trans. of....; To be published in.... When a report is revised, include a statement whether the new report supersedes or supplements the older report.

Block 12a. Distribution/Availability Statement. Denotes public availability or limitations. Cite any availability to the public. Enter additional limitations or special markings in all capitals (e.g. NOFORN, REL, ITAR).

DOD -   See DoDD 5230.24, "Distribution Statements on Technical Documents."
DOE -   See authorities.
NASA -   See Handbook NHB 2200.2.
NTIS -   Leave blank.

Block 12b. Distribution Code.

DOD -   Leave blank.
DOE -   Enter DOE distribution categories from the Standard Distribution for Unclassified Scientific and Technical Reports.
     Leave blank.
NASA -   Leave blank.
NTIS -

Block 13. Abstract. Include a brief *(Maximum 200 words)* factual summary of the most significant information contained in the report.

Block 14. Subject Terms. Keywords or phrases identifying major subjects in the report.

Block 15. Number of Pages. Enter the total number of pages.

Block 16. Price Code. Enter appropriate price code *(NTIS only)*.

Blocks 17. - 19. Security Classifications. Self-explanatory. Enter U.S. Security Classification in accordance with U.S. Security Regulations (i.e., UNCLASSIFIED). If form contains classified information, stamp classification on the top and bottom of the page.

Block 20. Limitation of Abstract. This block must be completed to assign a limitation to the abstract. Enter either UL (unlimited) or SAR (same as report). An entry in this block is necessary if the abstract is to be limited. If blank, the abstract is assumed to be unlimited.

SFRP FINAL REPORT TABLE OF CONTENTS i-xviii

<u>**APPENDICIES:**</u>

SFRP FINAL REPORTS

PREFACE

Reports in this volume are numbered consecutively beginning with number 1. Each report is paginated with the report number followed by consecutive page numbers, e.g., 1-1, 1-2, 1-3; 2-1, 2-2, 2-3.

This document is one of a set of 16 volumes describing the 1997 AFOSR Summer Research Program. The following volumes comprise the set:

VOLUME	TITLE
1	Program Management Report
	Summer Faculty Research Program (SFRP) Reports
2A & 2B	Armstrong Laboratory
3A & 3B	Phillips Laboratory
4A & 4B	Rome Laboratory
5A , 5B & 5C	Wright Laboratory
6	Arnold Engineering Development Center, United States Air Force Academy and Air Logistics Centers
	Graduate Student Research Program (GSRP) Reports
7A & 7B	Armstrong Laboratory
8	Phillips Laboratory
9	Rome Laboratory
10A & 10B	Wright Laboratory
11	Arnold Engineering Development Center, United States Air Force Academy, Wilford Hall Medical Center and Air Logistics Centers
	High School Apprenticeship Program (HSAP) Reports
12A & 12B	Armstrong Laboratory
13	Phillips Laboratory
14	Rome Laboratory
15B&15B	Wright Laboratory
16	Arnold Engineering Development Center

SRP Final Report Table of Contents

SRP Final Report Table of Contents

SRP Final Report Table of Contents

SRP Final Report Table of Contents

SRP Final Report Table of Contents

SRP Final Report Table of Contents

SRP Final Report Table of Contents

SRP Final Report Table of Contents

SRP Final Report Table of Contents

SRP Final Report Table of Contents

SRP Final Report Table of Contents

1.   INTRODUCTION

The Summer Research Program (SRP), sponsored by the Air Force Office of Scientific Research (AFOSR), offers paid opportunities for university faculty, graduate students, and high school students to conduct research in U.S. Air Force research laboratories nationwide during the summer.

Introduced by AFOSR in 1978, this innovative program is based on the concept of teaming academic researchers with Air Force scientists in the same disciplines using laboratory facilities and equipment not often available at associates' institutions.

The Summer Faculty Research Program (SFRP) is open annually to approximately 150 faculty members with at least two years of teaching and/or research experience in accredited U.S. colleges, universities, or technical institutions. SFRP associates must be either U.S. citizens or permanent residents.

The Graduate Student Research Program (GSRP) is open annually to approximately 100 graduate students holding a bachelor's or a master's degree; GSRP associates must be U.S. citizens enrolled full time at an accredited institution.

The High School Apprentice Program (HSAP) annually selects about 125 high school students located within a twenty mile commuting distance of participating Air Force laboratories.

AFOSR also offers its research associates an opportunity, under the Summer Research Extension Program (SREP), to continue their AFOSR-sponsored research at their home institutions through the award of research grants. In 1994 the maximum amount of each grant was increased from $20,000 to $25,000, and the number of AFOSR-sponsored grants decreased from 75 to 60. A separate annual report is compiled on the SREP.

The numbers of projected summer research participants in each of the three categories and SREP "grants" are usually increased through direct sponsorship by participating laboratories.

AFOSR's SRP has well served its objectives of building critical links between Air Force research laboratories and the academic community. opening avenues of communications and forging new research relationships between Air Force and academic technical experts in areas of national interest, and strengthening the nation's efforts to sustain careers in science and engineering. The success of the SRP can be gauged from its growth from inception (see Table 1) and from the favorable responses the 1997 participants expressed in end-of-tour SRP evaluations (Appendix B).

AFOSR contracts for administration of the SRP by civilian contractors. The contract was first awarded to Research & Development Laboratories (RDL) in September 1990. After completion of the

1990 contract, RDL (in 1993) won the recompetition for the basic year and four 1-year options.

2. PARTICIPATION IN THE SUMMER RESEARCH PROGRAM

The SRP began with faculty associates in 1979; graduate students were added in 1982 and high school students in 1986. The following table shows the number of associates in the program each year.

YEAR	SRP Participation, by Year			TOTAL
	SFRP	GSRP	HSAP	
1979	70			70
1980	87			87
1981	87			87
1982	91	17		108
1983	101	53		154
1984	152	84		236
1985	154	92		246
1986	158	100	42	300
1987	159	101	73	333
1988	153	107	101	361
1989	168	102	103	373
1990	165	121	132	418
1991	170	142	132	444
1992	185	121	159	464
1993	187	117	136	440
1994	192	117	133	442
1995	190	115	137	442
1996	188	109	138	435
1997	148	98	140	427

Beginning in 1993, due to budget cuts, some of the laboratories weren't able to afford to fund as many associates as in previous years. Since then, the number of funded positions has remained fairly constant at a slightly lower level.

3.   RECRUITING AND SELECTION

The SRP is conducted on a nationally advertised and competitive-selection basis. The advertising for faculty and graduate students consisted primarily of the mailing of 8,000 52-page SRP brochures to chairpersons of departments relevant to AFOSR research and to administrators of grants in accredited universities, colleges, and technical institutions. Historically Black Colleges and Universities (HBCUs) and Minority Institutions (MIs) were included. Brochures also went to all participating USAF laboratories, the previous year's participants, and numerous individual requesters (over 1000 annually).

RDL placed advertisements in the following publications: *Black Issues in Higher Education, Winds of Change,* and *IEEE Spectrum.* Because no participants list either *Physics Today* or *Chemical & Engineering News* as being their source of learning about the program for the past several years, advertisements in these magazines were dropped, and the funds were used to cover increases in brochure printing costs.

High school applicants can participate only in laboratories located no more than 20 miles from their residence. Tailored brochures on the HSAP were sent to the head counselors of 180 high schools in the vicinity of participating laboratories, with instructions for publicizing the program in their schools. High school students selected to serve at Wright Laboratory's Armament Directorate (Eglin Air Force Base, Florida) serve eleven weeks as opposed to the eight weeks normally worked by high school students at all other participating laboratories.

Each SFRP or GSRP applicant is given a first, second, and third choice of laboratory. High school students who have more than one laboratory or directorate near their homes are also given first, second, and third choices.

Laboratories make their selections and prioritize their nominees. AFOSR then determines the number to be funded at each laboratory and approves laboratories' selections.

Subsequently, laboratories use their own funds to sponsor additional candidates. Some selectees do not accept the appointment, so alternate candidates are chosen. This multi-step selection procedure results in some candidates being notified of their acceptance after scheduled deadlines. The total applicants   and   participants   for   1997   are   shown   in   this   table.

1997 Applicants and Participants			
PARTICIPANT CATEGORY	TOTAL APPLICANTS	SELECTEES	DECLINING SELECTEES
SFRP	490	188	32
(HBCU/MI)	(0)	(0)	(0)
GSRP	202	98	9
(HBCU/MI)	(0)	(0)	(0)
HSAP	433	140	14
TOTAL	1125	426	55

4.   SITE VISITS

During June and July of 1997, representatives of both AFOSR/NI and RDL visited each participating laboratory to provide briefings, answer questions, and resolve problems for both laboratory personnel and participants. The objective was to ensure that the SRP would be as constructive as possible for all participants. Both SRP participants and RDL representatives found these visits beneficial. At many of the laboratories, this was the only opportunity for all participants to meet at one time to share their experiences and exchange ideas.

5.   HISTORICALLY BLACK COLLEGES AND UNIVERSITIES AND MINORITY INSTITUTIONS (HBCU/MIs)

Before 1993, an RDL program representative visited from seven to ten different HBCU/MIs annually to promote interest in the SRP among the faculty and graduate students. These efforts were marginally effective, yielding a doubling of HBCI/MI applicants. In an effort to achieve AFOSR's goal of 10% of all applicants and selectees being HBCU/MI qualified, the RDL team decided to try other avenues of approach to increase the number of qualified applicants. Through the combined efforts of the AFOSR Program Office at Bolling AFB and RDL, two very active minority groups were found, HACU (Hispanic American Colleges and Universities) and AISES (American Indian Science and Engineering Society). RDL is in communication with representatives of each of these organizations on a monthly basis to keep up with the their activities and special events. Both organizations have widely-distributed magazines/quarterlies in which RDL placed ads.

Since 1994 the number of both SFRP and GSRP HBCU/MI applicants and participants has increased ten-fold, from about two dozen SFRP applicants and a half dozen selectees to over 100 applicants and two dozen selectees, and a half-dozen GSRP applicants and two or three selectees to 18 applicants and 7 or 8 selectees. Since 1993, the SFRP had a two-fold applicant increase and a two-fold selectee increase. Since 1993, the GSRP had a three-fold applicant increase and a three to four-fold increase in selectees.

In addition to RDL's special recruiting efforts, AFOSR attempts each year to obtain additional funding or use leftover funding from cancellations the past year to fund HBCU/MI associates. This year, 5 HBCU/MI SFRPs declined after they were selected (and there was no one qualified to replace them with). The following table records HBCU/MI participation in this program.

SRP HBCU/MI Participation, By Year				
YEAR	SFRP		GSRP	
	Applicants	Participants	Applicants	Participants
1985	76	23	15	11
1986	70	18	20	10
1987	82	32	32	10
1988	53	17	23	14
1989	39	15	13	4
1990	43	14	17	3
1991	42	13	8	5
1992	70	13	9	5
1993	60	13	6	2
1994	90	16	11	6
1995	90	21	20	8
1996	119	27	18	7

6. SRP FUNDING SOURCES

Funding sources for the 1997 SRP were the AFOSR-provided slots for the basic contract and laboratory funds. Funding sources by category for the 1997 SRP selected participants are shown here.

1997 SRP FUNDING CATEGORY	SFRP	GSRP	HSAP
AFOSR Basic Allocation Funds	141	89	123
USAF Laboratory Funds	48	9	17
HBCU/MI By AFOSR (Using Procured Addn'l Funds)	0	0	N/A
TOTAL	9	98	140

SFRP - 188 were selected, but thirty two canceled too late to be replaced.
GSRP - 98 were selected, but nine canceled too late to be replaced.
HSAP - 140 were selected, but fourteen canceled too late to be replaced.

7.   COMPENSATION FOR PARTICIPANTS

Compensation for SRP participants, per five-day work week, is shown in this table.

1997 SRP Associate Compensation

PARTICIPANT CATEGORY	1991	1992	1993	1994	1995	1996	1997
Faculty Members	$690	$718	$740	$740	S740	$770	$770
Graduate Student (Master's Degree)	$425	$442	$455	$455	S455	$470	$470
Graduate Student (Bachelor's Degree)	S365	$380	$391	$391	S391	S400	$400
High School Student (First Year)	S200	$200	$200	$200	S200	S200	$200
High School Student (Subsequent Years)	S240	$240	$240	$240	S240	S240	$240

The program also offered associates whose homes were more than 50 miles from the laboratory an expense allowance (seven days per week) of $50/day for faculty and $40/day for graduate students. Transportation to the laboratory at the beginning of their tour and back to their home destinations at the end was also reimbursed for these participants. Of the combined SFRP and GSRP associates, 65 % (194 out of 286) claimed travel reimbursements at an average round-trip cost of $776.

Faculty members were encouraged to visit their laboratories before their summer tour began. All costs of these orientation visits were reimbursed. Forty-three percent (85 out of 188) of faculty associates took orientation trips at an average cost of $388. By contrast, in 1993, 58 % of SFRP associates took

6

orientation visits at an average cost of $685; that was the highest percentage of associates opting to take an orientation trip since RDL has administered the SRP, and the highest average cost of an orientation trip. These 1993 numbers are included to show the fluctuation which can occur in these numbers for planning purposes.

Program participants submitted biweekly vouchers countersigned by their laboratory research focal point, and RDL issued paychecks so as to arrive in associates' hands two weeks later.

This is the second year of using direct deposit for the SFRP and GSRP associates. The process went much more smoothly with respect to obtaining required information from the associates, only 7% of the associates' information needed clarification in order for direct deposit to properly function as opposed to 10% from last year. The remaining associates received their stipend and expense payments via checks sent in the US mail.

HSAP program participants were considered actual RDL employees, and their respective state and federal income tax and Social Security were withheld from their paychecks. By the nature of their independent research, SFRP and GSRP program participants were considered to be consultants or independent contractors. As such, SFRP and GSRP associates were responsible for their own income taxes, Social Security, and insurance.

8. CONTENTS OF THE 1997 REPORT

The complete set of reports for the 1997 SRP includes this program management report (Volume 1) augmented by fifteen volumes of final research reports by the 1997 associates, as indicated below:

1997 SRP Final Report Volume Assignments

LABORATORY	SFRP	GSRP	HSAP
Armstrong	2	7	12
Phillips	3	8	13
Rome	4	9	14
Wright	5A, 5B	10	15
AEDC, ALCs, WHMC	6	11	16

APPENDIX A – PROGRAM STATISTICAL SUMMARY

A. Colleges/Universities Represented

Selected SFRP associates represented 169 different colleges, universities, and institutions, GSRP associates represented 95 different colleges, universities, and institutions.

B. States Represented

SFRP -Applicants came from 47 states plus Washington D.C. Selectees represent 44 states.

GSRP - Applicants came from 44 states. Selectees represent 32 states.

HSAP - Applicants came from thirteen states. Selectees represent nine states.

Total Number of Participants	
SFRP	189
GSRP	97
HSAP	140
TOTAL	426

Degrees Represented			
	SFRP	GSRP	TOTAL
Doctoral	184	0	184
Master's	2	41	43
Bachelor's	0	56	56
TOTAL	186	97	298

SFRP Academic Titles	
Assistant Professor	64
Associate Professor	70
Professor	40
Instructor	0
Chairman	1
Visiting Professor	1
Visiting Assoc. Prof.	1
Research Associate	9
TOTAL	186

Source of Learning About the SRP		
Category	Applicants	Selectees
Applied/participated in prior years	28%	34%
Colleague familiar with SRP	19%	16%
Brochure mailed to institution	23%	17%
Contact with Air Force laboratory	17%	23%
IEEE Spectrum	2%	1%
BIIHE	1%	1%
Other source	10%	8%
TOTAL	100%	100%

APPENDIX B – SRP EVALUATION RESPONSES

1. OVERVIEW

Evaluations were completed and returned to RDL by four groups at the completion of the SRP. The number of respondents in each group is shown below.

Table B-1. Total SRP Evaluations Received

Evaluation Group	Responses
SFRP & GSRPs	275
HSAPs	113
USAF Laboratory Focal Points	84
USAF Laboratory HSAP Mentors	6

All groups indicate unanimous enthusiasm for the SRP experience.

The summarized recommendations for program improvement from both associates and laboratory personnel are listed below:

A. Better preparation on the labs' part prior to associates' arrival (i.e., office space, computer assets, clearly defined scope of work).

B. Faculty Associates suggest higher stipends for SFRP associates.

C. Both HSAP Air Force laboratory mentors and associates would like the summer tour extended from the current 8 weeks to either 10 or 11 weeks; the groups state it takes 4-6 weeks just to get high school students up-to-speed on what's going on at laboratory. (Note: this same argument was used to raise the faculty and graduate student participation time a few years ago.)

2. 1997 USAF LABORATORY FOCAL POINT (LFP) EVALUATION RESPONSES

The summarized results listed below are from the 84 LFP evaluations received.

1. LFP evaluations received and associate preferences:

Table B-2. Air Force LFP Evaluation Responses (By Type)

| Lab | Evals Recv'd | How Many Associates Would You Prefer To Get ? | | | | | | | | (% Response) | | | |
| | | SFRP | | | | GSRP (w/Univ Professor) | | | | GSRP (w/o Univ Professor) | | | |
		0	1	2	3+	0	1	2	3+	0	1	2	3+
AEDC	0	-	-	-	-	-	-	-	-	-	-	-	-
WHMC	0	-	-	-	-	-	-	-	-	-	-	-	-
AL	7	28	28	28	14	54	14	28	0	86	0	14	0
USAFA	1	0	100	0	0	100	0	0	0	0	100	0	0
PL	25	40	40	16	4	88	12	0	0	84	12	4	0
RL	5	60	40	0	0	80	10	0	0	100	0	0	0
WL	46	30	43	20	6	78	17	4	0	93	4	2	0
Total	84	32%	50%	13%	5%	80%	11%	6%	0%	73%	23%	4%	0%

LFP Evaluation Summary. The summarized responses, by laboratory, are listed on the following page. LFPs were asked to rate the following questions on a scale from 1 (below average) to 5 (above average).

2. LFPs involved in SRP associate application evaluation process:
    a. Time available for evaluation of applications:
    b. Adequacy of applications for selection process:
3. Value of orientation trips:
4. Length of research tour:
5   a. Benefits of associate's work to laboratory:
    b. Benefits of associate's work to Air Force:
6.   a. Enhancement of research qualifications for LFP and staff:
    b. Enhancement of research qualifications for SFRP associate:
    c. Enhancement of research qualifications for GSRP associate:
7.   a. Enhancement of knowledge for LFP and staff:
    b. Enhancement of knowledge for SFRP associate:
    c. Enhancement of knowledge for GSRP associate:
8. Value of Air Force and university links:
9. Potential for future collaboration:
10.   a. Your working relationship with SFRP:
    b. Your working relationship with GSRP:
11. Expenditure of your time worthwhile:

12. Quality of program literature for associate:
13.   a. Quality of RDL's communications with you:
      b. Quality of RDL's communications with associates:
14. Overall assessment of SRP:

Table B-3. Laboratory Focal Point Reponses to above questions

	AEDC	AL	USAFA	PL	RL	WHMC	WL
# Evals Recv'd	0	7	1	14	5	0	46
Question #							
2	-	86 %	0 %	88 %	80 %	-	85 %
2a	-	4.3	n/a	3.8	4.0	-	3.6
2b	-	4.0	n/a	3.9	4.5	-	4.1
3	-	4.5	n/a	4.3	4.3	-	3.7
4	-	4.1	4.0	4.1	4.2	-	3.9
5a	-	4.3	5.0	4.3	4.6	-	4.4
5b	-	4.5	n/a	4.2	4.6	-	4.3
6a	-	4.5	5.0	4.0	4.4	-	4.3
6b	-	4.3	n/a	4.1	5.0	-	4.4
6c	-	3.7	5.0	3.5	5.0	-	4.3
7a	-	4.7	5.0	4.0	4.4	-	4.3
7b	-	4.3	n/a	4.2	5.0	-	4.4
7c	-	4.0	5.0	3.9	5.0	-	4.3
8	-	4.6	4.0	4.5	4.6	-	4.3
9	-	4.9	5.0	4.4	4.8	-	4.2
10a	-	5.0	n/a	4.6	4.6	-	4.6
10b	-	4.7	5.0	3.9	5.0	-	4.4
11	-	4.6	5.0	4.4	4.8	-	4.4
12	-	4.0	4.0	4.0	4.2	-	3.8
13a	-	3.2	4.0	3.5	3.8	-	3.4
13b	-	3.4	4.0	3.6	4.5	-	3.6
14	-	4.4	5.0	4.4	4.8	-	4.4

3. 1997 SFRP & GSRP EVALUATION RESPONSES

The summarized results listed below are from the 257 SFRP/GSRP evaluations received.

Associates were asked to rate the following questions on a scale from 1 (below average) to 5 (above average) - by Air Force base results and over-all results of the 1997 evaluations are listed after the questions.

1. The match between the laboratories research and your field:
2. Your working relationship with your LFP:
3. Enhancement of your academic qualifications:
4. Enhancement of your research qualifications:
5. Lab readiness for you: LFP, task, plan:
6. Lab readiness for you: equipment, supplies, facilities:
7. Lab resources:
8. Lab research and administrative support:
9. Adequacy of brochure and associate handbook:
10. RDL communications with you:
11. Overall payment procedures:
12. Overall assessment of the SRP:
13.   a. Would you apply again?
       b. Will you continue this or related research?
14. Was length of your tour satisfactory?
15. Percentage of associates who experienced difficulties in finding housing:
16. Where did you stay during your SRP tour?
      a.   At Home:
      b.   With Friend:
      c.   On Local Economy:
      d.   Base Quarters:
17. Value of orientation visit:
      a.   Essential:
      b.   Convenient:
      c.   Not Worth Cost:
      d.   Not Used:

SFRP and GSRP associate's responses are listed in tabular format on the following page.

Table B-4. 1997 SFRP & GSRP Associate Responses to SRP Evaluation

	Arnold	Brooks	Edwards	Eglin	Griffis	Hanscom	Kelly	Kirtland	Lackland	Robins	Tyndall	WPAFB	average
# res	6	48	6	14	31	19	3	32	1	2	10	85	257
1	4.8	4.4	4.6	4.7	4.4	4.9	4.6	4.6	5.0	5.0	4.0	4.7	4.6
2	5.0	4.6	4.1	4.9	4.7	4.7	5.0	4.7	5.0	5.0	4.6	4.8	4.7
3	4.5	4.4	4.0	4.6	4.3	4.2	4.3	4.4	5.0	5.0	4.5	4.3	4.4
4	4.3	4.5	3.8	4.6	4.4	4.4	4.3	4.6	5.0	4.0	4.4	4.5	4.5
5	4.5	4.3	3.3	4.8	4.4	4.5	4.3	4.2	5.0	5.0	3.9	4.4	4.4
6	4.3	4.3	3.7	4.7	4.4	4.5	4.0	3.8	5.0	5.0	3.8	4.2	4.2
7	4.5	4.4	4.2	4.8	4.5	4.3	4.3	4.1	5.0	5.0	4.3	4.3	4.4
8	4.5	4.6	3.0	4.9	4.4	4.3	4.3	4.5	5.0	5.0	4.7	4.5	4.5
9	4.7	4.5	4.7	4.5	4.3	4.5	4.7	4.3	5.0	5.0	4.1	4.5	4.5
10	4.2	4.4	4.7	4.4	4.1	4.1	4.0	4.2	5.0	4.5	3.6	4.4	4.3
11	3.8	4.1	4.5	4.0	3.9	4.1	4.0	4.0	3.0	4.0	3.7	4.0	4.0
12	5.7	4.7	4.3	4.9	4.5	4.9	4.7	4.6	5.0	4.5	4.6	4.5	4.6
Numbers below are percentages													
13a	83	90	83	93	87	75	100	81	100	100	100	86	87
13b	100	89	83	100	94	98	100	94	100	100	100	94	93
14	83	96	100	90	87	80	100	92	100	100	70	84	88
15	17	6	0	33	20	76	33	25	0	100	20	8	39
16a	-	26	17	9	38	23	33	4	-	-	-	30	
16b	100	33	-	40	-	8	-	-	-	-	36	2	
16c	-	41	83	40	62	69	67	96	100	100	64	68	
16d	-	-	-	-	-	-	-	-	-	-	-	0	
17a	-	33	100	17	50	14	67	39	-	50	40	31	35
17b	-	21	-	17	10	14	-	24	-	50	20	16	16
17c	-	-	-	-	10	7	-	-	-	-	-	2	3
17d	100	46	-	66	30	69	33	37	100	-	40	51	46

4. 1997 USAF LABORATORY HSAP MENTOR EVALUATION RESPONSES

Not enough evaluations received (5 total) from Mentors to do useful summary.

5. 1997 HSAP EVALUATION RESPONSES

The summarized results listed below are from the 113 HSAP evaluations received.

HSAP apprentices were asked to rate the following questions on a scale from
1 (below average) to 5 (above average)

1. Your influence on selection of topic/type of work.
2. Working relationship with mentor, other lab scientists.
3. Enhancement of your academic qualifications.
4. Technically challenging work.
5. Lab readiness for you: mentor, task, work plan, equipment.
6. Influence on your career.
7. Increased interest in math/science.
8. Lab research & administrative support.
9. Adequacy of RDL's Apprentice Handbook and administrative materials.
10. Responsiveness of RDL communications.
11. Overall payment procedures.
12. Overall assessment of SRP value to you.
13. Would you apply again next year?   Yes (92 %)
14. Will you pursue future studies related to this research?   Yes (68 %)
15. Was Tour length satisfactory?   Yes (82 %)

	Arnold	Brooks	Edwards	Eglin	Griffiss	Hanscom	Kirtland	Tyndall	WPAFB	Totals
# resp	5	19	7	15	13	2	7	5	40	113
1	2.8	3.3	3.4	3.5	3.4	4.0	3.2	3.6	3.6	3.4
2	4.4	4.6	4.5	4.8	4.6	4.0	4.4	4.0	4.6	4.6
3	4.0	4.2	4.1	4.3	4.5	5.0	4.3	4.6	4.4	4.4
4	3.6	3.9	4.0	4.5	4.2	5.0	4.6	3.8	4.3	4.2
5	4.4	4.1	3.7	4.5	4.1	3.0	3.9	3.6	3.9	4.0
6	3.2	3.6	3.6	4.1	3.8	5.0	3.3	3.8	3.6	3.7
7	2.8	4.1	4.0	3.9	3.9	5.0	3.6	4.0	4.0	3.9
8	3.8	4.1	4.0	4.3	4.0	4.0	4.3	3.8	4.3	4.2
9	4.4	3.6	4.1	4.1	3.5	4.0	3.9	4.0	3.7	3.8
10	4.0	3.8	4.1	3.7	4.1	4.0	3.9	2.4	3.8	3.8
11	4.2	4.2	3.7	3.9	3.8	3.0	3.7	2.6	3.7	3.8
12	4.0	4.5	4.9	4.6	4.6	5.0	4.6	4.2	4.3	4.5
Numbers below are percentages										
13	60%	95%	100%	100%	85%	100%	100%	100%	90%	92%
14	20%	80%	71%	80%	54%	100%	71%	80%	65%	68%
15	100%	70%	71%	100%	100%	50%	86%	60%	80%	82%

PARALLEL PROCESSING FOR TUBINE ENGINE MODELIJNG AND TEST DATA VALIDATION

Csaba Biegl
Research Assistant Professor
Department of Electrical Engineering

Vanderbilt University
Box 1840 Station B
Nashville, TN 37235

PARALLEL PROCESSING FOR TUBINE ENGINE MODELING AND TEST DATA VALIDATION

Csaba Biegl
Research Assistant Professor
Vanderbilt University
Department of Electrical Engineering

Abstract

The Arnold Engineering Development Center (AEDC) uses a variety of turbine engine simulation and test data analysis programs to enhance its testing capabilities. Traditionally, these programs have been executed off-line, sometimes at processing speeds that are several orders of magnitude slower than the speed at which the test data is acquired. Increasing demand for faster test turnaround times and data accuracy necessitates the integration of on-line simulation and data verification into the testing process. This paper describes two efforts towards this goal: a study to speed up a Computational Fluid Dynamics (CFD) solver code which is used in a three dimensional turbine engine compressor model, and a parallel signal feature extraction system.

PARALLEL PROCESSING FOR TUBINE ENGINE MODELING
AND TEST DATA VALIDATION

Csaba Biegl

Introduction

An important part of the testing of turbine engines at AEDC is the application of engine models for data validation. Turbine engine models are numerical simulation programs that model the behavior of the engine or some of its components under various operating conditions. Some models are based on exact modeling of the physical (transport, thermal and mechanical) processes inside certain engine components. Other models are less detailed, component-level simulations which try to approximate overall engine parameters from the engine layout and component characteristics.

Turbine engine models can be used for several purposes like design analysis, modeling the interactions between the airframe and the engine, education, etc.. From the point of view of an engine test facility, like AEDC, the most beneficial use of the models is to aid the testing operations. This requires models that are fairly accurate and have been verified in different operating modes of the engine. The development of accurate engine models is a time-consuming process that uses several possible input data sources:

- Data, component characteristics and sometimes incomplete models from the engine manufacturer

- Parameters inferred based on the geometry (duct diameters, etc..) and operating parameters (RPM, etc..) of various engine components

- Measurement data obtained from engine tests.

Typically, at the beginning of a test sequence, the amount of available data does not allow the creation of an accurate engine model. After test runs, some of the measured test parameters are used to verify and to refine the accuracy of the models. Simultaneously, the model is also used to determine some other engine parameters that cannot be measured in the test configuration. Thus the collection of test data and the refinement of the model are two interacting activities which proceed in parallel during the course of a typical engine test sequence.

Once the model has been refined to an acceptable accuracy level it can also be used to validate the test data. A turbine engine test facility is a very complex industrial plant (wind tunnels, compressors, extensively instrumented test cells, etc..) which must be highly reconfigurable in order to support various engine types and operating conditions. The probability of a component failure or a misrouted connection in

such a plant is quite high. Most of the time these plant failures will manifest in invalid measurement data. Since running the engine test facility is quite expensive, it is imperative that such plant failures be detected and corrected as early as possible. Accurate engine models play a very important role in the validation of the test data.

To best support the testing operations the engine model has to be available on-line during the test periods. Ideally, the simulation speed should be real-time or even super real-time so that events in the engine can be analyzed (or even predicted) as they happen. Most currently used engine models do not meet this requirement. On current single processor computers their execution speed is two or more orders of magnitude slower than the real physical process they simulate. This, of course, also depends on the computer platform being used. Future advances in hardware and software technology may narrow the gap. Alternatively, most engine models could be adapted for parallel execution.

In most test scenarios the models can be quite useful even if they cannot execute in real-time. Typical engine tests proceed with alternating periods of relatively short (few seconds to minutes) engine maneuvers (acceleration, deceleration, throttle snap, etc..) and longer (several ten minutes to few hours) setup periods for the next test point. The engine models can be used in the setup periods to analyze and validate the data from the previous maneuver. Many engine models execute fast enough for this kind of usage on today's high performance workstations.

Many times test data validation can be further enhanced by the application of higher-level signal processing techniques, called feature detection algorithms. These algorithms can detect changes in the measured signals that would normally be hidden by the "normal" signal component or noise. Such detectable features include drift, noise, level shift, peaks and spikes. Such algorithms can be used either with or without an accompanying model-based test data validation system. Their advantages are that there is no need to develop a custom simulation for every engine type being tested and that they are able to verify a wider range of test parameters. (Model-based verification methods require a careful selection of a set of test parameters which "drive" the simulation. The rest of the measured parameters are not verified.) On the other hand, feature detection methods typically analyze every parameter on its own, thus any parameter coherency checking offered by "true" model-based verification is lost. Nevertheless, the quality of the per parameter test data verification implemented using these techniques is far superior to older methods based on range checking or other simple calculations. Similarly to model-based data validation, feature extraction algorithms also demand high performance computing environments.

Overview of modeling techniques

AEDC currently uses different types of turbine engine or engine component simulation codes. Many of these are potential candidates for on-line execution in the test environment, provided that the data interfacing and execution speed issues can be resolved.

- Generic engine models (ATEST): ATEST models are built from pre-coded engine component model blocks, like models for compressors, burners, turbines, ducting, etc. These blocks model the overall characteristics of the given component, they are not detailed models of the physical processes inside the component. A complete engine model is built by specifying the components, their parameters (for example compressor map tables, sizes, etc.) and interactions. A numerical solver module is used to match the various component input/output parameters until the whole model converges to the specified operating point. ATEST can be used to perform either steady state or dynamic simulations. Sometimes the ATEST program is used as part of a more complex simulation system.

- Detailed component models (DYNTECC, ATEC, etc.): These simulations model the actual physical processes inside a single engine component. For example, the DYNTECC program models the processes inside the stages of a turbine engine compressor. It divides the compressor volume along a one-dimensional grid into up to 60 axial segments and calculates mass, flow, energy, etc. balances for each of these segments. It also allows the segmentation of the compressor volume into up to eight circumferential segments (each of which can again have up to sixty axial segments).

- Three dimensional grid models (TEACC): These simulations use the techniques of Computational Fluid Dynamics (CFD) to even more accurately model the processes inside various engine components. TEACC models turbine engine compressors by dividing their volume along a three dimensional grid, which allows much more accurate simulation. Similarly to other CFD codes, TEACC is very computationally demanding. For this model category execution on a supercomputer or parallel machine is necessary even in off-line use.

Overview of feature detection techniques

The feature detection code, which is currently being used at AEDC for test data validation, was developed at NASA Lewis Research Center. The goal of the package is to detect significant events in test data that may signal either a change in the operating conditions of the test article or a sensor failure. Detected event types include:

- Flat data (no signal)
- Noise
- Drift

- Level shift
- Peaks
- Spikes

All event detection algorithms require a time-tagged input data file for their operation. They provide reasonable defaults for the detection parameters (thresholds, etc.), but these can be overridden if necessary.

The feature detection package was written in C and ported to various UNIX environments. Originally it was intended as a stand-alone program, which is executed from the command line and takes input data and event detection parameters from data files of the appropriate format. Output is a simple log of detected events written into ASCII text files.

Parallel execution of a CFD solver on multiple PC class machines

One of the tasks of the Summer Research Program was to examine the feasibility of using affordable PC class hardware for running parallelized CFD type models. This work was driven by the development of the TEACC turbine engine compressor modeling code. At this time TEACC is undergoing extensive testing and verification using various test cases. Typical test runs can last hours on high-end PC class machines (200+ MHz P6 or better – these PC-s are very close to modern single processor workstations in terms of computing performance). Although AEDC uses High Performance Computing (HPC) platforms for CFD modeling and test data evaluation, the availability of these machines for code development and verification is limited. All of these factors contributed to the requirement of being able to execute TEACC on networked PC and/or workstation clusters.

TEACC models the flow and pressure fields inside the compressor of a turbine engine using a grid of small volume cells called nodes in the CFD terminology. It uses custom code to account for the special airflow, mechanical and thermodynamic effects characteristic of the operation of a rotating compressor. Once this code computes the necessary initial and boundary conditions a "standard" CFD solver is invoked to generate the flow, pressure, etc. fields. This process is repeated until the solution converges with acceptable error terms. In the current version of TEACC the NPARC CFD code is used.

The first step in evaluating the feasibility of the task was to benchmark the CFD solver code itself on the intended target architecture. Version 3.0 of the NPARC code was used for this purpose. This version of the code was already set up for parallel execution using a per block parallelization scheme. Blocks provide a way to group nodes in NPARC, in the benchmarked version it is possible to execute the solver for each such block in parallel on different machines. The NPARC solver implements the necessary exchange of block boundary conditions after every iteration of the solution.

NPARC 3.0 is distributed in source form for Windows based PC-s and UNIX workstations. Only the UNIX versions include the code necessary to build the parallel solver. For this reason the NPARC

benchmarking task was performed on Pentium class PC-s running the Linux operating system. Linux is a public domain POSIX compliant UNIX clone. Although the UNIX NPARC source distribution contains configuration files for many supported UNIX flavors (Sun, HP, SGI, etc.), Linux was not supported. Thus the first task was to port NPARC to Linux. The C (gcc) and FORTRAN (g77) compilers of the GNU compiler family (which are part of the standard Linux distribution) were used to build the code. Porting involved creating the necessary configuration files and making a few system-dependent modifications to the code. The NPARC distribution includes a few test cases and baseline outputs for these. The Linux port of the NPARC solver was verified using these.

The next step was to build and benchmark the parallel distributed version of the NPARC solver. NPARC uses a parallel communication library to implement the necessary data exchange functions. The distributed NPARC 3.0 version can use either the PVM (Parallel Virtual Machine) or the MPI (Message Passing Interface) library. PVM is an older parallel message-passing library, originally developed at Oak Ridge National Laboratories. MPI is the emerging standard for parallel message passing applications. (In fact, ORNL has just announced that it does not support PVM any longer.) Parallel versions of the NPARC code were built using both libraries. Building of the PVM version went seamlessly, minor configuration file and source changes were necessary for the MPI version – probably due to the fact that MPI is a much newer (and still changing) environment.

Benchmarking of the NPARC solver was performed using one of the standard three-dimensional test cases distributed with the source (only 3D cases were relevant as TEACC uses its built-in CFD solver in 3D mode), and an AEDC supplied test case whose dimensions better approximate the "typical" TEACC problem. In parallel mode NPARC can use N + 1 processors, where N is the number of blocks in the test case. The first processor only serves as a coordinator, the remaining ones are each responsible for computing the solution of their assigned block. All runs were executed on 133 MHz Pentium computers with 128 Mbytes of RAM installed, connected with a dedicated 10 Megabits/sec Ethernet network, running the Linux operating system.

The test case from the NPARC distribution had four node blocks with a total of 2875 nodes (510 + 810 + 510 + 1045). In theory this case could have used up to five processors in parallel mode. However, benchmarking was discontinued after completing the two worker (three processor) runs, as no speedup was detectable. The following execution times were measured (wall clock):

1-7

Sequential: 285 sec

PVM parallel: 762 sec

MPI parallel: 480 sec

The AEDC supplied large test case had five blocks with a total of 970417 nodes (429000 + 106053 + 240149 + 138375 + 56840). There were only two computers available which had the sufficient amount of RAM installed for this test case (100+ Mbytes).This also meant that the master (coordinator) process was sharing a processor with the first worker. This did not cause any problem, as the NPARC master process uses very little CPU time in parallel mode. The execution times for the large test case were as follows:

Sequential: 8 hours 15 minutes

PVM: 4 hours 42 minutes

MPI: 4 hours 27 minutes

Analysis of the results shows that the standard NPARC test cases are not suitable for parallel execution. These test cases contain relatively few nodes per block, which makes the communication overhead the dominant factor in the overall execution time. The large AEDC test case shows a close to linear speedup for the available (unfortunately too small) range of processors.

A surprising result is the poor performance of the PVM version of the code. A separate test run of the parallel PVM version with all processes (coordinator + workers) *on the same host*, gave an interesting insight into this result. In theory, these processes among themselves should have consumed all the available CPU time on the host. (At least one of the processes should always be ready to run, and there is very little disk activity – i.e. waiting for the disk – after program startup.) In practice, *the system was idle* for a significant fraction (30...40%) of the time while the PVM version of parallel NPARC was executing. The MPI version of the code was free from this effect. A possible explanation for this result is that PVM may use short programmed delays as it waits for incoming messages which may add up to the observed idle periods. (Actually, such behavior may be desirable on workstation clusters where multiple users are logged in and some of them are trying to run interactive applications.) Note that these remarks are relevant only to

the versions of the PVM and MPI libraries used for the testing (public domain versions from Oak Ridge and Argonne National Labs, respectively), vendor supported commercial versions may behave differently.

An other unexpected result was that the memory requirements of the distributed NPARC version did not decrease. It was expected that the different worker processes to which the node blocks were assigned would allocate only as much memory as necessary to store their assigned nodes. In reality each process of a parallel NPARC process group allocated the same amount of memory, which was identical to the amount requested by the sequential NPARC version for the same test case. This shortcoming may significantly limit the number of machines in a distributed cluster that may be suitable for running parallel NPARC on a larger test case.

Parallel feature detection

The other parallel processing related task completed during the Summer Research Program was the creation of a parallel control program and an interactive graphical front end to the NASA feature detection code described above. This system is used in addition to or in place of the parallel model-based data validation system described in [1] to validate so called Transient Data. In the AEDC terminology "transient data" describes engine and test cell operational parameters (like fuel flow, thrust, ambient pressures, etc.). Such data is acquired from various data sources at varying rates.

During testing operations this data is collected and made available on a file server which is part of AEDC's Analysis Network, which is a non-public network of mostly Silicon Graphics™ workstations. Data is stored in files on a per data point basis with new files becoming available as sampling for successive data points is completed. Typical data point files contain several thousand parameters sampled at a few hundred samples per second (extra repeated samples are inserted for parameters which are collected at lower rates) for a duration ranging from tens of seconds to a few minutes. Depending on testing schedule such data point files may be generated with only a few minutes of "idle" time between them.
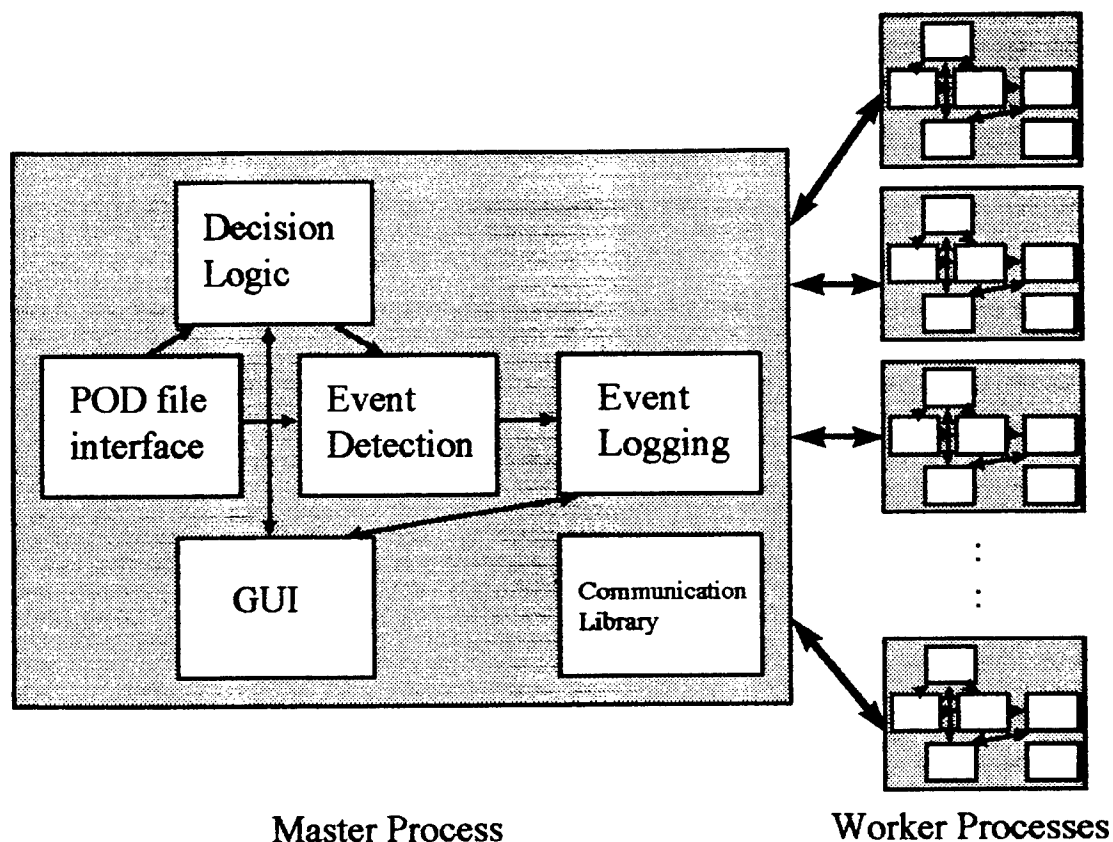
Figure 1. Feature Detection Program Architecture

The command line/file oriented original NASA feature detection code is not adequate to verify this data if the goal is to try to process data points as soon as they are taken. Improvements to both processing speed and data presentation to the user are needed to meet this goal. A parallel feature detection program which uses the original NASA routines was created to satisfy these goals. The main components of this program can be seen in Figure 1. These components include:

- *The POD file interface* includes routines to read the Parameter Oriented Data (POD) file format in which the transient parameters are stored. Besides the data itself this file format stores additional

information including parameter names, sampling rate, data point number and length, test classification, etc.

- *The Event Detection* routines are based on the original NASA code. Modifications include the ability to process data passed as subroutine parameters instead of the original file input. A few memory management related fixes were also necessary for the continuous operation of the code.

- *The Graphical User Interface (GUI)* is built using the elements of the universal simulation user interface toolkit described in [1]. It allows the user to control the operation of the program (input data source, parallel/sequential operation, select which events are detected on a per parameter basis, etc.) and to display the results of the event detection.

- *The Decision Logic* routines determine the operating conditions of the engine. During certain maneuvers (fast acceleration/deceleration, etc.) the event detection has to be turned off to avoid a large number of "false positive" events. This is accomplished by monitoring the rate of change of a few key engine parameters (shaft speeds, thrust, fuel flow, etc.) and enabling/disabling the event detection using user settable tolerances for these. Additionally, the length of the data samples for which feature detection is performed is also controlled here. Normally the code reads a few seconds' worth of data and passes it to the feature detection routines. Under certain conditions, like a slow acceleration or deceleration, event detection does not have to be turned off entirely, but it has to be performed for shorter sample records, again, to avoid false positives.

- *The Event Logging* routines generate a more detailed output than the original NASA feature detection code. Additionally to the original output files, event summary files are generated on a per data point and per entire run basis, on-line plot data and event history is generated for the Graphical User Interface. It is also possible to save all events detected during a run in a format which can be reloaded into the system during a successive run. This makes it possible to flag only newly detected events for successive tests or data points.

1-11

- *The Communication Library* routines make it possible (though not mandatory – the same code can be executed in sequential mode) to start several copies of the event detection code, dedicate one of them as the coordinator/GUI process, the rest as workers, and process large amounts of data in a parallel version of the program. This is made possible by the fact that detecting events in samples of different parameters can be done independently of each other. For portability reasons a simple TCP-IP based library was used here instead of a more comprehensive environment like PVM or MPI, which were not available on all targeted platforms. The routines of this library are used for the parallelization of the feature detection processing in a manner which allows the coordinator/GUI process to adapt to the different execution speeds of the worker processes. Faster worker processes will be assigned more parameters to check, slower workers less.

The parallel feature detection code was installed on the Silicon Graphics™ workstations of AEDC's Analysis Network. Figure 2 shows a typical screen image taken during the operation of the program which shows plots of the "special" parameters (which are used to determine the state of the engine) and displays of the last few detected events. Execution speed timing tests have shown that the system can process test data from a recent engine test at faster than real-time speeds using four processors (one master, three workers). Increasing the number of worker processes resulted in an essentially linear speedup which is not surprising considering the fact that different parameters can be processed independently. Unfortunately, this speed can be achieved only if the data files are copied to local hard disk on each host from the central file server. Otherwise, acquiring the data from the file server becomes the bottleneck.
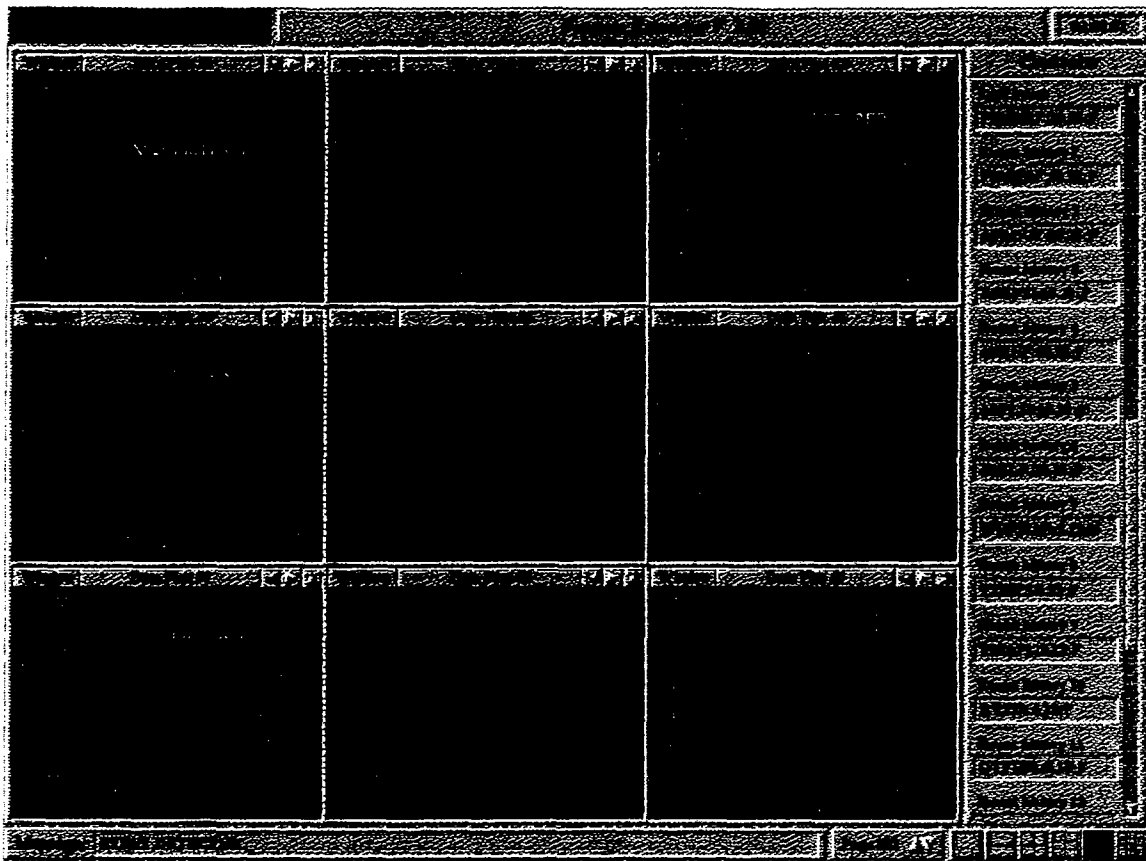
Figure 2. Example feature detection program screen

Conclusions and future work

    The goal of the research described in this document was to enable the introduction of existing turbine engine modeling technologies into the test environment. Work has progressed on two fronts: evaluating the feasibility of running a parallel CFD solver on a network of inexpensive PC-s, and creating a parallel system for the on-line detection of events in test data.

    The NPARC results show that PC technology has evolved to a point where it can be applied to large scale numerical modeling problems. Work has already begun on the actual TEACC code to develop a parallel version of it for distributed processor networks. The experience gained with porting and benchmarking the parallel NPARC code will be very useful for parallelizing TEACC. Nevertheless, TEACC will have its own parallel logic, instead of relying on the possible parallelization in the embedded CFD

1-13

solver. This decision is supported by the results of some preliminary execution timing profiles gained on a single processor version of TEACC. Additionally, this will make it easier to potentially replace the currently used NPARC CFD solver with a different code, while preserving parallel execution.

The result of the feature detection work is a parallel code which – given enough computing resources – is potentially capable of processing all generated transient data in real-time. As this program processes a very large amount of data, the bandwidth of the currently available AEDC data system seems to be the limiting factor. There is a new real-time data system being developed at AEDC which is based on Reflective Memory™ technology. This data network promises to have sufficient bandwidth to achieve the goal of real-time transient data validation (either model or feature detection based). Preliminary work has already begun to interface the system to this new network.

References

1. Biegl, C.: "Towards the Real-time Execution of Turbine Engine Simulation Programs", AFOSR SRP Final Report, 1995

Design of a Mass Spectrometer Sampling Probe for the
AEDC Impulse Facility

Frank G. Collins
Professor
Department of Mechanical and Aerospace Engineering

University of Tennessee Space Institute
Tullahoma, TN 37388

DESIGN OF A MASS SPECTROMETER SAMPLING PROBE FOR AEDC IMPULSE FACILITY

Frank G.Collins
Professor
Department of Mechanical and Aerospace
Engineering and Engineering Science

Abstract

The work reported was part of the NUNN/DLR FPST Project, which is an International Cooperative Research and Development Project between Arnold Engineering Development Center (AEDC) and the German Research Center for Aviation and Space Travel (DLR), to obtain hypersonic data for code validation in Free Piston Shock Tunnel (FPST) facilities. Of major concern is the measurement of the chemically pure run time of the facility. A time-of-flight mass spectrometer (TOFMS) probe sampling system which will be used to make this measurement is described. Computational fluid dynamics (CFD) computations were performed to design a system of three coaxial skimmers cones that will extract a gas sample in the form of a molecular beam and transmit the beam to the TOFMS. Interpretation of the mass spectra and integration of the TOFMS with the FPST are discussed, including the sizing of the various probe chambers and the associated vacuum pumps.

DESIGN OF A MASS SPECTROMETER SAMPLING PROBE FOR AEDC IMPULSE FACILITY

Frank G. Collins

Introduction

The work reported here was part of the NUNN/DLR FPST Project, which is an International Cooperative Research and Development (ICR&D) Project between the Arnold Engineering Development Center (AEDC) and the German Research Center for Aviation and Space Travel (DLR). Computational Fluid Dynamics (CFD) codes are vital to the future design of hypersonic flight systems, and adequate data for code validation are essential in the development of the necessary codes. Both the AEDC and DLR have chosen an advanced facility concept known as the Free Piston Shock Tunnel (FPST) as the test bed for acquiring the data. The AEDC FPST facility is referred to as the Impulse Facility.

The physics of the hypersonic flow in FPST facilities requires the use of advanced measurement techniques to obtain the required data. The first objective for these measurement techniques is to determine the free stream flow parameters, and the most essential measurement needed is the determination of the chemically pure run time of the facility. The end of the chemically pure run time of the facility occurs when the driver gas (helium) reaches the test section. Several advanced measurement techniques are being prepared to determine the time of the helium arrival; laser diode absorption of seed molecules in the helium, filtered Rayleigh scattering, pulsed electron beam fluorescence, and time-of-flight mass spectrometry (TOFMS). The design of the sampling probe for the TOFMS will be described in this work.

Impulse Facility Operation

The AEDC Impulse Facility is capable of generating very high enthalpy hypersonic flows for test times of a few milliseconds. This is accomplished by burning gun powder to generate combustion gases which drives an expendable piston into the helium driver gas, adiabatically compressing the helium. The test gas (air) is placed

in a shock tube, downstream of the primary diaphragm which separates it from the helium. At a given helium pressure the primary diaphragm bursts, initiating a shock wave that propagates down the shock tube, compressing the air. A shock wave reflects from the end of the shock tube, and propagates upstream, thereby compressing the air even further to stagnation conditions. The compressed air then ruptures the secondary (nozzle) diaphragm and flow is initiated through a hypersonic nozzle. Air, at high stagnation enthalpy and pressure, is then accelerated through the nozzle, yielding a steady flow of air in the test section for a few milliseconds. The test is terminated by the arrival of the helium in the test section. Impulse Facility operation and initial calibration are completely described by Blanks (Ref. 1).

Measurement of Free Stream Gas Composition

Many hypersonic test articles, such as scramjet engines, depend upon having chemically pure air in the facility test section free stream. Thus, it is important to measure the chemical composition of the free stream gas during the Impulse Facility test time, and especially to determine the driver gas (helium) arrival time in the test section.

The goal of the measurement system described in this report is to measure the relative composition of the gas species in the test section free stream. This will be accomplished by sampling a portion of the stream with a conical probe system. The sampled gases will be partially ionized and the ions separated by mass in a time-of-flight mass spectrometer (TOFMS). The problems associated with this task will be described along with analysis that has been performed in an attempt to show how to relate the TOFMS mass spectra to the free stream gas composition.

The need for the TOFMS probe is illustrated by the measurements of Skinner (Ref. 2). He made measurements in the Australian T3 hypersonic shock tunnel facility and found that helium contaminated the test section flow immediately upon the initiation of the test whenever the stagnation enthalpy exceeded a

value of 12 MJ/kg; the value of stagnation enthalpy in the Impulse Facility will exceed this magnitude during the present series of tests. Thus the T3 facility possesses chemically pure run time under high enthalpy test conditions. Since optical techniques cannot measure the entire gas composition at any instant, this finding could only have been obtained using a TOFMS.

A preliminary design of the probe to be used in the Impulse Facility is shown in Figure 1. A small gas sample will be collected by the orifice on the first skimmer. The gas sample that proceeds through each of the conical skimmers will form a molecular beam which will be directed into the TOFMS. A portion of the molecular beam will be ionized with a pulsed electron beam. The resulting ions will be deflected by 90°, accelerated, and allowed to drift in a meter long field-free tube to a multichannel plate detector. Current from the detector as a function of time will yield a measurement of the mass spectra. Ion masses will be separated in time during the drift in proportion to the square root of their mass. An entire spectrum with atomic mass units to 160 can be obtained in 20 μs, the period of the electron beam pulse. Considerations necessary to design a probe that will give a meaningful measurement of the free stream gas composition will be given in the next section.

The probe entrance orifice must be sealed until the initiation of the flow in the facility. The sealing cap will serve the dual purpose of providing a gas source for calibrating the mass spectrometer before each test.

Before initiation of the flow, the Impulse Facility test section will be pumped down to a pressure of approximately 1 torr. During this low pressure period the mass spectrometer will be calibrated. Before flow initiation, the sealing cap will be removed and stored out of the way of the flow.

TOFMS Probe Design Considerations

The following problems have been identified with accomplishing the sampling process as described in the previous section. They

are: a) chemical reactions continue to occur downstream of the skimmer since the quenching is not rapid enough to freeze the composition which exists at the skimmer entrance; b) mass separation will occur in the beam directed to the mass spectrometer due to the strong pressure and temperature gradients that are created by the presence of the probe; c) collisions between beam molecules and those that have collided with the inside surface of the skimmer cone will modify the properties of the molecular beam; d) the ionization efficiency in the detector will not be the same for all molecules (the ionization cross sections are different for each molecule); e) the ionizer/multichannel plate detector cannot not be operated at a pressure greater than 10^{-6} torr, which places a severe condition upon the pumping requirements for the various chambers downstream of the skimmers; f) the skimmers are very fragile and can be harmed by dust and other particles in the free stream flow; g) the probe must be small enough so that it will not interfere with starting the nozzle flow in the Impulse Facility. These issues will be briefly discussed.

A conical skimmer system will be used to extract a sample of the free stream gas and generate a molecular beam (Figure 1). The ideal sampling process would obtain an undisturbed sample of the free stream gas and freeze the beam composition. This requires having an attached shock wave on the first skimmer with rapid expansion downstream of the orifice. The first requirement places an upper limit on the outside cone angle while the second requires a much larger inside cone angle than the outside angle, i.e., these two requirements are at odds with each other. The cone angle was chosen to have an attached shock wave (30° half-angle) and computations were performed to estimate the effect of less-than-required downstream expansion on the beam properties.

Three skimmers will be used for the proposed system (Figure 1). The first skimmer orifice must be in continuum flow (Ref. 5), defined by having a Knudsen number less than 0.02. This Knudsen number is given by the ratio of the mean free path upstream of the probe to the orifice diameter. It was originally thought that the

there would be rapid expansion through the first skimmer, resulting in quenching of the upstream chemical composition. However, CFD computations, to be described in the next section, indicated that there will be much less expansion than anticipated. However, there is sufficient expansion to usefully reduce the mass flux through the second skimmer. DSMC computations showed that the second skimmer will be in the transition flow regime (Knudsen number about 0.22) while the third will be in free molecule flow (Knudsen number of 21). Thus the three skimmers will provide differential pumping to reduce the large mass flux in the free stream to a level that is acceptable to the TOFMS but will fail to quench the chemical composition of the sampled gas. CFD computations also predicted the 95% of the helium will be scattered out of the beam and will miss the ionized of the TOFMS.

Ionization of the beam molecules is required for their mass separation and detection since the free stream gas will consist almost exclusively of neutral species. The free stream will contain vibrationally excited molecules which have uncertain ionization cross sections (Ref. 3). This uncertainty will make calibration of the ionizer difficult because the calibrationcan only be performed with ground state molecules.

The current from the multichannel plate detector as a function of time will yield a measurement of the mass spectra. Individual ion species must be detected with adequate signal-to-noise ratio (S/N). This will require preventing any ions that occur in the free stream from entering the TOFMS by placing the first skimmer at a negative potential relative to the following skimmers (Ref. 3 and 4). Other precautions must also be taken, the most important being to keep the background gas density as low as possible in the detector chamber.

Some of the sampling errors described above can be accounted for by an adequate calibration procedure (Ref.3). Ionization cross sections applicable to the vibrational state of nitrogen must be used for data analysis. In addition, the procedure must account for the density variation across the molecular beam and the helium

2-7

separation that occurs in the sampling process.

It is recommended that a retractable cone sealing cap be placed over the skimmers and used as a thermal source calibration unit. However, this means of calibration cannot account for the vibrational excitation of the nitrogen or the helium separation that will occur during operation in the Impulse Facility.

Probe Computations

CFD was used to estimate the mass flow rates through the skimmers and to determine their influence on the free stream flow and on the composition of the gas stream which is captured by the skimmers (species separation effects). The first skimmer must operate in continuum flow (Ref.4) while the latter two skimmer are in rarefied flow. Thus a combination of continuum CFD and Direct Simulation Monte Carlo (DSMC) codes was required to examine the probe flow fields.

Previously the probe configuration used by Skinner (Ref. 4 and 5) was analyzed by the author using the NPARC2D code to compute the continuum part of the flow field and the Monte Carlo code DSMC2D-MP, developed by Dr. Timothy Bartel of Sandia National Laboratory, to compute the rarefied portion of the flow field (Ref. 6). This probe system is shown in Figure 2. The diameter of the first skimmer orifice was 2 mm and 0.7 mm for the latter two orifices. The first two skimmers were separated by 17.43 mm and the last two by 28.07 mm. The first two conical skimmers had outside and inside half-angles of 30° with a 20° chamfer and the third had an outside half-angle of 15° and an inside half-angle of 10.5°. The computations used the free stream test conditions for Run 21 in the Impulse Facility, performed 01-13-95 (Ref. 1).

Steady state inviscid NPARC2D computations were performed for the flow over the first two skimmers, without species diffusion. The flow between the first two skimmers was open to the free stream. The computations indicated that the shock was attached to the first skimmer. Properties were nearly constant and close to free stream values across the orifice of the first skimmer. The

2-8

flow appeared to expand little through the first orifice, although the density decreased downstream. The velocity was constant at the free stream value across the second orifice but the density was considerably less than the free stream value.

The computed mass flux through the first skimmer was close to the mass flux in the free stream, while the mass flux through the second skimmer was lower because of the diminished density before that skimmer.

The DSMC2D-MP code was run on the nCUBE machine at Sandia National laboratory, using the NPARC2D upstream solution as input with the third skimmer assumed to be connected to a 1000 l/s vacuum pump. One percent by mass of helium was added to the composition of the partially dissociated air which entered the first skimmer orifice. Chemical reactions were not included in the DSMC computations but species diffusion was included.

The DSMC2D-MP computations predicted velocity and temperature profiles across the second skimmer that were similar to those predicted by NPARC2D, but the expansion was predicted to be greater so the predicted mass flux through the second orifice was less by a factor of approximately 1/6. Much of this mass flux reduction can be attributed to the fact that the Knudsen number, based on the orifice diameter, was found to be 0.165 at the second skimmer, which places the second skimmer in the middle of the transition region. The continuum model would be expected to over-estimate the mass flux. The profiles of properties were uniform across the third skimmer orifice with the velocity reduced by 30% and the density reduced by a factor of 10^3, compared to the properties at the second skimmer. The Knudsen number at the third skimmer was 15.

The DSMC2D-MP computations predicted that most of the helium diffused and scattered off of the beam centerline. The mole fraction of helium in the final beam was reduced by a factor of 1/20 of the free stream value. Additional computations need to be performed to confirm this severe influence upon the beam composition.

Bird (Ref. 7) used DSMC to look at the free molecular flow through a conical skimmer. Two types of molecular collisions were found to interfered with the molecular beam passing through the skimmer orifice, namely, collisions between molecules that first collided with the outside conical surface and then with the entering molecules and collisions between molecules which had passed through the orifice but collided with the inside conical surface and then with the entering molecules. The first collisions can be eliminated by using a cone with an attached shock and by making the lip thickness very small compared to the orifice diameter. The second collision class was more insidious and could lead to a sudden decrease in the beam mass flux (the beam essentially broke down). By examining the flow through a cone with 25°/ 32° inside/outside total angles in a flow with Mach number 24 or greater, it was shown that the flow broke down when a modified Knudsen number was about 1 and the cone was essentially interference free when the modified Knudsen number was 2. A larger angle cone was not interference free until the modified Knudsen number was much greater, i.e., 8 for a 60° cone. The modified Knudsen number is defined as

$$\acute{Kn} = Kn \left(\frac{T_0}{T}\right)^{-\frac{2}{(\eta-1)}}$$

where η is the power of the intermolecular force law, T_0 is the stagnation temperature, and T is the static temperature. Small angle cones break down due to collisions between molecules which strike the inner cone surface and the beam while the large angle cones fail due to collisions between molecules which strike the outside of the skimmer and the entering molecules. A small angle skimmer will produce an unhindered beam at a smaller Knudsen number but its length must be short. The breakdown as the Knudsen number decreases is much more sudden for slender cones than for wider angle cones.

From the above analysis the effective Knudsen number of the third skimmer is 0.556 of the value computed by the DSMC2D-MP code, or 11.7, which is still large enough for unblocked flow into the skimmer. However, Bird's computations were not performed for a multispecies gas and the present DSMC calculations indicate that a is a large mass separation downstream of the third skimmer can be expected.

Additional computations have been initiated using the CFD code NASTD, developed by McDonnell Douglas Corporation, with the objective of determining the species separation in the continuum portion of the flow. NASTD is a very versatile code which can include finite rate chemistry and species diffusion. Whereas the NPARC2D code used the central difference scheme of Beam and Warming with Jameson artificial viscosity, NASTD allows the selection of numerous computational algorithms. The present computations were performed using the Roe first order spatial integration scheme (second order was always unstable for this problem). Viscous effects were included in the flow but not on the walls.

The original computations predicted mass fluxes into the probe vacuum chambers that could not be handled by any practical system. Therefore, it was decided to diminish the orifice diameters to 0.5 mm for each skimmer and the computations were performed for the probe system shown in Figure 2. Only the first two skimmers were included in the continuum computations. The computational domain was broken into twenty-one blocks, as shown in Figure 3. The skimmers were assumed to have 30° outside and 27° inside half angles and were separated by 18 mm. The first skimmer could communicate with the exterior flow but the second was connected to a closed chamber, which will be pumped by a Roots blower pump.

It was determined that the multispecies diffusional option was not fully implemented in NASTD so only single species computations are reported here; the multispecies option will be made available in the near future.

Computed Mach number contours downstream of the first skimmer are shown in Figure 4. NASTD predicts that the flow expands in the

form of a small jet downstream of the first skimmer. The maximum Mach number in the jet is 9.77. Since the Reynolds number is so small (965/mm) the jet does not re-expand after the diffuse Mach disk, but remains subsonic through the second skimmer. The maximum Mach number occurs at $x/D = 8.5$. Ideally this is where the second skimmer should be placed. It is concluded from the computations that the first skimmer orifice diameter should be increased to 2 mm, the value used for the original calculations.

Vacuum System Design

The previously computed values of mass and number fluxes through the skimmer orifices (Ref. 8) were used to size the vacuum chambers and pumps for the probe geometry shown in Figure 1, with the diameter of the second orifice decreased to 0.5 mm (from the previously assumed value of 0.7 mm) and the cone inside angle changed to 27°. Geometric relations were used to estimate the reduction in mass flux through the second skimmer due the its smaller diameter. These predictions will be checked against future NASTD computations for this geometry.

The volume flow rate, pumping speed, and pressure rise computations assumed that the gas flux into each chamber scattered many times with the chamber walls and thus acquired the chamber wall temperature of 300° K.

Initially the Impulse Facility test section will be pumped to a pressure of 133 Pa (1 torr). After the test section has been pumped to this level, the calibration cap will be removed from the second skimmer (it will have to surround the first two skimmers to seal the vacuum system). The pumps on the skimmer chambers were sized to maintain certain pressure levels when the calibration cap was removed and there was a direct connection of the vacuum systems to the test section.

Assuming that the first chamber must be maintained at a chamber pressure of 0.1 Pa (7.5×10^{-4} torr) while the probe is connected the Impulse Facility test section, a pumping speed of 52 ℓ/s will be required. This will be provided by a Roots blower type

pump, which would have to be placed outside of the test facility.

Maintaining the detector chamber at a pressure of 1.33×10^{-5} Pa (10^{-7} torr) under the same conditions requires of pump with a pumping speed of 173 ℓ/s. This could be accomplished by placing a 6 inch turbomolecular pump (blank-off pumping speed of 470 ℓ/s for nitrogen) in close proximity to the ionizer in the detector chamber.

The pumping system described above was much too small to handle the anticipated mass flux from the free stream. Those pumping requirements were 2000 ℓ/s for the first chamber and 13,000 ℓ/s for the second for steady state operation. The latter could never be provided because of the large conductance that would be required leading to the pump. Rather, the volumes of the chambers were sized to allow for a given maximum pressure rise during the assumed duration of the test (0.005 sec), assuming that the gas makes multiple collisions with the chamber walls during the test duration. The critical pressure is that in the detector chamber, which must not rise above 10^{-6} torr while the filament is hot or high potential is applied to the microchannel plate detector.

Using the techniques described above it was determined that the first chamber must have a volume of 20 to 40 liters and the detector chamber must have a volume of at least 60 liters. Assuming that the first chamber has an outer diameter of 8 inches, that the detector chamber has an outer diameter of 6 inches, to mate cleanly with the 6 inch turbomolecular pump, and that both chambers have 0.125 inch wall thicknesses, the total required length of the detector chamber would be 141 inches and the length of the surrounding chamber would be 106 inches. The detector chamber must have an L-shape, with the drift tube at right angles to the facility flow. Since the drift tube is approximately 42 inches long, the detector chamber would be 102 inches in the stream direction. The first chamber would have a volume of 38.6 ℓ and the detector chamber 60ℓ.

The dimensions of the chambers given above are much larger than desired. The only way to reduce them is to capture the

directed molecular beam in a LN_2/LHe cryogenic trap/pump. This seems feasible since the gas flow is very directed and this possibility is being explored.

Time-of-Flight Mass Spectrometer

The time-of-flight mass spectrometer (TOFMS) was manufactured by Comstock. Its operation is completely described in Ref. 8. Some additional points concerning the interpretation of the measured spectra and signal-to-noise issues will be discussed here.

The TOFMS accelerates ions of all masses through an equal potential difference giving all ions equal energy in the drift region. Thus mv^2 = constant and masses are separated by the square root of their mass with the lighter molecules arriving first. The drift time is obtained from a consideration of the kinematics of the individual ions. After creation, the ions are repelled by one-half of the repeller voltage (typically the repeller voltage is 150 volts). If all of the ions are assumed to be generated on the beam centerline, then the repeller acceleration region of length x_s is 0.00418 m. Next the ions proceed through a region of zero voltage of length x_b = 0.0024 m. They are then accelerated by the voltage at the entrance to the drift tube (typically 2500 volts) for a distance x_d = 0.009538 m, and then they drift in the drift tube of length x_D = 1.01049 m. Using the fact that the acceleration of an ion in a region of electric potential E is a = qE/m, and the electric field is related to the voltage of a region of length x by E = V/x, the arrival time of an ion of mass m is given by the solution of the following equations (Q = q/m):

$$x_s = QE_s t_s^2; \quad v_s = QE_s t_s^2$$

$$x_b = v_b t_b; \quad v_b = v_s$$

$$x_d = v_s t_d + \frac{1}{2} QE_d t_d^2$$

$$v_d = v_s + QE_d t_d$$

$$x_D = v_d t_D; \qquad v_D = V_d$$

$$t = t_s + t_b + t_d + t_D$$

The positive square root of the equation for t_d must be used. The equations predict measured times to with 8%, with most comparisons within 4%. However, comparisons with a regression line placed through measured points gives maximum errors in the range of 3% (Ref. 8).

The equations above assume that stationary ions are accelerated from a single location (on the beam centerline), gaining equal kinetic energy. In reality one must consider the effect of the initial velocity of the ions on the spread of the peaks (peak width) and the width of the beam in the ionizer since both contribute to the peak width.

If the initial ion velocity is small and the drift distance large, then the peak spread is related to the variation of the potential over the beam width, as given by

$$\frac{\Delta t}{t} = \frac{1}{2} \frac{\Delta V}{V}$$

The variation of the electrostatic potential is determined by the spatial spread of the region where the ionization occurs. This equation predicts that the width of the peaks increases with ion mass because the spread of the arrival time is proportional to the spread in the energy of the peaks. In addition, the spatial spread of the beam at ionization introduces a spread in the path length, ΔL, and the magnitude of the initial velocities of the molecules, Δv, relative to the drift velocity, v, will introduce a variation in the arrival time

$$\frac{\Delta t}{t} = \frac{\Delta v}{v}$$

These latter two effects are smaller than the first. The first effect is minimized by using spatial focusing (Ref. 9). However, for the Comstock instrument the measured peak widths will make id difficult to distinguish the NO peak from the N^2 peak. Note that the area under a peak is proportional to the number of ions of a particular mass which reach the detector.

Operation of TOFMS in Impulse Facility

A transient start-up period exists at the beginning of the test period. Initially the Impulse Facility test section pressure will be 1 torr and the probe chambers will be at stable pressures which will be determined by the gas influx from the test section, the chamber dimensions, and the attached vacuum pumps. After the first expansion wave passes the probe, there will be a transient that will last up to 500 μs (Ref. 2) during which the probe chambers will react to the changing test section conditions. Stable mass spectra can be obtained only after this initial transient.

The ionizing electron beam in the Comstock mass spectrometer will be pulsed at a 50 kHz rate, which will give a mass spectra every 20 μs. The output from the multichannel plate detector will be stored using a high speed computer board that can store 100 MSPS (million samples per second). The board is capable of storing 2000 spectra, which could be obtained in 40 ms. Each sample will contain 2000 data points, which will place about 10 data points across the helium signal. The computer board must be triggered with a upstream signal from a pressure transducer located close to the nozzle exit plane. A counter will be placed in the first bin at the beginning of each TOFMS scan so that the mass spectra sequence number can be identified for later analysis.

Post-test analysis of the spectra will include averaging groups of spectra to improve the signal-to-noise ratio and

integration under each peak to get a measure of the number of ions of each species that were detected by the TOFMS.

When the pressure in the detector chamber reaches 10^{-6} torr the filament current and high potential on the microchannel plate detector must be removed. This will require modifying the electrometer circuit on the ion gage controller to generate a cut-off signal that has a microsecond response.

Calibration

The measurement of the species concentration can be different from the actual concentration due to six effects: a) species separation can occur in the skimming process; b) chemical reactions can continue to occur downstream of the first skimmer orifice; c) the ionization cross section depends upon the species; d) ionization will dissociate some molecules giving an atomic plus a molecular signal; e) the transmission probability of the TOFMS is mass dependent; and f) scattering with background gas within the TOFMS is mass dependent (scattering cross sections depend upon scattering molecules). Where possible these influences must be determined by calibration.

Species separation has been discussed in the section on probe computations. It cannot be examined by static calibration.

The TOFMS generates ions by electron impact. Ions in the Comstock unit typically have an energy of 90 eV. Atoms will be singly ionized. Molecules can be singly ionized or dissociated, with some of the atoms ionized. The ionization cross sections depend on the electron energy, the species, and for molecules, the internal state of the molecule. Crane (Ref. 3) and Skinner (Ref. 2) used ionization energies that were much greater than the ionization threshold energy to make the ionization cross section independent of the internal state of the free stream molecules. This is not possible with the Comstock TOFMS since the greatest possible electron energy is 160 V. This uncertainty will remain for the Impulse Facility tests.

The ion transmission efficiency will vary between molecules

2-17

for an number of reasons. These include the different velocities that each species will possess in the ionization region, the different scattering cross sections with the background molecules in the drift tube, and a dependence of the ion beam divergence on the charge-to-mass ratio. In addition, transmission probabilities for atoms which exist in the molecular beam will be different from those produced by electron impact dissociation since the latter will have a larger initial random velocity spread. Transmission efficiency can be determined by static calibration at different drift tube pressures with fixed electrode potentials. The electrode potentials will be set during each calibration to maximize the helium peak. Transmission efficiency can be stabilized for a test only by capturing the beam with a cryogenically pumped trap.

The instrument calibration factors can vary from run the run and must be determined before each run. Other calibration factors, such as the probe-induced gas separation, depend upon the test conditions. Present CFD computations indicate a strong Reynolds number influence on the flow through the first skimmer and the helium separation. Separation is also strongly related to the skimmer geometry. Separation has been investigated for the probe system shown in Figure 2 but the investigation of the proposed skimmer system shown in Figure 1 remains to be performed.

References

1.   Blanks, J. R., "Initial calibration of the AEDC Impulse Facility," AEDC-TR-95-36, August 1996.

2.   Skinner, K. A., "Mass spectrometry in shock tunnel experiments of hypersonic combustion," Ph.D.Dissertation,The University of Queensland, Australia, March 1994.

3.   Crane, K. C. A., "Mass-spectrometric analysis of hypersonic flows," J. Phys. D: Appl. Phys. Vol. 10, pp. 679-695, 1977.

4.   Skinner, K. A., "Species measurements using mass spectrometry in high Mach number flows," 11th Australasian Fluid

temperature-gas collision processes," Phys. Fluids, Vol. 4, No. 9, pp. 1172-1176, 1961.

6. Collins, F. G., "Time-of-flight mass spectrometer sampling system for AEDC Impulse Facility," AFOSR Summer Faculty Research Program Final Report, August, 1995.

7. Bird, G. A., "Transition regime behavior of supersonic beam skimmers," Phys. of Fluid, Vol. 19, No. 10, pp. 1486-1491, 1976.

8. Collins, F. G., "TOF mass spectrometer sampling system for AEDC Impulse Facility," Report to AEDC, September, 1995.
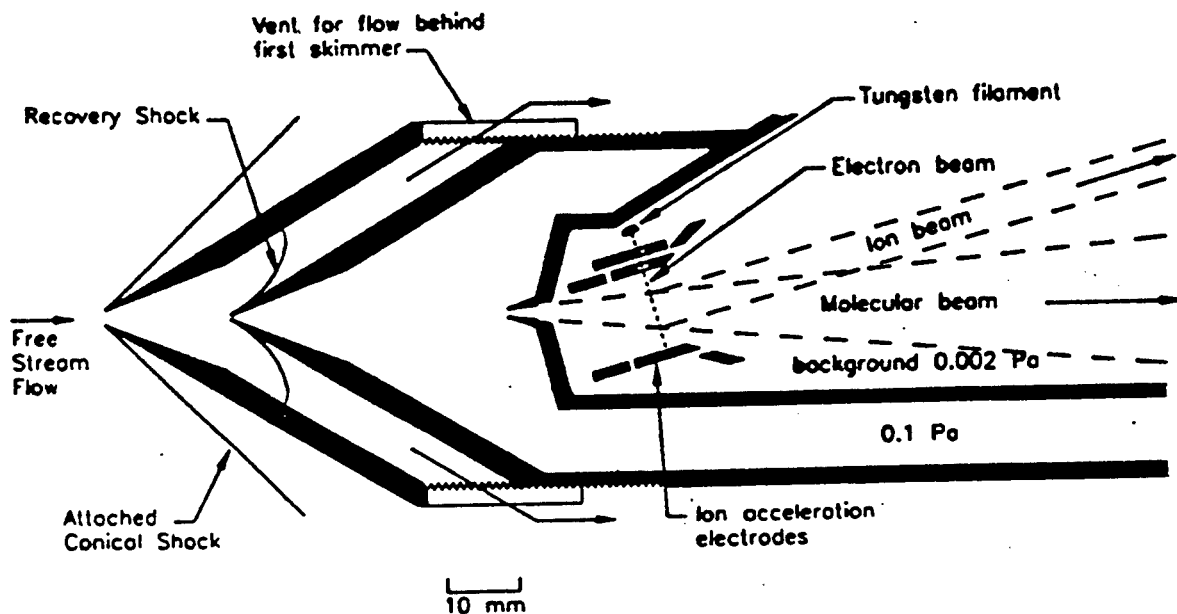
Figure 1. Proposed probe design without cryogenic pumping.
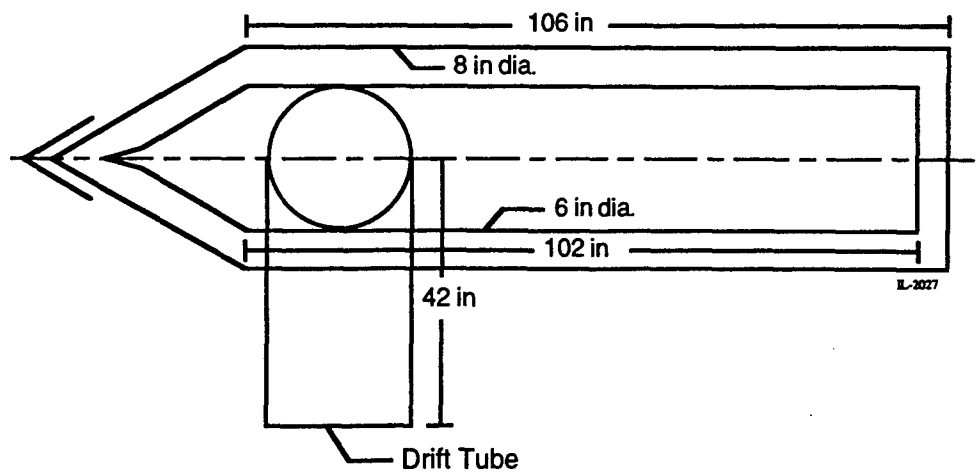
Figure 2.   Skimmer arrangement and ionizer used by Skinner (Ref.2 and 4).
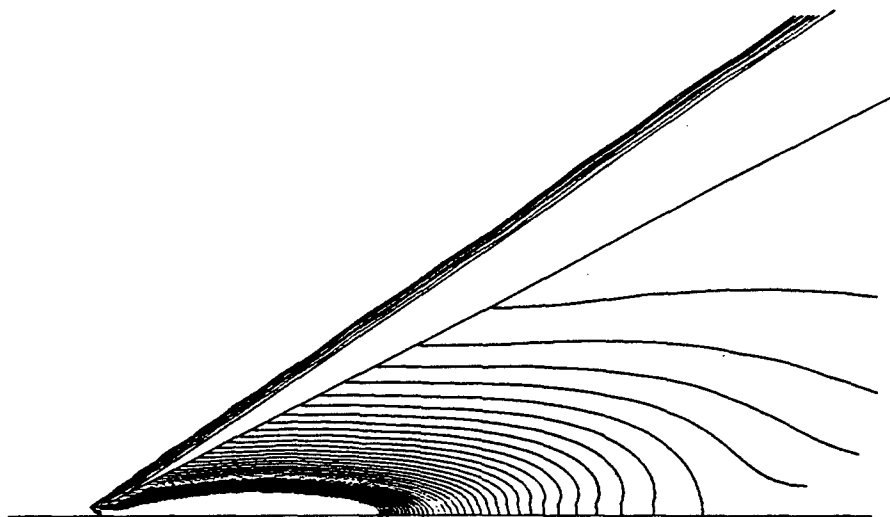
Figure 3.   Constant Mach number contours downstream of the first skimmer - NASTD computations.

Associate did not participate in the program.

VELOCITY FIELD MEASUREMENTS
USING FILTERED-RAYLEIGH SCATTERING

Kevin M. Lyons
Assistant Professor
Department of Mechanical and Aerospace Engineering

North Carolina State University
Box 7910
Raleigh, NC 27695-7910

VELOCITY FIELD MEASUREMENTS
USING FILTERED-RAYLEIGH SCATTERING

Kevin M. Lyons
Assistant Professor
Department of Mechanical and Aerospace Engineering
North Carolina State University
Box 7910
Raleigh, NC 27695-7910

Abstract

Investigations into Filtered Rayleigh Scattering for large scale velocity field measurements in AEDC high speed wind tunnels were initiated during the summer 1997 program. The injection seeded Nd:YAG laser was optimimized for measurements, refining the alignment and cooling system of the laser and assessing its temperature stability, tunability, mode hopping characteristics and other pertainent optical requirements. An iodine cell used in the collection optics was also constructed and tested in the laboratory. The main accomplishments of the research were assessment of the feasability and the difficulties in making measurements in large scale wind tunnels. Mainly, this was done to investigate and hopefully any problems encountered in a laboratory setting before a complete full scale run was attempted. The results indicate that filtered-Rayleigh scattering is a promising technique for tunnel applications and more work will be done in the coming year in making tunnel measurements feasable as well as improving measurement accuracy.

VELOCITY FIELD MEASUREMENTS
USING FILTERED-RAYLEIGH SCATTERING

Kevin M. Lyons

Background

In the history of experimental fluid mechanics and combustion, a wide variety of techniques have been used to study both large-scale and small-scale features of reacting and non-reacting flows. These techniques can be divided into categories based on the nature of the method employed. Measurements of scalars such as temperature have been made using thermocouples by physically probing the environment of interest. Velocity measurements have been made by hot wire anemometers, again by perturbing the flow with a probe. All of the methods classified as techniques that physically probe a medium have drawbacks related to the intrusive nature of the measurement. In addition, data obtained from physical probes need to be corrected in order to be made quantitative; for example, thermocouples need to be corrected for conduction and radiation effects. Also, as the measurement environment gets more hostile, it becomes more and more difficult to build probes that are able to withstand high temperatures and pressures and to remain accurate in the presence of particulates.

A category of techniques that has seen much fruitful use in fluid mechanics and combustion is that of optical diagnostics. Shadowgraphy and Schlieren photography have been applied widely in fluid mechanics, particularly in the area of compressible flow. These methods are sensitive to changes in refractive index and hence density, and have the advantage that they allow two-dimensional measurements to be made (Merzkirch 1987). These techniques have the draw back that they are line-of sight, i.e. the detector records

images integrated through the measurement volume. With two spatial dimensions well resolved and one poorly resolved, applications of these techniques are limited.

Probably the best known and most widely applied optical technique used to measure velocity in flows is laser Doppler velocimetry (LDV). The technique obtains its name from preliminary experiments where the Doppler shift of light incident on moving particles in a flow was detected; the shift is indicative of one component of the flow velocity in the resolution volume. Today's velocimeters, while referred to as laser "Doppler" anemometers, do not in general employ the same principles. Using these instruments, intensity gratings are created in the flowfield by the intersection of two laser beams. As a particle travels through the bright fringes in the grating, a time trace of the scattered light is detected. Since the spacing in the grating is known from the wavelength of the laser and the orientation of the beams, one component of the velocity can be found. Instruments are commercially available that use these principles for measuring all three components of the velocity. These instruments have frequently been used to determine velocity statistics at a point. The advantages of the current LDV instrumentation is the high data rate (> 10 kHz) and ease of optical alignment. The major drawback is the single point nature of the technique, being unable to instantaneously characterize turbulent flow structures and the need for particle seeding. Heitor et al. 1993 surveys recent advances in laser Doppler velocimetry techniques and instrumentation.

A variety of molecular tagging techniques have been used to determine velocity fields in gas flows using laser-induced fluorescence and phosphorescence of molecules in the flow. Velocity field visualizations in subsonic gas flows have been obtained by laser-induced phosphoresce of biacetyl molecules. This technique is restricted to applications in low speed flows and due to the uncertainty in phosphorescence lifetime of the biacetyl molecule, the accuracy of the velocity measurement is poor. Velocity profiles in a sonic jet have been obtained by Raman excitation and laser-induced electronic fluorescence of oxygen molecules (RELIEF) (Miles and Lempert 1997). These studies have resulted in

velocity measurements of one component along a line. Another class of velocity measurement techniques, better termed spectroscopic techniques, rely on the Doppler shifted absorption curve of molecules such as iodine and sodium. These and other similar techniques have met with varying degrees of success. The shortcoming in all of these approaches is the difficulty in measuring more than one component of the velocity. Additionally, most of these techniques are not suitable for application in reacting flows. However, their major strength is that no particulates need to be added to the flow as markers, as is required in LDV or PIV. One of these techniques will be discussed in the next section.

Seeding flows with small particles and subsequently tracking their motion is an approach taken by many to both qualitatively and quantitatively investigate fluid motion. Van Dyke's *An Album of Fluid Motion* has many examples of light scattering from smoke particles in air to visualize flowfields. Particle streak velocimetry has been used to investigate velocity fields by recording scattered light from moving particles interrogated by light sources of long enough duration to result in streaks rather than particle images. This technique allows the determination of the velocity field by knowledge of the length of the streak and the duration of the laser pulse. However, its general applicability and accuracy are questionable since it depends on measurement of the length of individual particle streaks by visual or computer-aided inspection.

In recent years studies of velocity fields in various flows have been made using particle image velocimetry (PIV). By using strobed laser light to illuminate a plane within a particle seeded flow, multi-exposed images are recorded, which contain information about the displacement of the particles. Processing this data yields the in-plane velocities within the illumination sheet. The time interval between pulses is adjusted according to flow velocity in order to optimize the measurement of the particle displacement. Though many PIV studies to date examine laminar flow fields, this technique does have the ability to yield quantitative information with high spatial resolution about two-dimensional velocity fields

in turbulent flows, and is being presently applied to such flows. Investigations of steady and turbulent flows have been made including uncertainty estimates in turbulent flows. Measurements of velocity fields and vorticity in a plane have been made in a motored internal combustion engine. This study is encouraging in that it demonstrates application of PIV to practical devices.

Techniques Suitable for Large Scale Tunnel Facilities

Although the techniques discussed above have produced many important results in laboratory and small scale device diagnostics, most of them are unsuitable for large scale wind tunnel/model testing applications. Firstly, most require the detector to be in close proximity to the laser sheet, which is difficult to accomplish in most tunnel/model geometries. Also, the addition of particulates to the flow, as required in LDV and PIV, limits the number of tunnels that can be probed with such techniques. Filtered Rayleigh Scattering is a promising technique that can be used to make quantitative measurements of velocity by using the Doppler shifted scattered light from particulates (and/or molecules in the flow). In addition, in order to discriminate the scattered light at the Doppler shifted wavelength from the background laser light, the absorption spectra of an iodine vapor cell is used due to its sharp-cut properties and its shape in the vicinity of the operational wavelength of the Nd:YAG laser. The theory and basic details are contained in many papers and monographs, among them Elliott and Samimy (1996) and Forkey (1996).

Achievements During Summer 1997

A)   An injection seeded Continuum Nd:YAG was optimized for the proposed measurements. This involved precise alignment of the diode laser used to seed the main YAG laser in order to reduce the laser linewidth by an order of magnitude.

B)   An iodine cell to be used in the Filtered Rayleigh experiments was constructed and tested. The cell was made of a 4 inch diameter Pyrex tube 12 inches in length with imaging

quality flat glass ends that were fused to the Pyrex. A sidearm was installed in the Pyrex tube through which 5 grams of iodine was inserted. The cell was then evacuated and the sidearm sealed. Heat tape was then applied to the cell, as well as to the sidearm independently, and each of the regions were controlled separately. The sidearm temperature was below the cell temperature (nominally 30 C compared to 80 C for the cell) and effectively controlled the number density of iodine in the vapor. Once the cell was temperature stabilized, absorption measurements were using an unintensified CCD detector to detect scattered light from a jet interrogated by a laser sheet from the seeded laser. Reproductions of numerically predicted iodine absorption were made and compared to previous results (Forkey 1996). In addition, relative velocity measurements were attempted in a high speed jet seeded with a fogging agent. Although the results were qualitative, many of the problems associated with making quantitative measurements were investigated in a preliminary manner. The future will see these issues in A) and B) reinvestigated in the research follow-on program, as well as make preparations for the full scale tunnel test anticipated to be in late 1998.

C) Other Efforts

On June 19, 1997, Fred Heltsley (AEDC) and Kevin Lyons (AFOSR Summer Faculty) met with Campbell Carter (ISSI), Jeff Dunbar (WPAFB) and Stephen Arnette (University of Dayton) at WPAFB to discuss the filtered-Rayleigh scattering technique for velocity field measurements in high speed flows. Some of the issues discussed concerned experimental setup, data processing, background suppression and system calibration. Based on these conversations, along with the current literature on flowfield assessment (Arnette et al. 1995, McKenzie 1996, Miles and Lempert 1997), the filtered-Rayleigh technique seems promising as a technique for assessment of velocity fields in large scale facilities such as 16T at AEDC. Although currently the filtered Rayleigh technique requires particle seeding to obtain adequate signal levels, it is anticipated that the technique will be

applicable in the future in unseeded flows with the advent of higher energy pulse lasers (Miles and Lempert 1997).

Future Work

The future work in 1998 involves a number of engineering/ image acquisition issues, including:

A) Testing the system by examining larger fields of view.

B) Taking steps to environmentally seal cameras so one can operate in ambient tunnel conditions.

C) Working on software to facilitate data processing and analysis.

Theses issues are to be addressed during 1998 during either the Summer Research Program, at the faculty member's university in conjunction with a student or at both times. The intentions are that if enough progress can be made by 8/98, preparations for a full scale run should begin in early 1998 and a run attempted possibly in Fall 1998.

References

Arnette, S. A., Samimy, M. and Elliott, G. S., AIAA Jl. 33, 430 (1995)

Eckbreth, A., *Laser Diagnostics for Combustion Temperature and Species,* (Abacus Press, Cambridge, MA 1988).

Elliott, G. S. and Samimy, M., AIAA96-0304, Reno (1996).

Forkey, J. N., Ph.D. dissertation #2067-T, Princeton University (1996).

Laufer, G., *Introduction to Optics and Lasers in Engineering*, (Cambridge 1996)

McKenzie, R. L., Applied Optics 35, 948 (1996)

Merzkirch, W., *Flow Visualization*, 2nd ed. (Academic Press, Orlando 1987).

Miles, R. B. and Lempert, W. R., Annual Review of Fluid Mechanics, 26, 285 (1997)

Watson, K., Lyons, K. M., Donbar, J. and Carter, C. D., to be submitted to Comb. Flame (1997)

Gerald Micklow's report was not available at the time of publication.

EXTENSION AND INSTALLATION OF THE MODEL-INTEGRATED
REAL-TIME IMAGING SYSTEM (MIRTIS)

Michael S. Moore
Research Associate
Department of Electrical and Computer Engineering

Vanderbilt University
400 24th Ave. S.
Nashville, TN 37235-1824

EXTENSION AND INSTALLATION OF THE MODEL–INTEGRATED REAL–TIME IMAGING SYSTEM (MIRTIS)

Michael S. Moore
Research Associate
Department of Electrical Engineering
Vanderbilt University

Abstract

MIRTIS (Model Integrated Real-Time Imaging System) has been developed during the past 5 years through a joint effort between Vanderbilt University and Arnold Engineering and Development Center (AEDC), with support from the Air–Force Office of Scientific Research (AFOSR). MIRTIS is a real–time image processing development environment which employs Model–Integrated Program Synthesis (MIPS) techniques for generating real–time image processing applications to be executed on a parallel hardware architecture. MIRTIS is capable of creating very high performance parallel implementations of a large class of image processing computations from high–level graphic specifications (models) of the system. These include models of the computations to be performed, the available resource, and the timing constraints of the application. From these graphical system models the MIRTIS *model interpreter* automatically parallelizes the computations using a two–level decomposition technique, scales the parallelism, and allocates the scaled decomposition to the available parallel hardware architecture (a network of Texas Instruments TMS320C40 DSPs) such that the resulting implementation will meet the specified timing constraints.

During the 1997 RDL/AFOSF Summer Faculty Research Program, the capabilities of the MIRTIS system were extended to enable MIRTIS to be used as an off–line compute server for reducing and enhancing large archived image data sets, as well as the real–time video applications for which it was originally designed. This required extensive modifications to several MIRTIS components (see figure 1), including the host interface programs, the Pipeline Cut–Through (PCT) run–time system (the kernel that runs on the C40s) and the Multigraph–Architecture (MGA) components of the system (the graphical model editor and model interpreter). The new system was installed, and a manual set was written so that AEDC personnel can use and extend the algorithmic capabilities of the system.

1 Introduction

This section provides background information, including the motivation for the development of MIRTIS, the MIRTIS architecture, and short descriptions of each of the MIRTIS components.

1.1 Motivation

Non–dedicated image processing applications users usually have to sacrifice algorithm implementation flexibility for real–time performance. Most existing off–the–shelf real–time systems use specialized hardware architectures to perform specific algorithms in real–time (eg. convolution). A draw–back of specialized hardware is that it cannot always be re–programmed with new or non-standard algorithms. Some imaging facilities, such as the one at Arnold Engineering Development Center (AEDC), require a system which can be rapidly programmed, configured, and scaled to solve a wide variety of problems.

Through the search for a replacement for AEDC's current real–time imaging systems, it became evident that the real–time image processing equipment available "Commercial Off–The–Shelf" (COTS) was not the most cost effective solution to AEDC's test support needs. Since the support requirements constantly change, they require a system which is rapidly programmable, configurable, and scalable. To lessen the hardware costs, the system needed to be modular and require as little non–COTS hardware as possible. To minimize training and programming costs, the system needed to include a very high level programming environment providing simple and transparent exploitation of the underlying parallel hardware architecture.

1.2 The MIRTIS Project

MIRTIS (Model–Integrated Real–Time Imaging System) is a real–time image processing environment and architecture developed through a 5–year joint effort between Vanderbilt University and AEDC. The goal of the on–going project is to build a machine and programming environment which not only meets AEDC's needs, but also solves as general a class of problems in as many imaging application domains as possible. In addition to AEDC's real–time video processing applications, the system should be applicable to other problem domains, such as low–level vision for robotics, autonomous navigation, tracking, remote sensing, etc. The design goals of MIRTIS are:

- **Scalable Real–Time Performance** to meet the requirements of a broad class of applications. The system must be easily scaled up or down for a particular application.

- **Programmability** so engineers can rapid prototype experimental/custom algorithms and create real–time computational data flows consisting of both these and standard algorithms.

- **Flexibility** so that the system can be rapidly changed to meet the requirements of new applications, including the software and hardware configurations, and the real-time constraints.

- **High Level Parallel Programming Interface** so the programmer can write parallel programs without dealing with parallel issues. Programming style should be independent of the underlying parallel hardware architecture.

- **High Level Tools** to enable rapid reconfiguration and scaling, and to make the system easy to use.

- **Dynamic Interactivity** to enable algorithmic parameters to be adjusted "on the fly", while the system is running. Users should be able to experiment with and fine tune the algorithms to meet the application requirements.

In order to meet these goals, MIRTIS employs Model-Integrated Program Synthesis (MIPS), a method of managing complexity in large engineering systems.[1] Using this technique, MIRTIS automatically "synthesizes" data parallel image processing applications from high level graphical specifications. It generates configurations for an execution environment running on a parallel network of Texas Instruments TMS320C40 DSPs (C40s). Real-Time execution is achieved by scaling the data parallelism to the appropriate level to meeting the timing specifications. The inter-processor communication is performed while causing virtually no overhead via a communications concept called Pipeline Cut-Through (PCT). Since the parallel system is automatically synthesized, the user is insulated from the complexities involved in programming and integrating a parallel system.

1.3 The MIRTIS Architecture

The MIRTIS architecture, shown in figure 1, follows the MGA framework,[1] and consists of the following: (1) the IPDL graphical model building environment, (2) the model database, (3) the MIRTIS model interpreter, (4) the IPLib image processing application library, (5) the PCT-C40 run-time system, (6) the MIRTIS host interface server, and (7) a network of C40s. The most important aspects of these elements are discussed next.

1.4 The IPDL Modeling Paradigm

The MIRTIS modeling paradigm, called Image Processing Description Language (IPDL), was designed specifically for real-time image processing. The concepts were developed by extracting the set of information required to support a automatic data flow decomposition, parallelization, and scaling, and allocation strategy, which is described in detail in 2 3 4. This section briefly describes the IPDL modeling paradigm, putting emphasis on the novel concepts.

Figure 1. The MIRTIS architecture

As is shown in table 1, IPDL contains three types of models, *Signal Flow*, *Hardware*, and *Con-straints*, which represent the data flow computation to be performed, the hardware resources available for the solution, and the timing constraints required by the solution, respectively. The combination of a Signal Flow Application model, a Hardware Network model, and a Constraints RealTimeCon-straints model together form a the specifications for a real-time image processing *system*.

1.4.1 Signal Flow Models

Signal flow models specify the image processing computations to be performed. The two types of signal flow models are *applications*, and *algorithms*. Applications are data flow graphs made up of algorithm models, each which declares pertinent information about an algorithm in the image processing library. Figures 2 and 3 show examples application and algorithm models being edited in the *xvpe.ipdl* graphical model editor. For brevity, the following sections outline only the most important parts of the application and algorithm models. For a more detailed description, along with more examples, see 3.

Algorithm data dependency specification: When implementing data parallel decomposition, the interpreter must be able to determine how the input data is to be split and allocated to the worker processors so it can (1) configure the communication system correctly, and (2) accurately model the communication overheads.

For modeling the data dependency behavior of the algorithms, a mathematical dependency spec-

The IPDL Paradigm		
Paradigms	Models	Model Aspects
SignalFlow	Algorithm	Structure Constraints DataDependency ParameterInterface
	Application	Structure
Hardware	Node	Hardware
	HostNode	Hardware
	Network	Hardware
Constraints	RealTimeConstraints	Goals

Table 1. Concepts in the IPDL modeling paradigm

ification format was devised that is unique to this development. The idea is to show an algebraic relationship between an output pixel location and a region in an input image sequence. The data dependency specification is contained in an attribute of each algorithm model, the format of which is given by:

$$Out[rvar, cvar, tvar] \quad " \longleftarrow " \quad In[rowrange, colrange, timerange]$$

$$range = begin() [\text{ } "..." \text{ } end()] \quad | \quad " : "$$

where *Out* and *In* are names of two of the algorithm's image signals, and *rowrange*, *colrange*, and *timerange*, following the *range* format. In the range specification, *begin()* and *end()* are algebraic formulas specifying the first and last indices of the range, and a ":" specifies the entire gamut of valid indices.

A data dependency parser was implemented which parses and evaluates the data dependency specifications. During interpretation, the data dependency specifications are parsed to determine the data access patterns for each algorithm in the data flow. For a particular decomposition method, this information is used in characterizing communications overheads in the performance models and
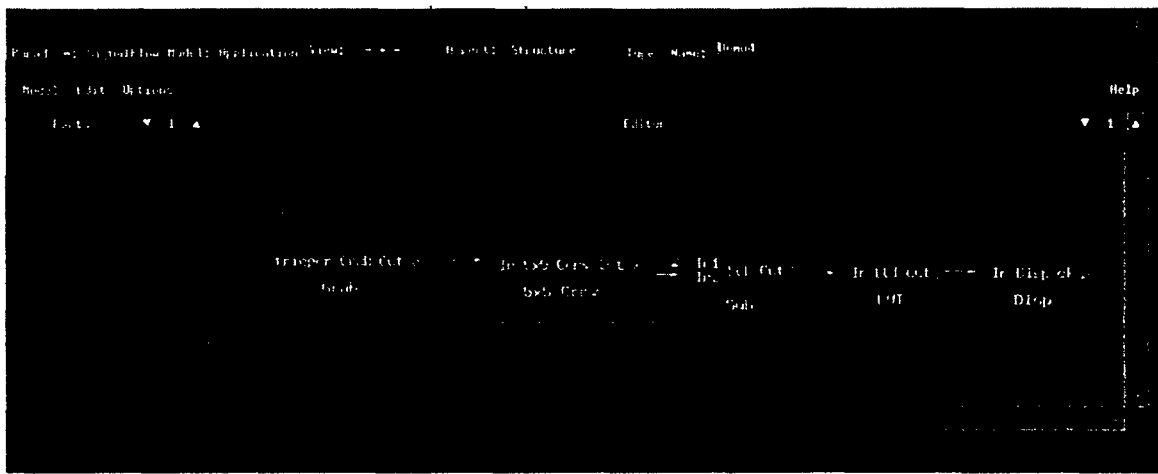
Figure 2. An Example Application Model
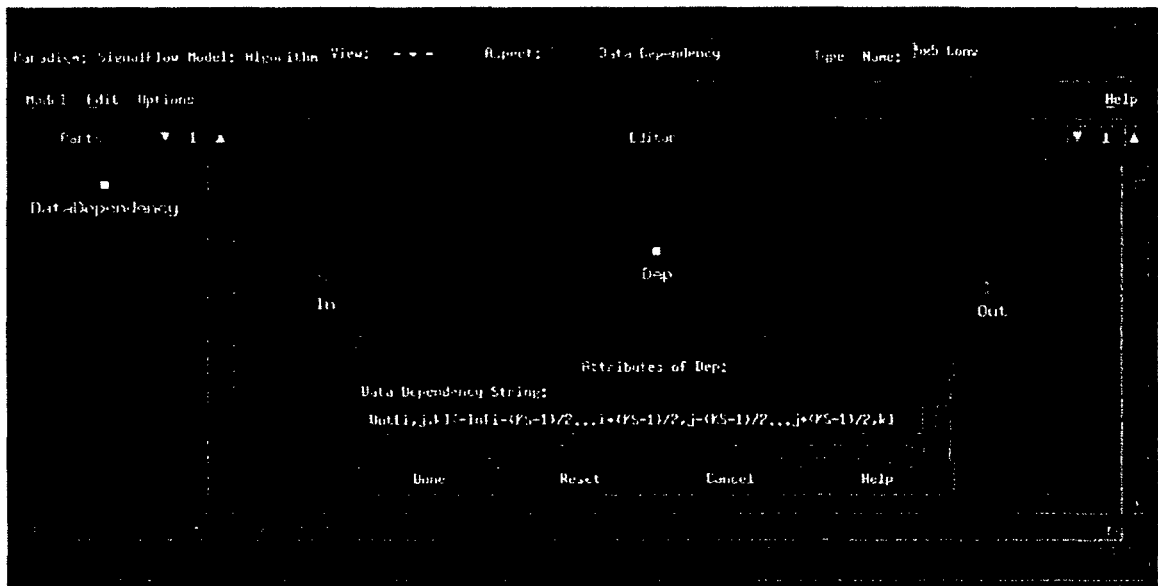
Figure 3. 5x5 Convolution Model Showing Data Dependency Aspect

in configuring the PCT communications engines (see 3).

Algorithm benchmarks: The approach taken in modeling each algorithm's execution time as a function of data size was to rely upon empirically gathered benchmarks. Using measured execution times instead of other approaches is both more accurate and straight-forward.

Each algorithm model contains a set of *benchmark* parts, each having an attribute specifying the dimensions of the algorithm's input images for that measurement. By providing execution time benchmarks for various data sizes (on a single node of the target architecture), an execution time versus data size curve can be constructed. The method of estimating the execution time of an

6-7

algorithm for a particular data size is to use linear interpolation on the bench-marked data sizes, including an implicit benchmark of zero seconds for zero data size. This unique benchmark–based interpolative approach to approximating execution time has proven to provide very accurate results in testing.

1.4.2 Hardware Models

The types of hardware models are *Node models*, *HostNode models*, and *Network models*. Network models are inter–connected hierarchies containing nodes, hostnodes, and other networks. Node models represent the processing nodes which perform the image processing computations on the data stream. In the case of the prototype, these are C40s. Each node model has attributes describing the node type and configuration, the ports, and any special resources, such as frame digitization hardware. HostNode models represent the PCs or workstations, which provide services such as network loading and disk I/O. They contain various attributes and parts, notably host interface card parts, which are boards in the host bus that provide communication links to the C40 network through which loading and run–time communication occur.

1.4.3 Constraints Models

Constraints models contain explicit declarations of the target latency and throughput required for an application. Throughput models have a numerical attribute specifying frame rate in $\frac{frames}{sec}$, and latency models have attributes specifying latency in *frames*. Both throughput and latency models have attributes specifying whether it is a *hard* or *soft* constraint. As will be seen, this attribute is used in the interpretation procedure in the case that the constraints cannot be met exactly. It specifies whether that constraint can be relaxed to allow a *best effort* implementation on the available hardware.

1.5   The PCT–C40 Run–Time System

A real–time image processing kernel called PCT–C40 provides run–time support for data parallel execution of image processing data flows on pipeline–connected C40 networks. The kernel runs on each C40 node and performs the scheduling, communication, and synchronization necessary for data parallel execution.

The scheduler on each node runs a Periodic Admissible Sequential Schedule (PASS)[5] which implements the computation block (sub–graph of the application data flow) local to that node. The kernel configures and starts the PCT communication engine, which, in cooperation with the neighboring nodes, distributes the input data appropriately across the processors and combines the local results to form the output data. Since the computation and communication schedules are

static, the scheduler introduces minimal run-time overhead. This also has the effect of simplifying the kernel by pushing the work of generating computation and communication schedules into the model interpretation process.

1.5.1 Pipeline Cut-Through Overview

Pipeline Cut-Through (PCT) is a communication technique which allows synchronous data flows parallelized with the spatial or temporal data parallel constructs to be mapped to a group of C40s connected in a pipeline (a PCT group). PCT achieves this by routing all communications, including the distribution (splitting) of input data and collection (merging) the partial results, along the C40 pipeline. PCT also provides coordination between the communication and computation processes. Since PCT implements the parallel facilities automatically, the data parallelism is absolutely transparent to the programmer.

Each node of a PCT group performs the same computations on a different section of the image data. The incoming stream is split and spread across the memory banks of the group nodes, and after the local data flow computation has produced the partial results, they are combined (merged) to form the output data stream. As well as splitting and merging the data stream, the communication engine also supports the sharing of pixels from the input sequences between two or more nodes working together to implement a data parallel computation.

The implementation is achieved by programming the Direct Memory Access co-processors (DMAs) of the C40 in such a way that they execute the correctly timed sequences of *receive & send, forward*, and *insert* that cause the input data stream to be distributed appropriately across the memories of the group, and the output stream to be assembled.

For more information about PCT, see Refs. 6,2,3.

1.5.2 Related Techniques

The name pipeline cut-through was chosen due that fact that the technique routes all communication along the hardware pipeline by "cutting through" intermediate processors. The technique is similar to two general message routing techniques, *circuit switching* and *wormhole routing*,[7] in that messages between non-neighboring nodes are routed *through* intermediate nodes with no local buffering. During transfers, messages pass through intermediate nodes without entering the local memory or interrupting the computation. The largest difference between pipeline cut-through and these techniques is that they are designed for routing asynchronous messages in more general inter-connection topologies. The messages can vary in length, and can be sent at any time. In pipeline cut-through, the message traffic patterns are known prior to run-time, and transfers are routed along the pipeline by *coordinating* the communication engines to execute pre-determined

transfer sequences. PCT is not a general message passing routing algorithm, and is unique to this development.

1.6 The IPLib Image Processing Algorithm Library

The actual image processing functionality is provided by the IPLib Imaging Algorithm Library (see figure 1), a library of image processing algorithms written in C and compiled with the standard Texas Instruments C40 compiler. This library is the simplest component of the system, since the image processing functions can be written as if they were to be used in a normal uni–processor system.[1] It was decided to take this approach instead of generating the image processing specific code directly from the most so that image processing libraries optimized for the target architecture could be re–used, which saves time and results in better resource utilization.

1.7 The MIRTIS Interpreter

The model interpreter is the heart of any MGA system, and requires the largest implementation effort. The job of the MIRTIS model interpreter is to translate the IPDL models into a scaled decomposition of the data flow, map the decomposition to the underlying hardware architecture, and construct network communication and computation schedules which configure the real–time image processing kernel and realize the parallel real–time data flow. Referring to figure 1, the products of the interpretation are (1) PCT network configuration files, and (2) host interface configuration files. These files are used in (1) booting the network, (2) configuring the network communication engines and schedulers, and (3) configuring the host interface utility (dynamic parameter adjustment interface and boot loader).

1.8 Performance

The first MIRTIS prototype was capable of performing data flows of local image processing algorithms on 512x480, $8\frac{bits}{pixel}$ (256 grey levels) frames at video rate (30 frames per second), given that enough C40s are available in the network. The digitization resolution (width and height of the digitized frames) is adjustable. However, $7\frac{Mbytes}{sec}$ is the highest data rate yet achieved due to the limitations of the C40 communication link (using ribbon cables reduces the C40 link bandwidth significantly). By doubly linking the hardware pipeline, though, an increase to $14\frac{Mbytes}{sec}$ is expected. A prototype MIRTIS system containing 51 nodes has been bench-marked at $520\frac{Mops}{sec}$ sustained (using wall-clock time and counting only useful computations) while performing a complex edge detection on live video. Systems as small as 4–6 processors have been used for simpler applications.

---

[1]The functions must follow a loose format and set of rules. For instance, there is a programming API through which must be used in obtaining the input and output buffers.[6]

2 Extensions of the MIRTIS Capabilities

Forthcoming applications at AEDC will require not only real–time image processing capabilities, but also high speed processing of very large archived image data sets. There are currently no COTS systems available that are capable of handling the data sizes and computation rates required by these applications. Although MIRTIS was designed for real–time video processing (capturing frames from video with an A/D frame grabber and processing the images at frame rate), it is also capable of processing images stored in files. It was seen that MIRTIS could eventually provide the utility to support the forthcoming applications requiring high speed processing of large data sets by making several extensions to the existing run–time system and adding more flexible file I/O support on the host interface server. Several modifications to the model interpreter were also required. These are discussed in the following sections.

2.1 PCT Run–Time System Modifications

The following modification were made to the PCT run–time system, requiring a large portion of the kernel to be re–written.

2.1.1 Image Data Types and Sizes

The first version of the PCT run–time kernel worked only on packed byte images (8 bit unsigned pixels packed into 32 bit C40 words) and required that each connection in the data flow have the same data size (the size of the result images were the same as the size of the source images). This was not restrictive in the real–time imaging domain, since most applications operate on such data streams, and the output data size is the same as the input. However, considering the widely varying data types and formats used to store images in files, this would be un-necessarily restrictive when processing archived data sets. Also, many algorithms consume and produce images of different sizes.

The image data model used by the PCT run–time system was expanded to handle the data types shown in table 2. The image data type and image size of the connections need not be the same throughout the computational data flow. For example, an algorithm may consume a 512x480 PACKED_UINT8 image and produce a 200x200 FLOAT image. This capability will add a great amount of flexibility to the applications that MIRTIS can support.

One note about the IPLib algorithms: As of now, the IPLib includes only algorithms which consume/produce PACKED_UINT8 images of the same size. Although the IPLib can now contain algorithms which operate on image of different sizes, containing the image data types in table 2, the benefits will not be seen until such algorithms have been developed and integrated into the IPLib. The existing algorithm must also be modified to query the input image data types to ensure that

Table 2. Image Pixel Data Types Supported By PCT

Type Identifier	data type	$\frac{bits}{pixel}$	storage in 32-bit word
PACKED_UINT8:	unsigned int	8	4*8 packed
PACKED_INT8:	signed int	8	4*8 packed
PACKED_UINT16:	unsigned int	16	2*16 packed
PACKED_INT16:	signed int	16	2*16 packed
PACKED_RGB24:	unsigned int	24	$8\frac{bit}{red,green,blue}$ padded
INT16:	signed int	16	1*16 padded
UINT16:	unsigned int	16	1*16 padded
INT32:	signed int	32	1*32
UINT32:	unsigned int	32	1*32
FLOAT:	float	32	C40 float format

they are compatible with the algorithm. The run–time system defaults all image connections to PACKED_UINT8 images (see the MIRTIS IPLib Programmer's Manual).

In order to support the above data types and data flows with varying image sizes, the PCT run–time kernel had to be vastly revamped. The data size of an image connection is set at run–time by the algorithm by which it is produced. However, the PCT communications scheme must know both the input and output image dimensions and types before setting up the static communication schedules. For this reason, and elaborate initialization procedure must be followed by the PCT. First, the configuration is down-loaded and interpreted to determine the local data flow schedule. The the image dimensions and data type of the local PCT group input connection are received. Neither the local connections or the group output connection dimensions or data type can be determined until the algorithms have set them, so the scheduler must then call the setup routines of each of the algorithms. After the algorithm setup functions query the algorithm input types and dimensions and set the output types and dimensions. After the setup functions have run, all local and output image connections will have been initialized, and the scheduler can send the output image connection data type and dimensions to the next group scheduler. This seemingly innocuous initialization procedure entailed the re-writing of approximately one half of the PCT kernel.

2.1.2 Improved PCT IPLib API

To ease the programmer's task in integrating new algorithms into the IPLib, an improved API (Application Programmer's Interface) was built. A description of the PCT Iplib API can be found in the MIRTIS IPLib Programmer's Manual.

6-12

2.1.3 Support of "Non-Streamed" Data

The original PCT run–time system was built for executing computations on long streams of input data. The systems were considered to be *persistent*, in that they could be assumed to execute forever on and infinite stream of data. Obviously, for applications using file I/O, this is not a fair assumption. It may be that a single, large image is to be processed.

The PCT communication system operates in such a way that no data exits the system unless data is entering the system (the input data can be though of as pushing the data through and out of the system). In order to better support file I/O applications, the PCT run–time system must be modified so that output data progresses independently of the availability of input data. This enhancement will require the modification of the PCT C40 DMA programs. This work was planned for this summer, but has yet to be implemented. The system will operate on files as is, but to better serve file I/O applications, the modification should be made as soon as possible.

2.2 Modifications to the MIRTIS Model Interpreter

The addition of varying data dimensions and types required changes to the model interpreter. The interpreter builds performance models which are used in decomposing and scaling the data flow. Since the image dimensions and types are no longer the same throughout the data flow, the performance models must be aware of the dimension and type of each image connection before constructing communication and computation costs functions (communication and computation times vary with image size). The interpreter modification were made during the summer.

A second interpreter modification was required to support file I/O because, in the past, the data sources and sinks were assumed not to be HostNodes (see 3). In other words, no computation could be allocated to a PC host. This required a change in the part of the interpreter which finds appropriate nodes for the data source and data sink, and a path in between them for data to flow.

A third modification to the interpreter was required. Since the path the data follows through the hardware can now include PCs, which will not be booted by the network loader, the part of the interpreter which determines the C40 sequence was separated from the data path selection. The interpreter now first determines the data path (a Hamiltonian path from the source node to the sink node), allocates the data flow to the path, then finds the boot sequence (a spanning tree of the network connectivity graph) in a separate step.

2.3 Additional Host Interface Services

In order to flexibly support file I/O, the host interface services had to be broadened. In particular, the system originally only supported image sequences stored as Sun rasterfiles. AEDC Advanced

Missile Signature Center (AMSC) has developed an in-house image, image sequence, and image archive file format called Standard Archive Format (SAF). The SAF library is available, but the only PC version that is supported is compiled with Microsoft Visual C++ (MSVC). For this reason, the entire MIRTIS host interface library was ported to MSVC, including the *tick* boot loader, the PCT scheduler interface utilities, the dynamic parameter adjustment server, and the existing file I/O utility. The new MSVC-compatible host interface suite now also includes utilities for reading/writing SAF files.

3   Installation and Documentation of MIRTIS

During the summer, a MIRTIS system was installed at AEDC's AMSC to be used in new large image sequence applications. Jim Nichols, who has been involved in the MIRTIS effort as both the AEDC sponsor and an active developer, has over-seen the writing of the MIRTIS installation and documentation. Although the entire manual set is still being written, a strong draft of the MIRTIS IPLib Programmer's Manual has been delivered. The utility of the manual was tested by allowing a programmer to develop a new IPLib algorithm. The editing process of the manual set is still under way.

4   Conclusions

The migration of MIRTIS from the pure real-time imaging domain to more off-line file I/O applications is expected to broadened it's usefulness to AEDC AMSC personnel greatly. Although the support required substantial modifications to the run-time system, the interpretation algorithm, and the host interface services, it should be noted that re-implementing a new system from scratch to support the new applications would have been far more costly. Because MIRTIS was build using the MGA approach of generating implementations from high-level models, migrating the existing system to new domains can be (and has been) done while re-using a large part of the implementation.

REFERENCES

1. G. Karsai, "A configurable visual programming environment," *IEEE Computer* , pp. 36–44, March, 1995.

2. M. S. Moore and J. Nichols, "Model-based synthesis of a real-time image processing system," *Proceedings of the First IEEE International Conference on Engineering of Complex Computer Systems (ICECCS)* , (Ft. Lauderdale, FL), November, 1995.

3. M. S. Moore, *Model-Integrated Program Syntheis for Real-Time Image Processing*. PhD thesis, Vanderbilt University, Nashville, TN, May, 1997.

4. M. S. Moore *et al.*, "A model-integrated program synthesis environment for parallel/real–time image processing," *Proceedings of the SPIE Conference on Parallel and Distributed Methods in Image Processing (SPIE97)* , (San Diego, CA), July, 1997.

5. E. A. Lee *et al.*, "Static scheduling of synchronous data flow programs for digital signal processing," *IEEE Transactions on Computers* **36**(1), pp. 24–35, January, 1987.

6. M. S. Moore, "A dsp-based real-time image processing system," *Proceedings of the 6th International Conference on Signal Processing Applications and Technology (ICSPAT)* , (Boston, MA), October, 1995.

7. A. Ledeczi, *Parallel Systems With Flexible Topology*. PhD thesis, Vanderbilt University, Nashville, TN, 1995.

INVESTIGATION OF FLUID MECHANICAL PHENOMENA RELATING TO
AIR INJECTION BETWEEN THE SEGMENTS OF AN ARC HEATER

Robert L. Roach
Assistant Professor
Department of Aerospace & Mechanical Engineering

University of Tennessee Space Institute
Tullahoma, TN 37388

Final Report for:
Summer Research Program
Arnold Engineering Development Center

September 1997

INVESTIGATION OF FLUID MECHANICAL PHENOMENA RELATING TO AIR INJECTION BETWEEN THE SEGMENTS OF AN ARC HEATER

Robert L. Roach
Assistant Professor
Mechanical & Aerospace Engineering Department
University of Tennessee Space Institute

Abstract

The details of the fluid mechanical behavior near the air injection slot in a segmented arc heater were computed. A vortex was found to exist at the downstream corner of the gap between segments. This vortex was shown to have the potential to entrain hot gas into the area between the segments. The presence of hot gas in this area increases the effective electrical conductivity making the arc heater much more susceptible to arcing between segments. The computations also suggest alternative air injection arrangements which could minimize the slot vortex strength and the vulnerability to arcing.

INVESTIGATION OF FLUID MECHANICAL PHENOMENA
RELATING TO AIR INJECTION
BETWEEN THE SEGMENTS OF AN ARC HEATER

Robert L. Roach

Introduction

Hypersonic testing requirements for propulsion, materials, and structures testing dictate high pressures and enthalpies not attainable by conventional means. Electric arc heaters are the only viable method to achieve high temperatures to simulate atmospheric reentry conditions for a test duration of several minutes. Arc heaters have been used with success in both aeronautical and industrial testing.

An arc heater is an apparatus that uses a continuous electrical discharge to heat air to very high temperatures. An arc is stretched down the length of the constrictor from an anode to a cathode (see Fig. 1). Air is forced through he constrictor and becomes heated by the electric arc. The configuration of the standard AEDC H1 segmented heater includes a heater constrictor which consists of some 200 segments that are insulated form each other, so that a reasonable voltage drop may be sustained between each pair of segments. Air is injected tangentially to the inside wall to create a vortex to help stabilize the arc. Each of the copper segments is internally water cooled. The arc heated air is subsequently expanded through a supersonic nozzle and allowed to impinge on test models in order to simulate the high enthalpy condition experienced during flight. The standard AEDC H1 arc heater configuration uses eight groups of 24 copper segments each (column modules) and a nozzle throat diameter of 0.9 in as shown in Fig. 2.

The arc has traditionally been stabilized in the center portion of the constrictor by (a) the cold constrictor wall and (b) the vortex created by tangential injection of the air at the constrictor wall. In high pressure arc heaters (20-100 atm), vortex stabilization becomes the dominant phenomena. It has been observed, however, that at pressures above 100 atm, vortex stabilization is not always sufficient to prevent the arc from seeking an alternate path through the copper segments by jumping the air injection gaps between the segments. This can result in severe hardware damage.

Discussion

It has been shown in Felderman and Bruce [1] that the presence of hot gas near the injection slots between arc heater segments make it much more likely that a portion of the arc will seek an alternate path through the copper segments by jumping the air gap between segments. Basically, the hot gas has increased electric conductivity so that the only significant resistance remaining is that associated with the surface phenomena. If this is comparable to the

resistance of the gas inside the constrictor, the alternate path becomes feasible. This investigation concentrated on the fluid phenomena in this area and how it may relate to the presence of hot gas in this area.

Methodology

A computational grid was defined consisting of a one-quarter section of a local area of the arc heater surrounding one injection slot as shown in Figure 3. A time-dependent code, the VENUS code, was used to compute the flow in the defined portion of the arc heater. Heat addition was used to simulate the joule heating process inside the arc heater.

The Solver

A Navier-Stokes solver, VENUS [2] was used for the flow computations. VENUS (Vectorized Euler-Navier-Stokes Unsteady Solver) is an Approximate Factorization (AF) algorithm which uses 4th-order compact differencing for the evaluation of all explicit derivatives and a 2nd-order evaluation of the implicit derivatives. When the steady state is reached, the implicit side is zero leaving the solution spatially 4th-order accurate. For unsteady problems, 2nd-order time accuracy is used and inner iterations can be used to drive the implicit side to zero, keeping 4th order spatial accuracy in the unsteady solution. The code was used here to solve for the flow in the slot region described in Figures 3 and 4. As the details of the solver have been described elsewhere, it is only necessary to describe the initial and boundary conditions used for this problem. These are given in the next section. For the heat addition computations, a heat source term was added to the energy equation.

Initial Conditions

Initial Conditions in the Main Chamber

The initial conditions were created in the FORTRAN program 'arcjetic.f' using input from the SWIRLARC [3] PC-based computer program. SWIRLARC computes an axisymmetric flow in an arc heater. Radial profiles of the dependent variables are provided at various axial stations as part of the output of the computation. These profiles are thus available to be used as initial conditions for a genuinely 3D computation. The one change necessary, to use the SWIRLARC profiles, is to zero out the radial velocity component at the inflow. To mimic the gas injection from the side walls, SWIRLARC assumes the injected gas velocity to be uniform along the length of the walls and not at discrete locations corresponding to the slots. As a consequence, the normal velocity on the real solid surfaces was not assumed to be zero in the SWIRLARC computation. In order to properly compute the effects of the slots, this is changed to match the real conditions in the present computation. In this case, the radial velocity is quite small and is safely set to zero for the inflow.

The solution at the x = 2.0 meters station was used for the inflow profiles. In addition, the profiles at the

7-4

station x = 2.41 meters were also available for the downstream initial conditions. Again, the radial velocity was set to zero to account for the no slip walls. Since the SWIRLARC program uses a different grid for its computations, these profiles must be interpolated. This is done using parametric cubic splines.

Once the inflow profiles have been set, the same profiles are assumed in the remainder of the chamber. Hence the profiles of the dependent variables at every station downstream is set equal to those at the inflow.

Initial Conditions in the Slot

Injection ports exist along the outer radius of the slot. These ports are rectangular and their locations are specified by the grid indices (IS1,KS1)-(IS2,KS2) and so forth for as many ports as are in the domain. In the results reported, there were two ports.

Initial conditions in the slot depend on the flow from the injection ports. For the present, these ports were assumed to be aligned tangentially to the slot outer radius. Gas flow through the injection ports was assumed to be an isentropic expansion from the difference between the stagnation pressure of the injected gas and the stagnation pressure from the main chamber. As the flow is subsonic, the mass flow rate must also be specified. Finally, parabolic profiles are assumed for the tangential velocity in two directions across the slot. With these assumptions a maximum velocity is found an placed into the program. Computation of maximum velocity should be automated since it can be found, but this has not yet been done.

Initial conditions from the injection ports are found by resolving the tangentially injected gas into the y- and z-velocity components. These are multiplied by an average gas density. The axial component of the velocity (x-direction here) is assumed to be zero. The remaining variables in the injection port faces are set equal to those along the remainder of the slot outer radius. The solid walls in the slot are set with no slip conditions (all velocities zero) and density and energy set equal to their stagnation values.

Conditions on the two circumferential planes in the slot are also set equal to the stagnation conditions with zero velocities. This condition, unfortunately may be responsible for the slow convergence of the flow in the slot region. The values of the velocities at convergence will be quite far from zero. Since the initial conditions of the interior depend on the values of the slot boundaries, any conditions which are far from the final conditions will slow the convergence. Hence, it is recommended that these conditions be changed to something more resembling the final solution. Perhaps some fraction of the injection velocity distributed over the region would be reasonable.

The final boundary of the slot region is the opening to the main chamber along the slot inner radius which abuts the main chamber. The conditions on this boundary have already been given by the initial conditions in the

main chamber.

With all boundaries in the slot specified, the conditions in the volume are filled in using a three-dimensional TransFinite Interpolation [4]. This technique produces smooth values of the dependent variables in the volume using information from all the boundaries.

<u>Boundary Conditions</u>

Inflow Boundaries

Inflow boundaries are located at the entrance to the main chamber and the injection ports in the circumferential slot. In both cases the inflow is subsonic meaning that there is one characteristic leaving from the interior. This requires the adjustment of one of the flow variables. In this case, the total energy was allowed to relax using the boundary condition formulation of Rudy and Strikwerda [5] which is based a type of Riemann invariant argument. In their formulation the sound speed on the inflow boundary is computed using the values of the dependent variables at the old time level. A new inflow pressure is then computed based on the difference between the normal velocity between the inflow and the next station:

$$P_1 = P_2 - \rho_1 a_1 (u_2 - u_1)$$

With this new pressure, a new inflow total energy is computed.

Outflow Boundary

The only outflow boundary is in the main chamber. It lies geometrically close to the inflow boundary also in the main chamber. Since the outflow boundary is subsonic, one characteristic crosses the boundary from outside the domain and hence one quantity should either be held fixed or some invariant quantity should be held. Much has been written in the literature on these boundaries given such considerations. Usually it is desired to place the outflow boundary far from the main parts of the flow so that any errors propagating in from the quantities that are held fixed don't disrupt the solution. For internal flows, this is very difficult to do without putting a sonic throat downstream. For these reasons, and for a more stable computation from the initial conditions, a simple first-order extrapolation of all variables from the station upstream of the outflow boundary was used. This has the advantage of being a rather dissipative condition which then damps any disturbances from the outflow boundary. It has the disadvantage of not allowing the imposition of a lower downstream pressure. As the solution began to settle, the outflow boundary was changed to a more proper imposition of the downstream pressure. In this case, the pressure at the outflow was computed using the non-reflecting condition of Rudy & Strikwerda [5] which depends on the unsteady conditions happening at the outflow and the downstream imposed pressure. Thus the outflow pressure was computed using:

$$p_{imax}^{n+1} = \frac{1}{1 + \alpha \Delta t} \left[p_{imax}^n + \alpha \Delta t p_{out} + \rho_{imax}^n a_{imax}^n \left(u_{imax}^{n+1} - u_{imax}^n \right) \right]$$

where a_{imax}^n is the outflow boundary sound speed at the nth (old) time level, and α is a relaxation parameter found

to be optimal in the range of [.3-.5]. Note that the axial velocity must be found before this pressure can be computed. The axial velocity and the other velocity components, along with the density are all extrapolated. With the pressure found from the above expression, the total energy is then computed from the other variables.

Solid Wall Boundaries

No-slip, adiabatic solid wall boundary conditions were used. Thus the velocities were held fixed at zero. The zero-gradient density and pressure were computed to second order by expressions like:

$$\rho_1 = \frac{(\rho_2 \Delta x_3^2 - \rho_3 \Delta x_2^2)}{\Delta x_3^2 - \Delta x_2^2}$$

The total energy is then computed from the new pressure.

Centerline Boundary

The centerline boundary is a geometric axis of symmetry so that only the tangential velocity component is nonzero. Thus the two normal components were held fixed at zero. The tangential velocity and the thermodynamic variables are all found by an averaging procedure. The value at the centerline for each of the k-planes is found using the same zero gradient expression as is used at solid walls. Since all these values are coincident for any given axial station, these values are averaged. The average value is then given to all of the coincident points. This process is done for the axial velocity, the density, and the pressure. The total energy is then computed from these variables.

Periodic Boundaries

The periodic boundaries are the k = 1 and k = kmax planes. Since advantage has been taken of the symmetry of the problem, these planes are involved in the imposition of periodic conditions on the flow. These two planes are located at one station beyond the quarter-plane region as shown in Fig. 5. Since the conditions at planes k = 2 and k = kmax-1 are supposed to be the same, then it makes sense that the conditions for plane k = 1 should match those of the plane just before k = kmax-1. Similarly, the conditions on the plane k = kmax should match those of plane k = 3. Thus, after the interior points are computed, the thermodynamic variables and axial velocities on the k = 1 and k = kmax planes are set equal to the values on their respective matching planes. The velocity components perpendicular to the axial direction, though, must be rotated in each case by 90 degrees due to the different orientation of the planes.

Results

Computations were first done without heating. A vortex was identified as shown in Fig. 6 and 7.. In the close-up view shown in Fig. 6, the vortex is clearly evident near the downstream edge of the injection slot where it intersects the constrictor wall. Fig. 7 shows that this vortex persists throughout the entire circumferential slot region. The reverse flow noted (ie, back into the injection slot) clearly has the potential to entrain phenomena from the free stream.

Figure 8 shows the temperature contours of the cold flow solution in the near region of the intersegment slot. Note that, consistent with the vortex at the same location, the temperature in the upper portion of the slot is drawn down the upstream edge of the slot while the main chamber temperature is drawn up into the slot on the downstream edge. This indicates that had the main chamber been heated, it would be possible for hot gas to enter the slot region. This was borne out in the heat addition cases.

Next, heat was added to the constrictor flow in a manner predicted by the SWIRLARC code. This was done by using the heating initial conditions from the SWIRLARC code and computing a reasonable radial heat addition. The new centerline temperature was approximately 8000 °K. The flow was computed and temperature profiles are shown in Figures 8 and 9. The alternate light and dark areas shown indicate temperature contours with the darker regions being the hotter gas. Note that a finger of the dark heated gas from the constrictor has been entrained up into the injector slot. A series of radial location is shown in Fig. 9. The entrainment problem is more severe at the injection points but appears to be a problem at all locations.

A simple computation of an average conductance was computed using an average temperature across the slot from the two computations. These are summarized in Table I below.

Table I. Average Temperature and Conductance in the intersegment slots of an Arc Heater

Case	Slot Temperature	Conductance, σ_1
cold flow	815 °K	3.9 1/ohm-cm
hot flow	1566 °K	82.1 1/ohm-cm

It is thus seen that the conductance increases by over an order of magnitude under such conditions. Hence it may be reasonably assumed that this may significantly affect the susceptibility to arcing between the intersegment slots.

Conclusions

It has been shown that a) a vortex exists at the downstream corner of the injection slot between the segments in a segmented arc heater, b) this vortex can entrain hot gas from the constrictor, and c) this makes it much more likely that a portion of the arc will seek a path along the wall, through the copper segments, jumping the gap between them. It can further be concluded that some hardware redesign will be required to eliminate or reduce this problem.

7-8

References

1. Felderman, E.J. and Bruce, W.E., III, "Application of a Near-Electrode Model to Predict Inter-Segment Arcing", AIAA-96-2315, June 1996.

2. Roach, R.L. and J. Jenkins, "Improvements to the Numerical Simulations of Flows over Delta Wings," AIAA 95-0761, AIAA 33rd Aerospace Sciences Meeting, Reno NV, Jan. 1995.

3. Shaeffer, J.F., "SWIRLARC: A Model for Swirling, Turbulent, Radiant Arc Heater Flow Fields," AIAA 78-68, AIAA Aerospcae Sciences Meeting, Jan 1978.

4. Gordon, W.J. and C.A. Hall, "Construction of Curvilinear Coordinate Systems and Applications to Mesh Generation," J Numerische Math., Vol. 1, 1973, pp. 461-477.

5. Rudy, D.H. and J.C. Strikwerda, "Boundary Conditions for Subsonic Compressible Navier-Stokes Calculations," Computers and Fluids, Vol. 9, 19981, pp.327-338.

ARC LENGTH OPTIMIZED ON ELECTRODE RINGS

DISTRIBUTED AIR INJECTION - VORTEX FLOW

SEGMENTED HEATER CONSTRICTOR (ELECTRICALLY INSULATED, WATER COOLED)

(−) CATHODE

(+) ANODE
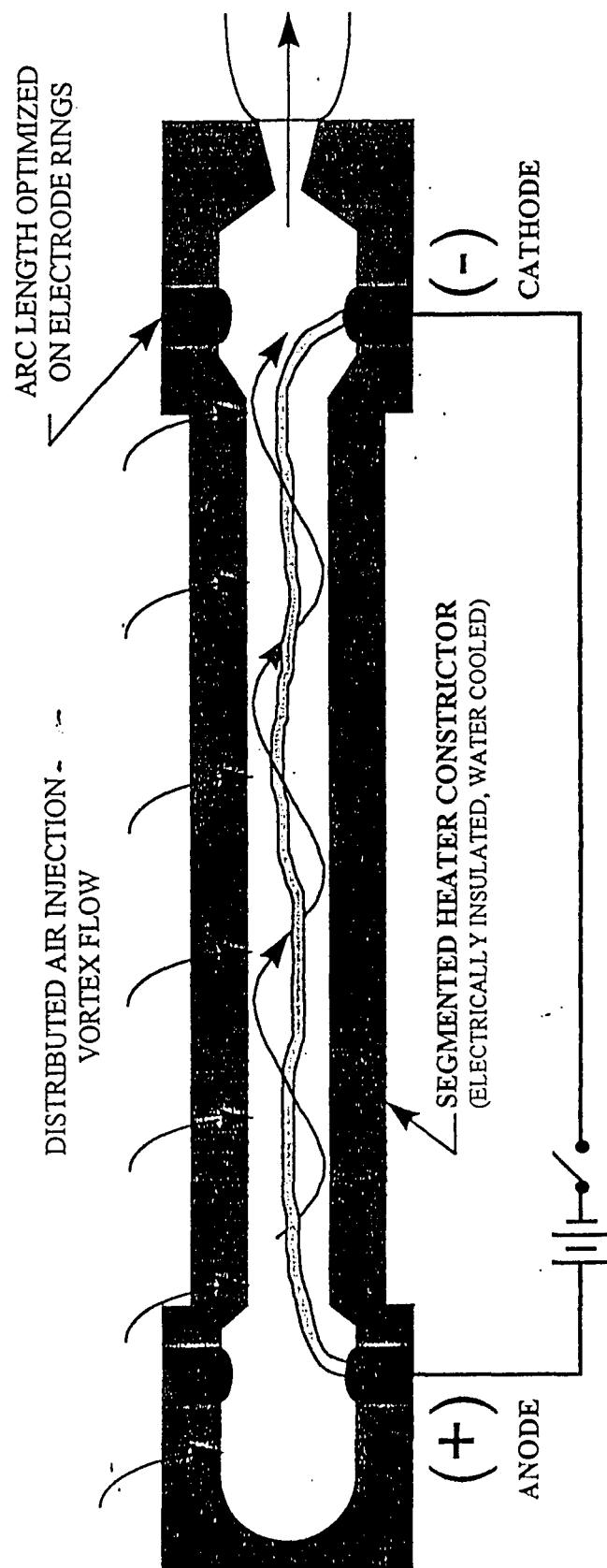
Figure 1. Segmented Arc Heater Operational Characteristics.

Note: Segment 2 - 16 Identifies
Module 2, Segment 16

Not to scale

TYPICAL MODULE SEGMENTS
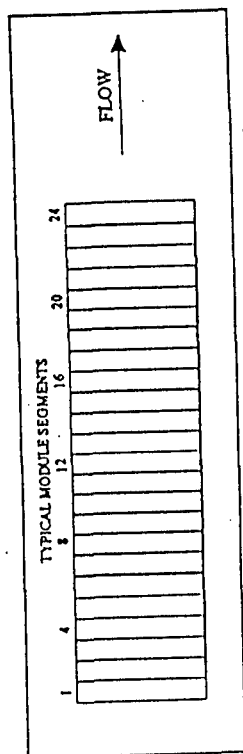
FLOW

Constrictor Modules
(Module 8 - Module 1)

FLOW

Upstream Tapered Segments (TU6-TU1)

Anode Segment (EU1)

Double Width Anode (AN1-2)
Unpowered Anode (AN3)
Upstream Segments (US8-US1)

End Cap

End Window

Downstream Tapered Segments (TD6-TD1)

Cathode Segment (ED1)

Double Width Cathode (CA1-2)

Unpowered Cathode (CA3)

Nozzle Segments (3N, 2N)

Nozzle
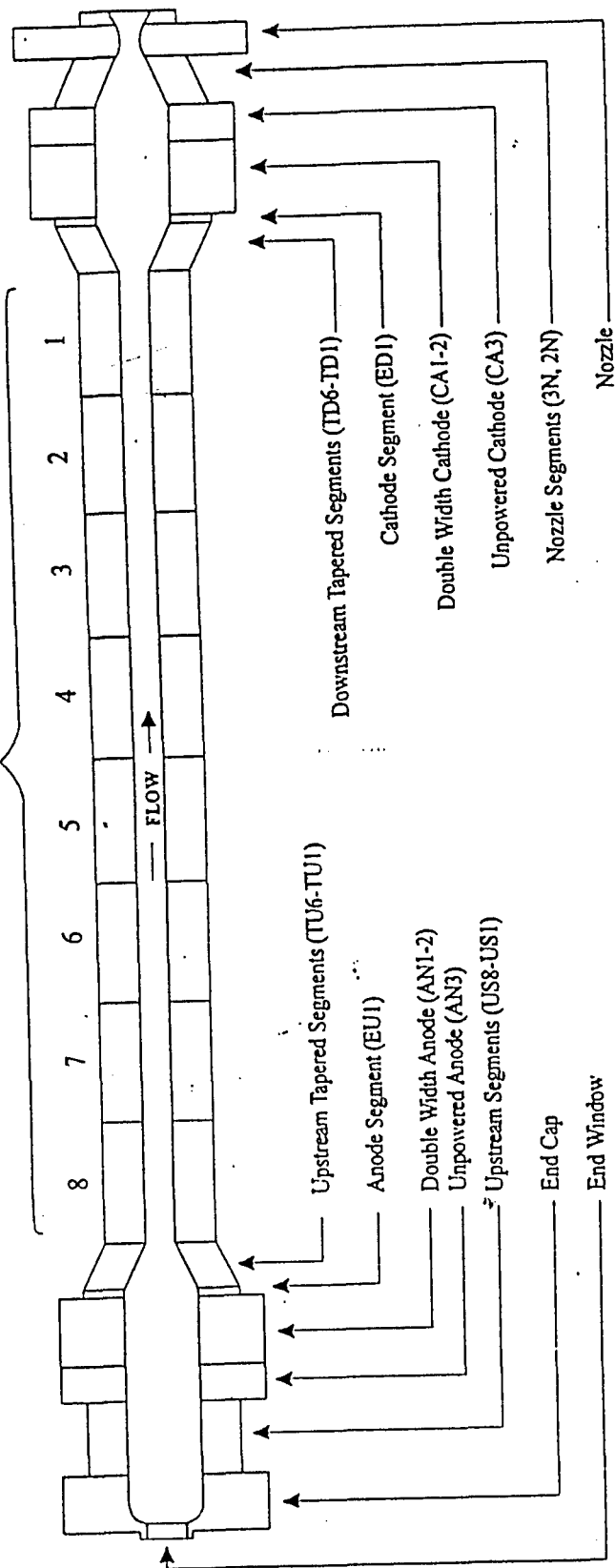
Figure 2. H1 Component Designation and Relative Locations.

Domain for Computational Grid
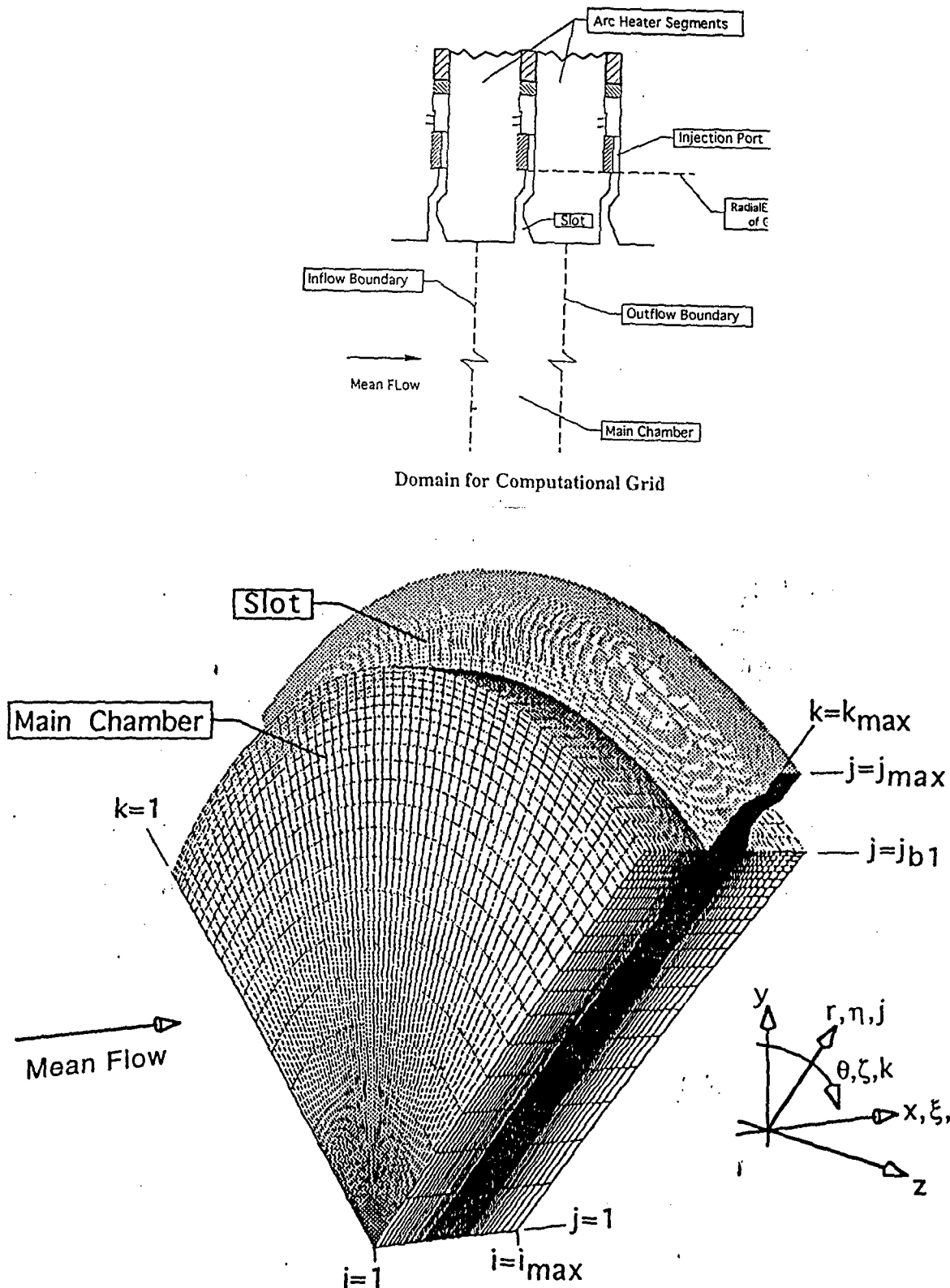
Fig. 3 Overall computational domain and grid for flow in
one-quarter of region between segments of the AEDC H-3 Arc Heater.

7-12
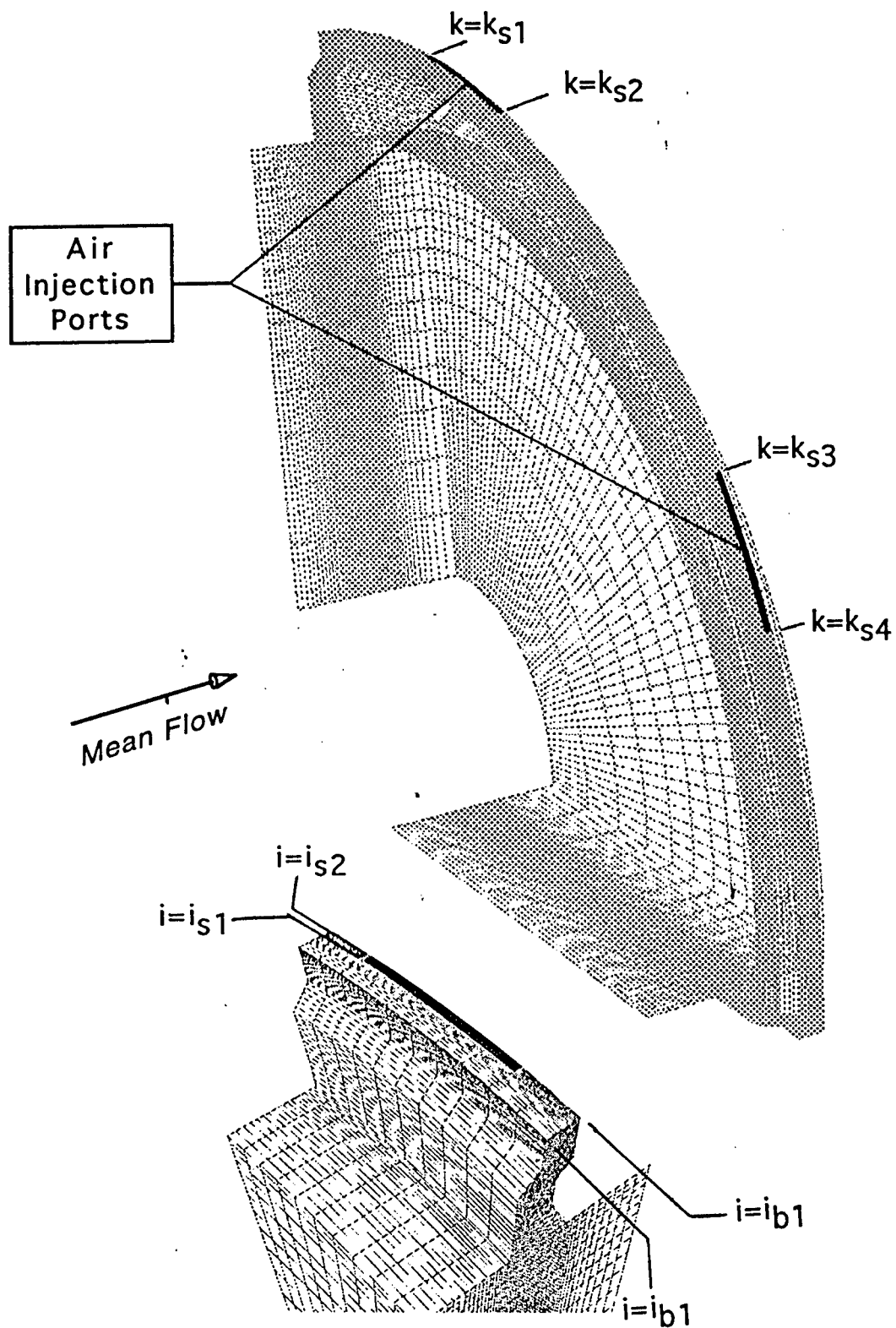
Fig. 4 Detail of grid near injection ports of the computation domain.
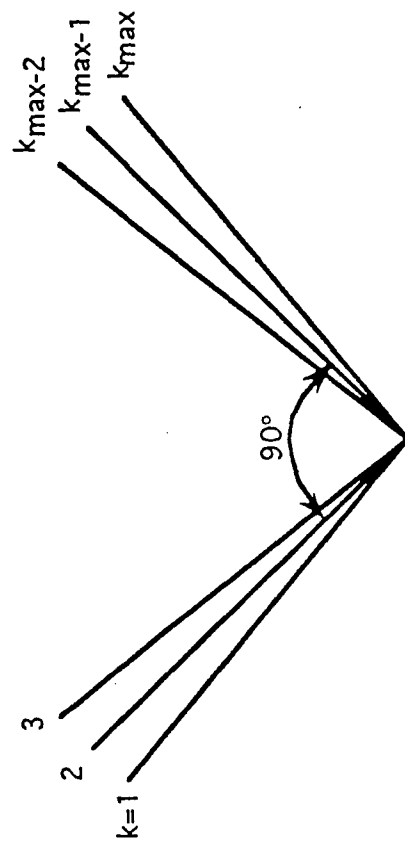
Fig. 5. Orientation of the planes in the k-direction showing the periodic planes, k =2 and k= k$_{max}$-1.
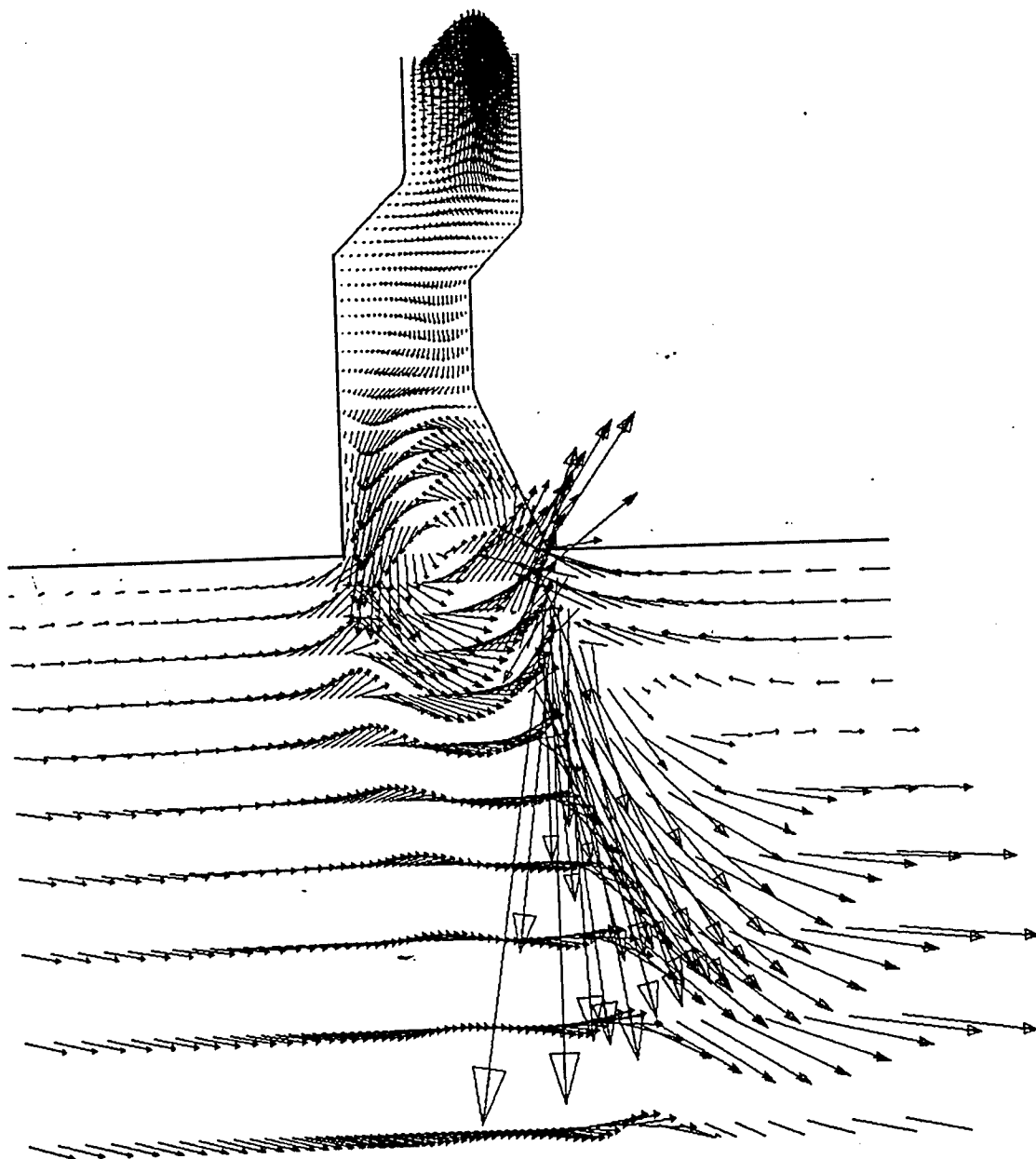
Fig. 6. Velocity Vectors at a typical k-station in the AEDC H-3 Electric Arc Heater.
$P_0 = 100$ atm, $P_{0_{plenum}} = 125$ atm.

Air Injection Port
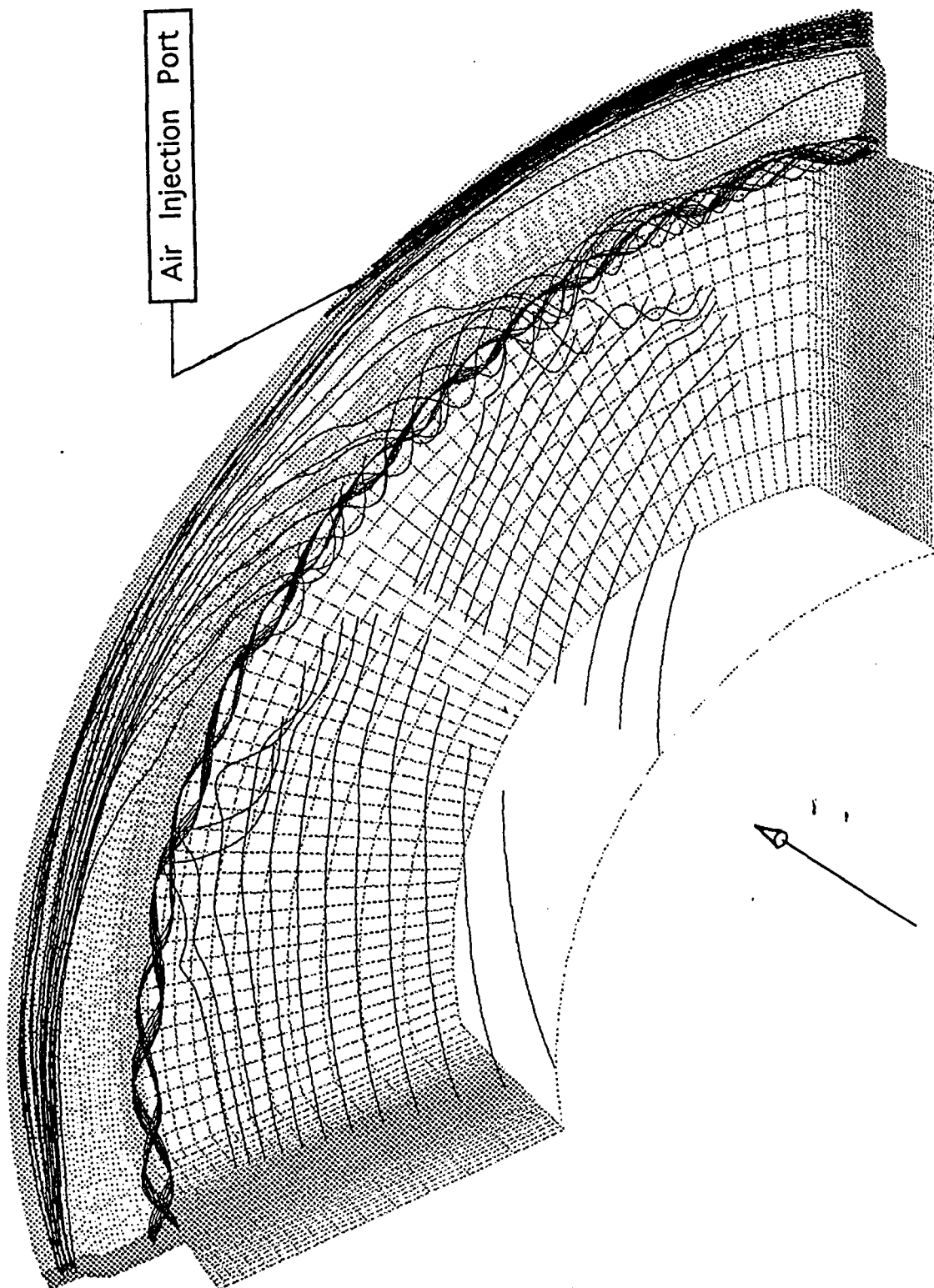
Fig. 7. Particle traces in the AEDC H-3 Electric Arc Heater.
$P_0 = 100$ atm, $P_{0plenum} = 125$ atm.
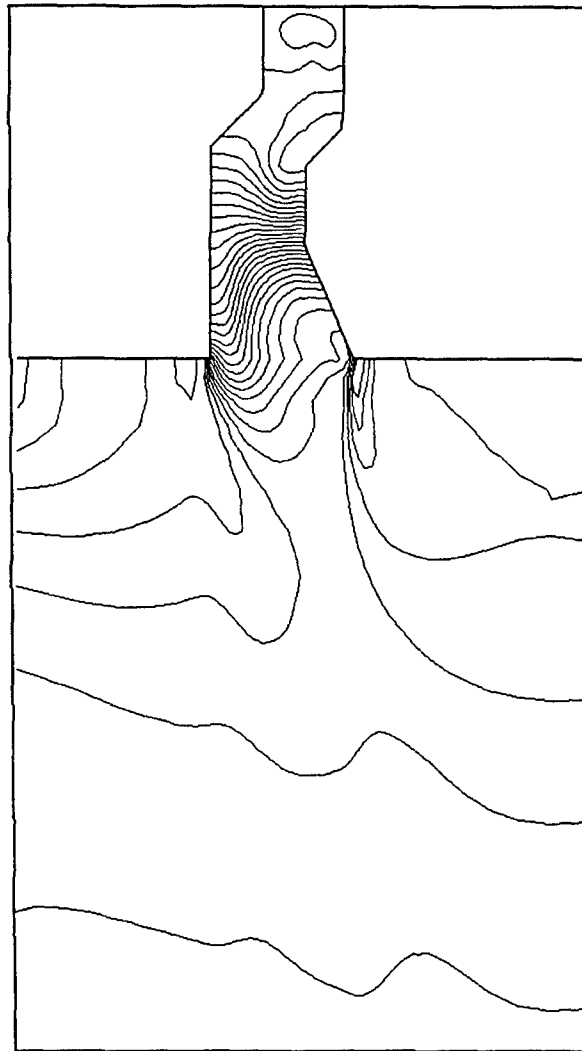
Fig. 8. Temperature contours at a typical k-station in the AEDC H-3 Electric Arc Heater. $p_0 = 100$ atm, $p_{0_{plenum}} = 125$ atm.

Fig. 9. Temperature contours at a typical k-station in the AEDC H-3 Electric
Arc Heater using heat addition.
$p_0 = 100$ atm, $p_{0_{plenum}} = 125$ atm.
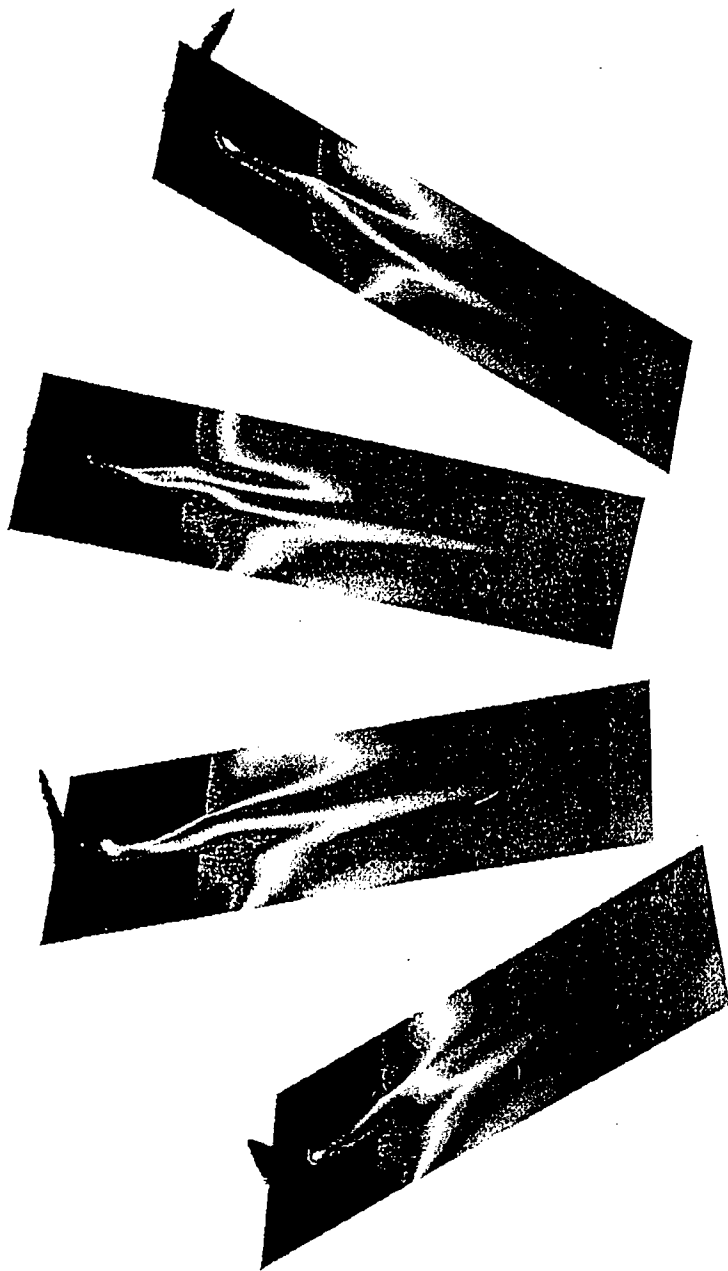
Fig. 10 Temperature contours at a several k-stations in the AEDC H-3 Electric Arc Heater using heat addition.
$P_0 = 100$ atm, $P_{0plenum} = 125$ atm.

Nicholas Winowich's report was not available at the time of publication.

Synthesis of Salts with Delocalized Anions for Use as Third Order
Nonlinear Optical Materials

Daniel M. Knauss
Assistant Professor
Department of Chemistry

Colorado School of Mines
Golden, CO 80401

Synthesis Of Salts With Delocalized Anions
For Use As Third Order Nonlinear Optical Materials

Daniel M. Knauss
Assistant Professor
Chemistry Department
Colorado School of Mines

Abstract

The synthesis of molten salts from delocalized anions was studied for application as third order nonlinear optical materials. Sodium O-methylxanthate was synthesized and the sodium counterion was exchanged for 1-ethyl-3-methylimidazolium. Molten xanthate salts with butyl pyridinium counterion were prepared by a similar cation exchange. The resulting molten salts were characterized by NMR, IR, and UV-Vis spectroscopy. The molten salt with 1-ethyl-3-methylimidazolium counterion was found to display no observable freezing point above –195°C (boiling point of liquid nitrogen) by differential scanning calorimetry. A transition to a glassy solid was observed visually, but the transition temperature could not be detected by differential scanning calorimetry.

Synthesis Of Salts With Delocalized Anions
For Use As Third Order Nonlinear Optical Materials

Daniel M. Knauss

Introduction

Materials which show third order susceptibilities are potentially important as all optical

devices. The development of third order materials has received very little attention compared to

second order NLO materials. This has resulted in a poor understanding of the structure-property

relationships for third order optical nonlinearity. Much still needs to be learned about the

fundamental factors that determine the third order response, but it is known that a high

concentration of polarizable electrons, a large conjugation length, and a small optical bandgap

influence the magnitude of the third order susceptibility.[1] The third order materials do not

require noncentrosymmetric structures, any intramolecular charge transfer due to electron

withdrawing - electron donating pairs, or any bulk orientation.

As in second order nonlinear optical materials, the major research thrust has been to find

ways to increase the value of γ, the second hyperpolarizability of the individual molecule, in

order to increase the value of X^3, the second hyperpolarizability tensor term for the bulk material.

Most studies have been undertaken to determine how γ is affected by synthesizing sequentially

built conjugated structures.[1,2] Various classes of conjugated polymers such as polydiacetylenes,

polyacetylenes, polythiophenes, poly-p-phenylenevinylene, and ladder polymers are being studied to help elucidate the optimal materials and the structure - property relationships.

Substantial improvement in current properties of this class of materials is required for device application. A value of X^3 on the order of 10^{-7} esu or higher is required. Elimination of resonant nonlinearity (where absorption takes place) is also required for useful application.[1]

Recent efforts at the Chemistry Research Center at the US Air Force Academy have examined novel techniques for enhancing the polarization of diffuse electrons.[3] It was found that delocalized anions show potential for large hyperpolarizabilities, γ. The objective of the summer research was to synthesize delocalized organic anions and form salts with delocalized cations and to ultimately test their third order nonlinearities and evaluate this technique for the design of nonlinear optical materials.

Results and Discussion

Computational work has calculated the hyperpolarizabilities,γ, for delocalized anions of higher row elements such as sulfur (Figure 1)[4]. Based on these results, a project was initiated to synthesize simple organosulfur anions and to make organic salts with 1-ethyl-3-ethylimidazolium or N-butylpyridinium counterions. O-methyldithiocarbonate ion was chosen as the anion and was produced as O-methylxanthic acid, sodium salt because of its ease of synthesis and relative stability. Trithiocarbonate dianion was considered due to its higher

calculated second hyperpolarizability, γ, but it was decided that for a preliminary investigation, the simpler O-methyldithiocarbonate would be more suitable.

O-methylxanthic acid, sodium salt was prepared by a method from the patent literature.[5]

This same reference described the method for exchanging the sodium counterion for a hydrazinium ion. This method was adopted for the preparation of both 1-ethyl-3-methylimidazolium and N-butylpyridinium salts.

Preparation of O-methylxanthic acid, sodium salt

4.63 g (0.116 moles) sodium hydroxide and a magnetic stir bar were placed into a dry 100 ml round bottom flask. 50 ml of methanol were added and the reaction mixture was purged with nitrogen and stirred by magnetic stirring until all of the sodium hydroxide had reacted into solution. 7.2 ml (0.120 moles) of carbon disulfide were added and the reaction flask was stoppered. An immediate yellow color was observed and the reaction mixture was stirred for 12 hours. At the end of this time, the solution was filtered to remove a small amount of insoluble materials. The solvent and excess carbon disulfide were removed by rotary evaporation to yield 14.98 g (99%) of a yellow solid. NMR spectrum of this solid in DMSO-d_6 is shown in Figure 2. The O-methylxanthic acid, sodium salt was found to be soluble in THF, acetone, methanol, acetonitrile, and ethyl acetate, but insoluble in ether and methylene chloride.

Preparation of N-butylpyridinium chloride.

13.35 ml (0.165 moles) pyridine and 15.0 ml (0.14 moles) chlorobutane were combined in a 100 ml round bottom flask. A condenser was attached and the mixture heated to reflux for three days. At the end of this time, 50 ml of chlorobutane were added to precipitate the product salt. The salt was filtered and washed with chlorobutane and quickly placed into a tared round bottom flask which was placed in a vacuum oven at 80°C to dry. (Yield, 6.02g, 25%.) NMR spectrum of the resulting solid in DMSO-d_6 is shown in Figure 3.

Preparation of N-butylpyridinium-O-methylxanthate salt

5.35 g (0.031 moles) N-butylpyridinium chloride were added to a tared 100 ml round bottom flask and dried under vacuum. 60 ml methanol were added followed by 4.056 g (0.031 moles) of O-methylxanthic acid, sodium salt. The flask was sealed with a rubber septum. The colorless pyridinium solution turned yellow upon the addition of the O-methylxanthic acid, sodium salt and immediate precipitation of NaCl was observed. The reaction mixture was left to stir for 12 hours. The methanol was removed by rotary evaporation followed by placement on the high vacuum line. Dry acetone was added to the rust colored oil to yield a reddish solution and precipitated NaCl. The NaCl was filtered, washed with acetone and dried under vacuum to yield 1.78 g (98.3 %). The acetone solution of the product was worked up by removing the acetone by rotary evaporation followed by placement on the high vacuum line. A quantitative yield of product as a reddish oil was obtained. NMR spectrum of the resulting solid in DMSO-d_6 is

shown in Figure 4. The N-butylpyridinium-O-methylxanthate salt was found to be soluble in methanol, acetone, and slightly soluble in ethyl acetate.

Preparation of 1-ethyl-3-methylimidazolium-O-methylxanthate salt

Previously synthesized 1-ethyl-3-methylimidazolium chloride was dried on the high vacuum line. 8.14 g (0.056 moles) of 1-ethyl-3-methylimidazolium chloride was weighed into a 250 ml round bottom flask inside a glove box. 100 ml of methanol were added to dissolve the salt. 7.23 g (0.056 moles) of O-methylxanthic acid, sodium salt were then added all at once to the clear, colorless solution. The solution immediately became a turbid yellow. The reaction was left to stir at room temperature for 12 hours. The methanol was removed by rotary evaporation followed by placement on the high vacuum line. 100 ml of dry acetone were added and to precipitate NaCl which was removed by filtration. The NaCl was washed with acetone and dried under vacuum to yield 3.14 g (96.0%). The yellow/green acetone solution of the product was concentrated by rotary evaporation and the remaining acetone was removed on the high vacuum line to yield a quantitative amount of yellow/green product. NMR spectrum in DMSO-d_6 is shown in Figure 5. Solubility of the product was similar to that of N-butylpyridinium-O-methylxanthate salt.

<u>Characterization of products</u>

Characterization of the reactions was done by NMR as depicted in Figures 2-5. These figures demonstrate the quantitative nature of the reactions. Peak shifts from that of the starting N-butylpyridinium chloride and 1-ethyl-3-methylimidazolium chloride shown in Figures 3 and 6, respectively can be seen when compared with Figures 4 and 5. The cation exchange for the imidazolium compound shows a large shift for the downfield acidic proton from 9.51 ppm to 9.23 ppm. Other peaks display relatively small shifts, although the introduction of the O-methyl peak at 3.72 ppm can be readily seen. Integration of the peaks further shows quantitative product formation. The cation excahnge for the pyridinium compound shows much less pronounced peak shifts. Nevertheless, the most downfield aromatic doublet shifts from 9.26 and 9.29 ppm for the pyridinium chloride to 9.13 and 9.16 ppm for the pyridinium O-methylxanthate with no residual peak above 9.20 ppm.

UV-Vis spectroscopy was used to characterize the organic salts. Very different absorbance behavior was observed for the two samples. The UV-Vis absorbance spectrum for N-butylpyridinium-O-methylxanthate salt is shown in Figure 7 and that for 1-ethyl-3-methylimidazolium-O-methylxanthate salt is shown in Figure 8. Of interest is that neither material displays absorbance above 500 nm and 1-ethyl-3-methylimidazolium-O-methylxanthate salt displays no absorbance from 450 to 800 nm.

The thermal stability and thermal transition temperatures were examined by thermogravimetric analysis (TGA) and differential scanning calorimetry (DSC), respectively. 1-

ethyl-3-methylimidazolium-O-methylxanthate salt was analyzed by TGA (Figure 9) and compared with O-methylxanthic acid, sodium salt (Figure 10). Some weight loss, presumably due to water, can be seen below 200°C for each of the materials. At approximately 180°C, both materials begin to lose weight, with very different profiles observed for each. DSC performed on the 1-ethyl-3-methylimidazolium-O-methylxanthate salt (Figure 11) shows no readily discernible transitions. Optical observation of the salt also shows no crystallization down to liquid nitrogen temperatures.

Characterization of nonlinear optical properties

Attempts at characterizing the nonlinear optical properties of the salts produced were unsuccessful due to a lack of the proper equipment. Current work has been directed towards the characterization of these materials and the synthesis of other potential candidates based on delocalized anions.

Conclusions

The successful synthesis of O-methylxanthate salts with N-butylpyridinium and 1-ethyl-3-methylimidazolium counterions was accomplished. These materials are unique room temperature molten salts, displaying a large liquid-state range. These materials may show third order nonlinear optical behavior.

References

(1) Reinhardt, B. A. *Trends in Polymer Science* **1993**, *1*, 4.

(2) Prasad, P. N. *Introduction to Nonlinear Optical Effects in Molecules and Polymers*; John

Wiley & Sons: New York, 1991.

(3) Yeates, T.; Wilkes, J. S., Personal communication

(4) Figure by Yeates, T.; Wilkes, J. S.

(5) Steyermark, P. R. *Hydrazinium xanthates and method for their preparation*; W. R. Grace

and Co.: USA, 1967.

Computational Results: Second Hyperpolarizability
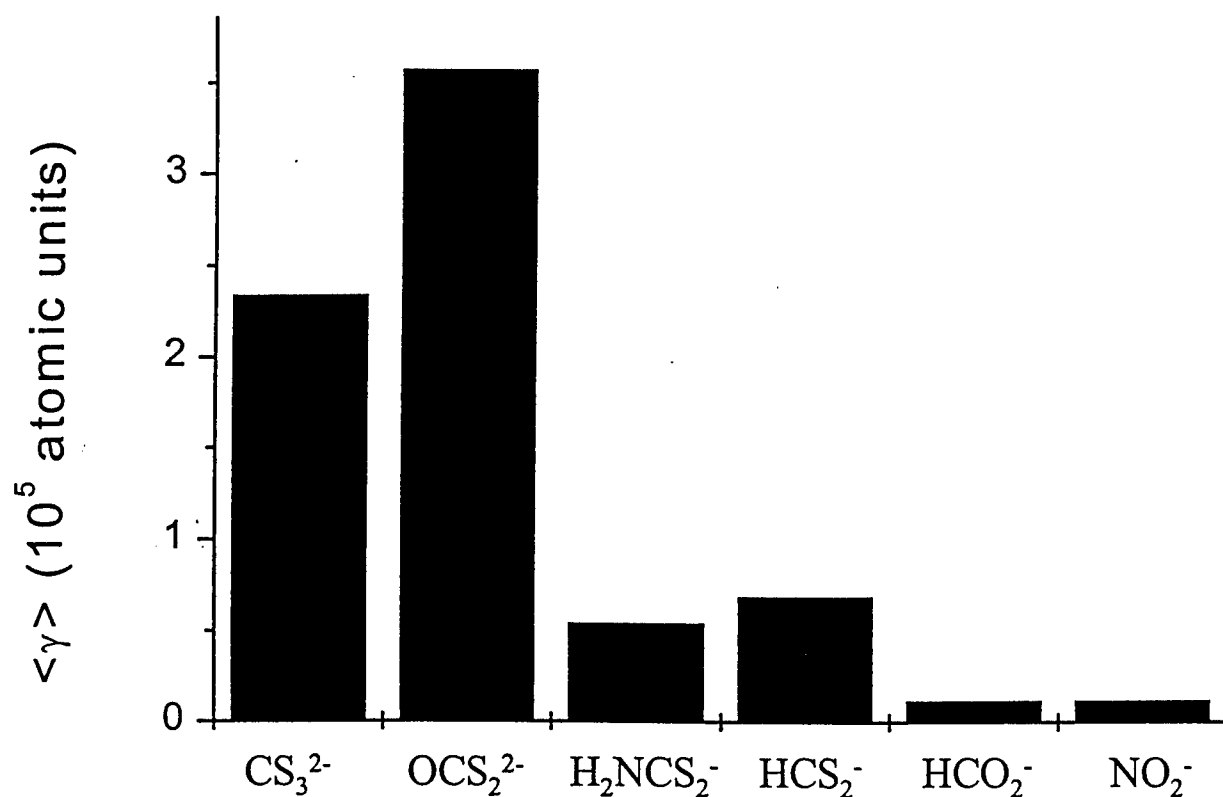
Figure 1: Calculated second hyperpolarizabilities for selected anions of higher row elements
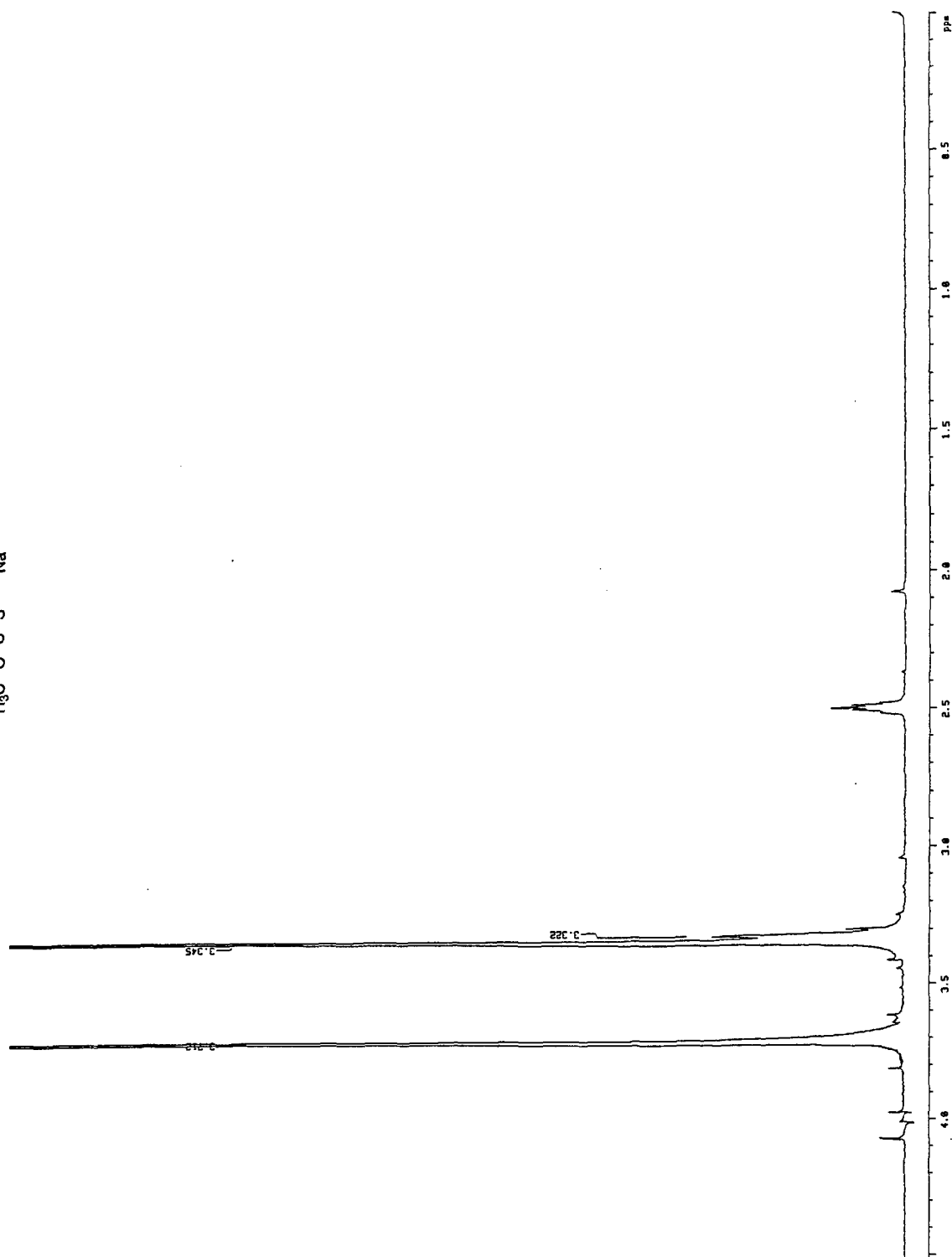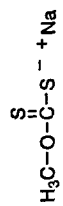
Figure 2: NMR spectrum of O-methylxanthic acid, sodium salt in DMSO-d$_6$
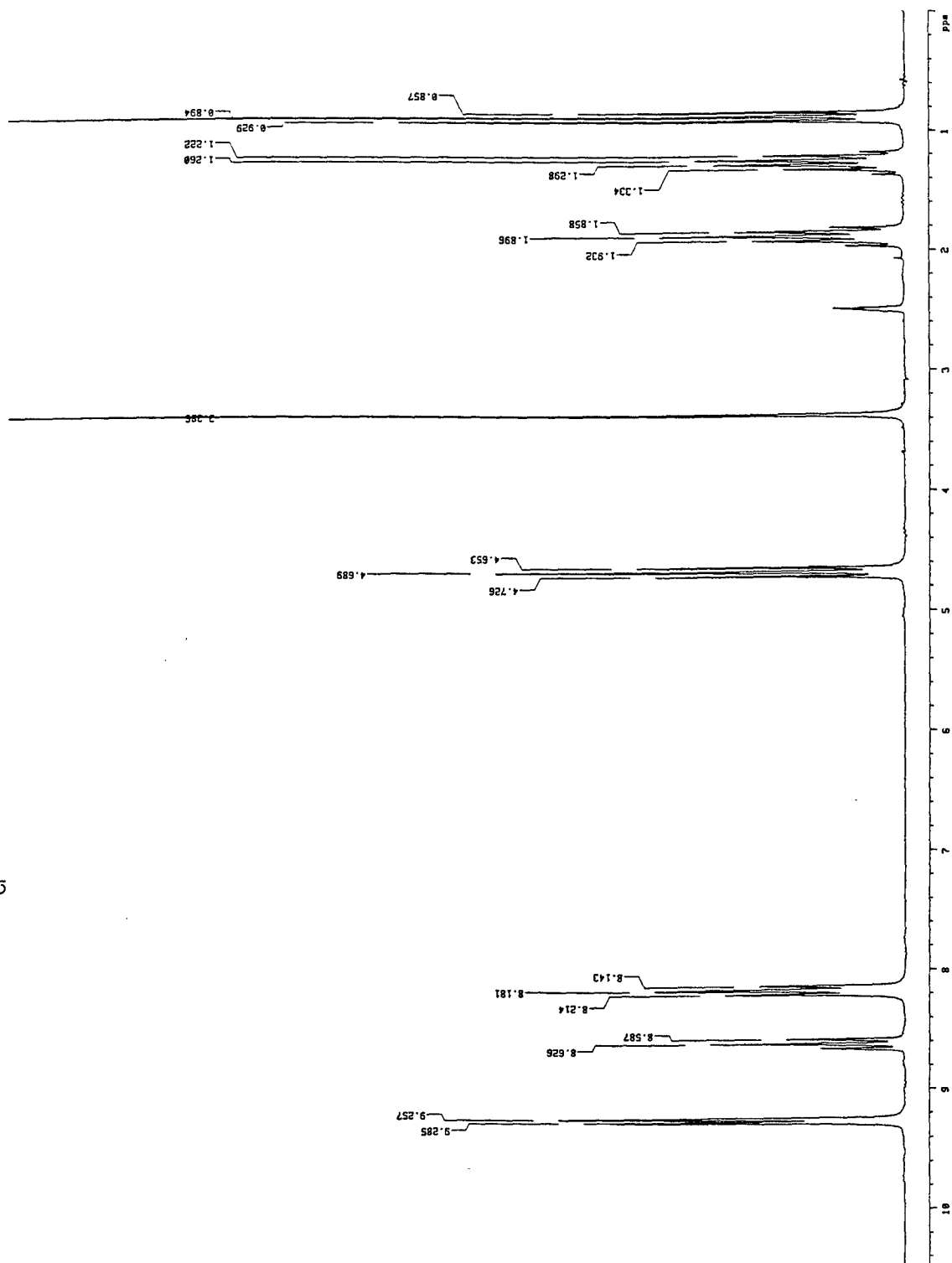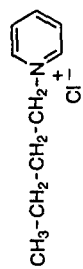
Figure 3: NMR spectrum of N-butylpyridinium chloride in DMSO-d$_6$
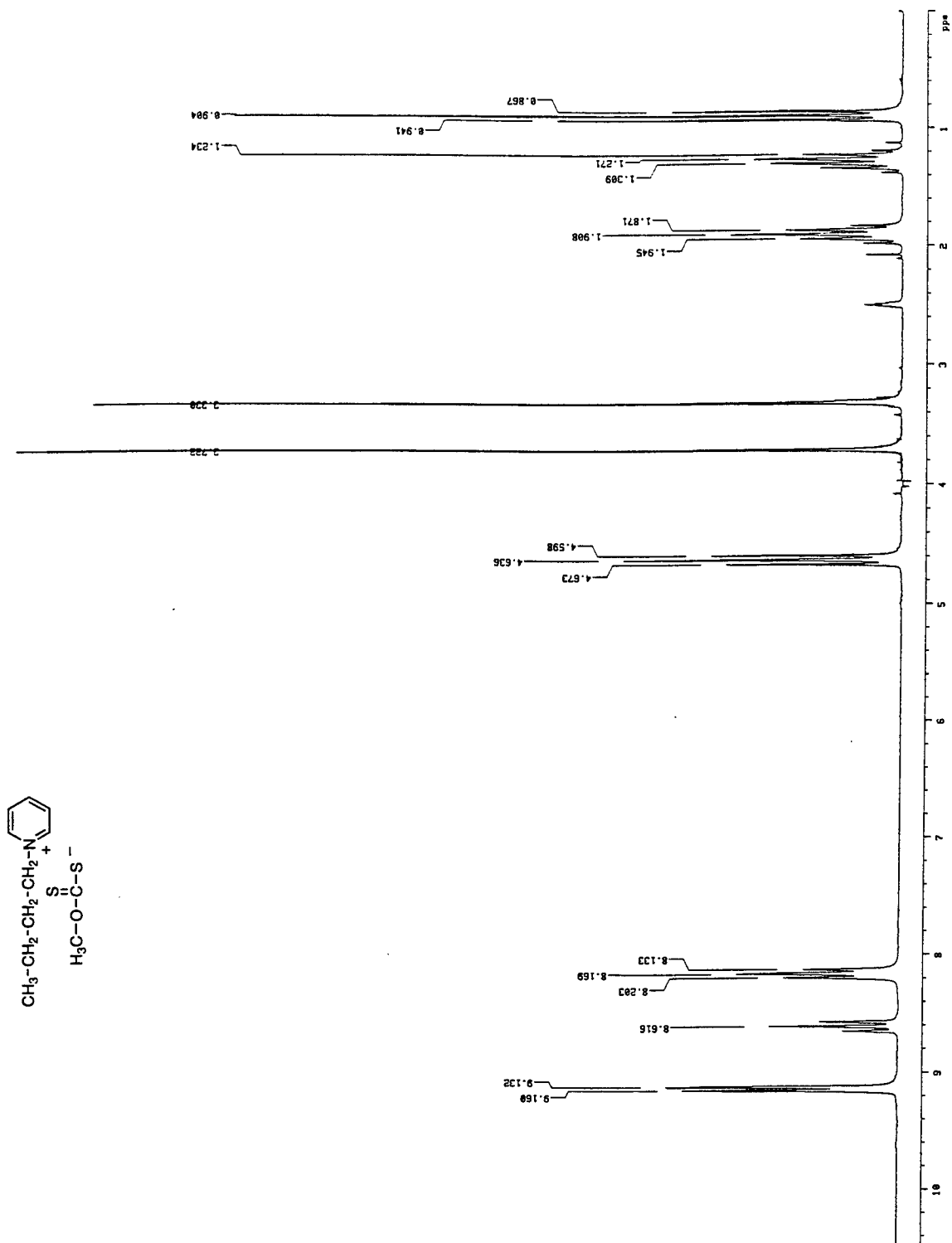
Figure 4: NMR spectrum of N-butylpyridinium O-methylxanthate in DMSO-d$_6$

9-14
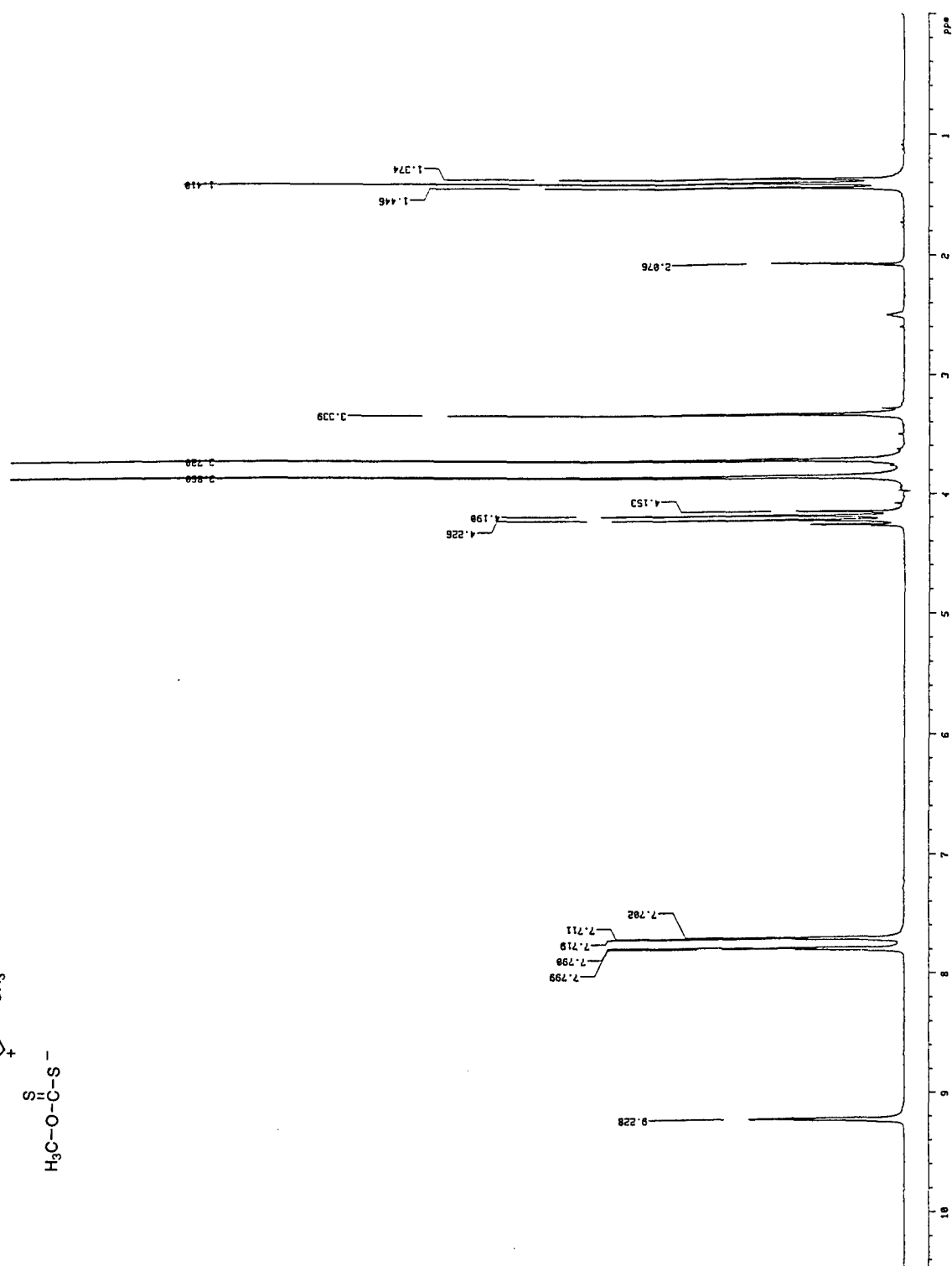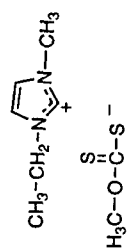
Figure 5: NMR spectrum of 1-ethyl-3-methylimidazolium-O-methylxanthate in DMSO-d$_6$
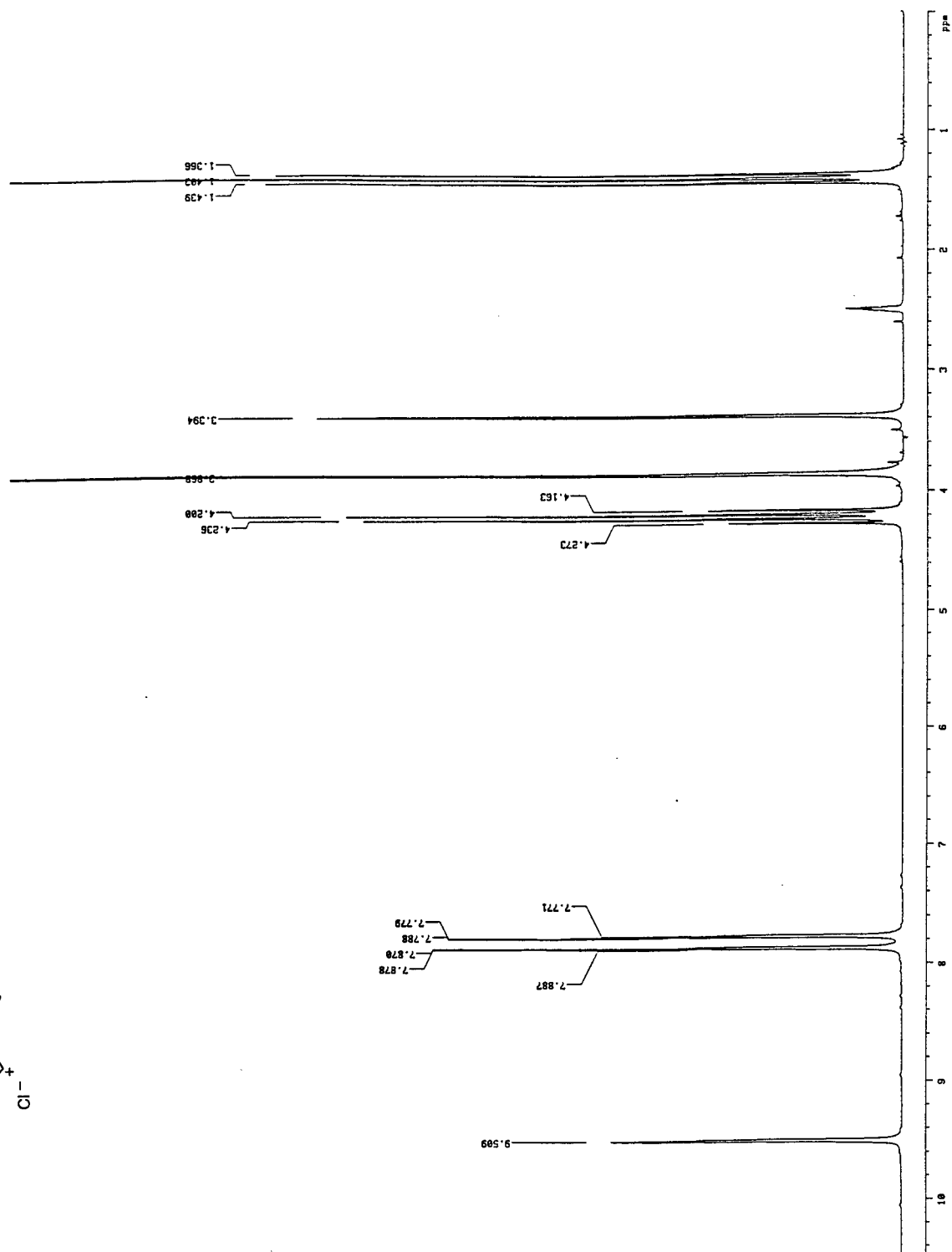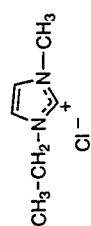
Figure 6: NMR spectrum of 1-ethyl-3-methylimidazolium chloride in DMSO-d$_6$
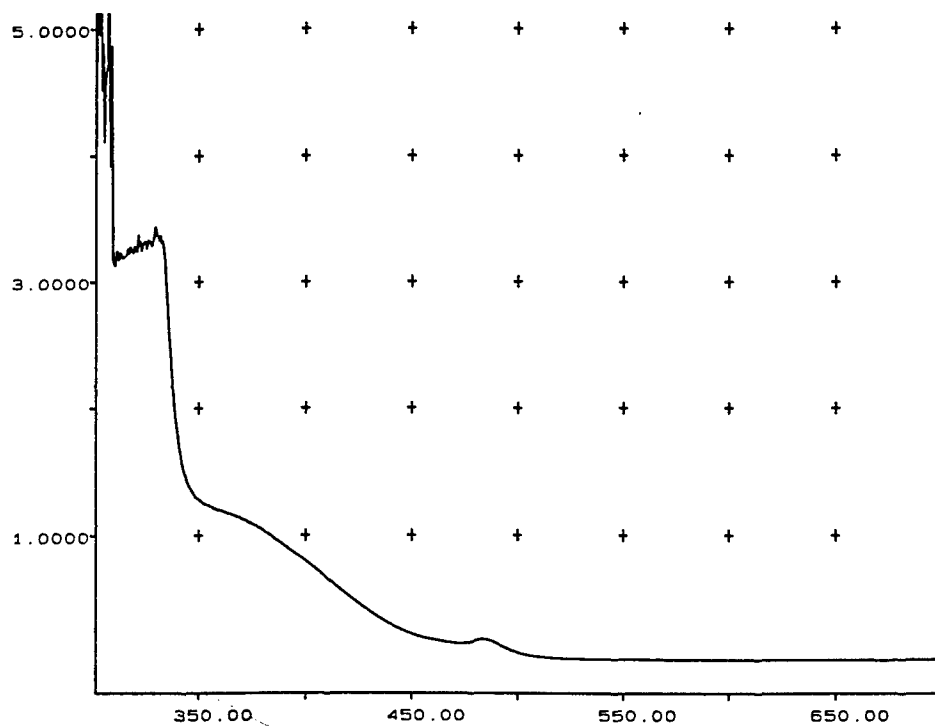
Figure 7: UV-Vis absorbance spectrum of N-butylpyridinium-O-methylxanthate
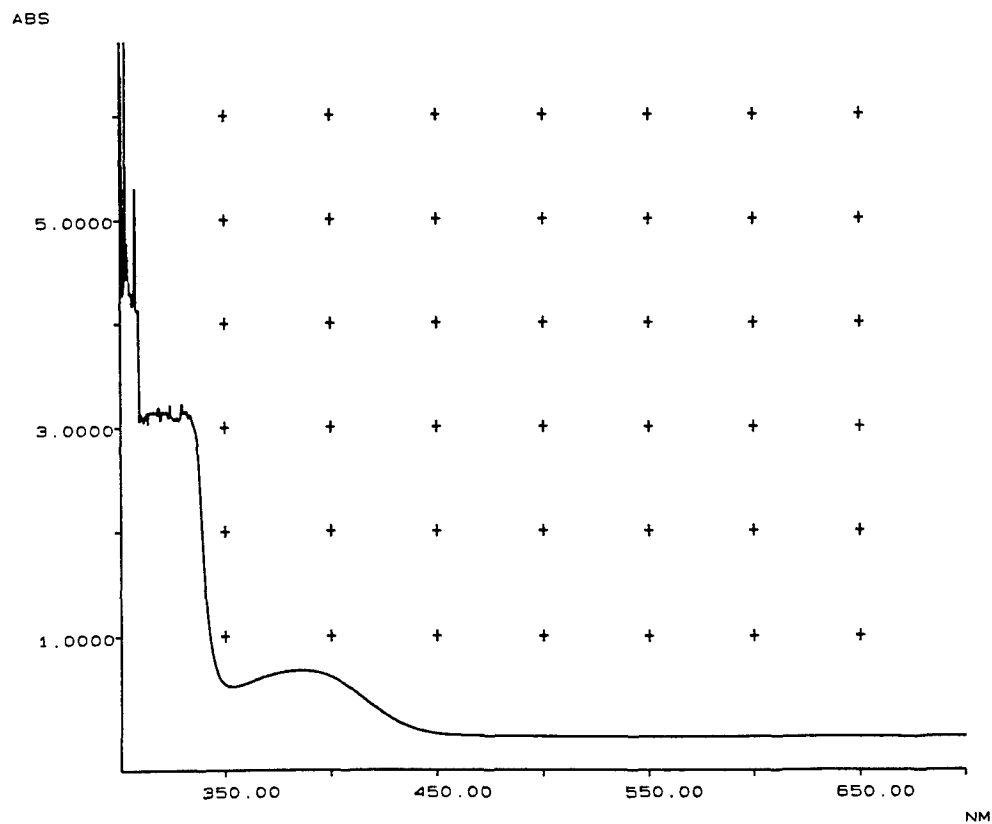
Figure 8: UV-Vis absorbance spectrum of 1-ethyl-3-methylimidazolium-O-methylxanthate
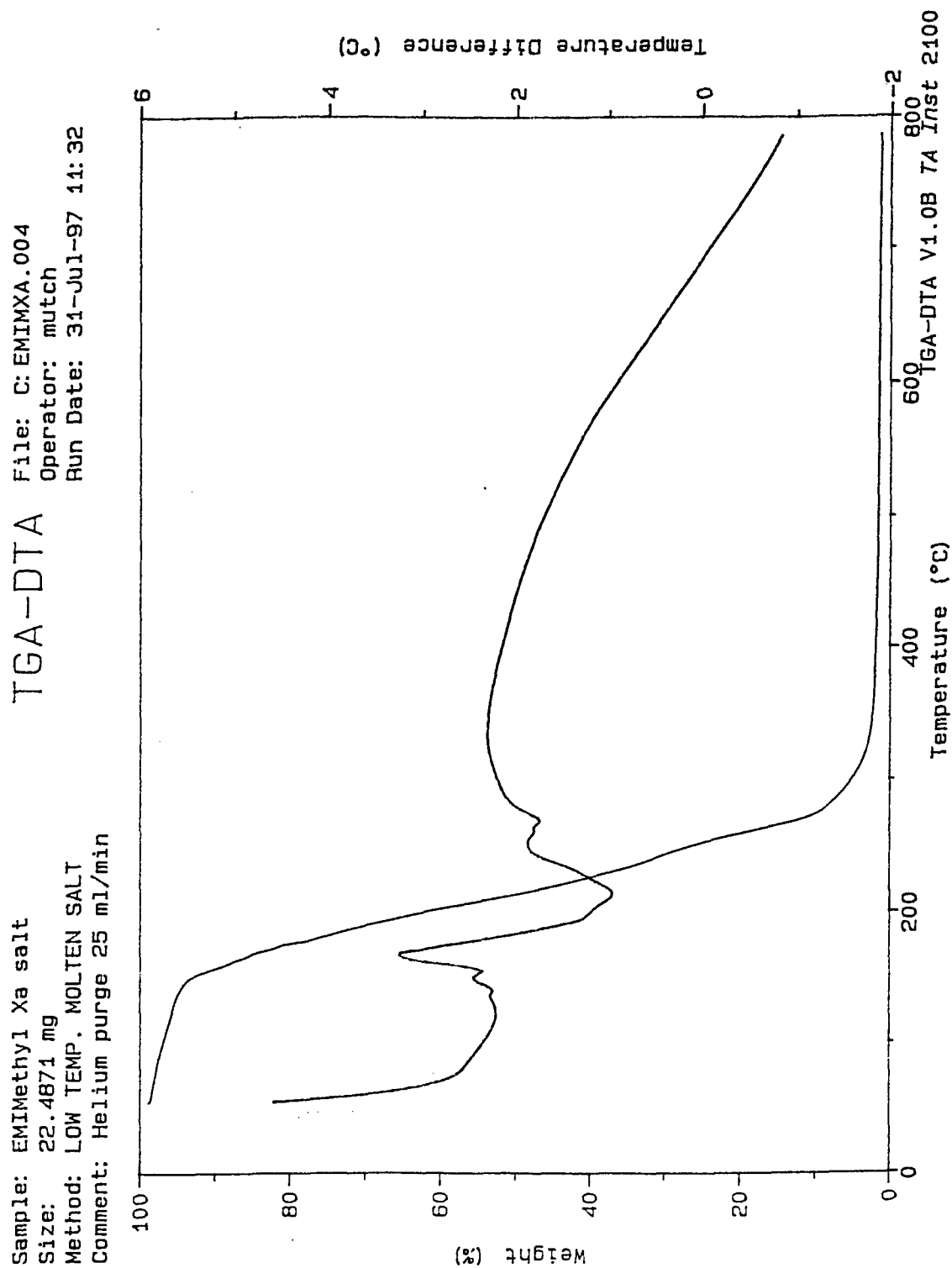
Figure 9: TGA-DTA analysis of 1-ethyl-3-methylimidazolium-O-methylxanthate
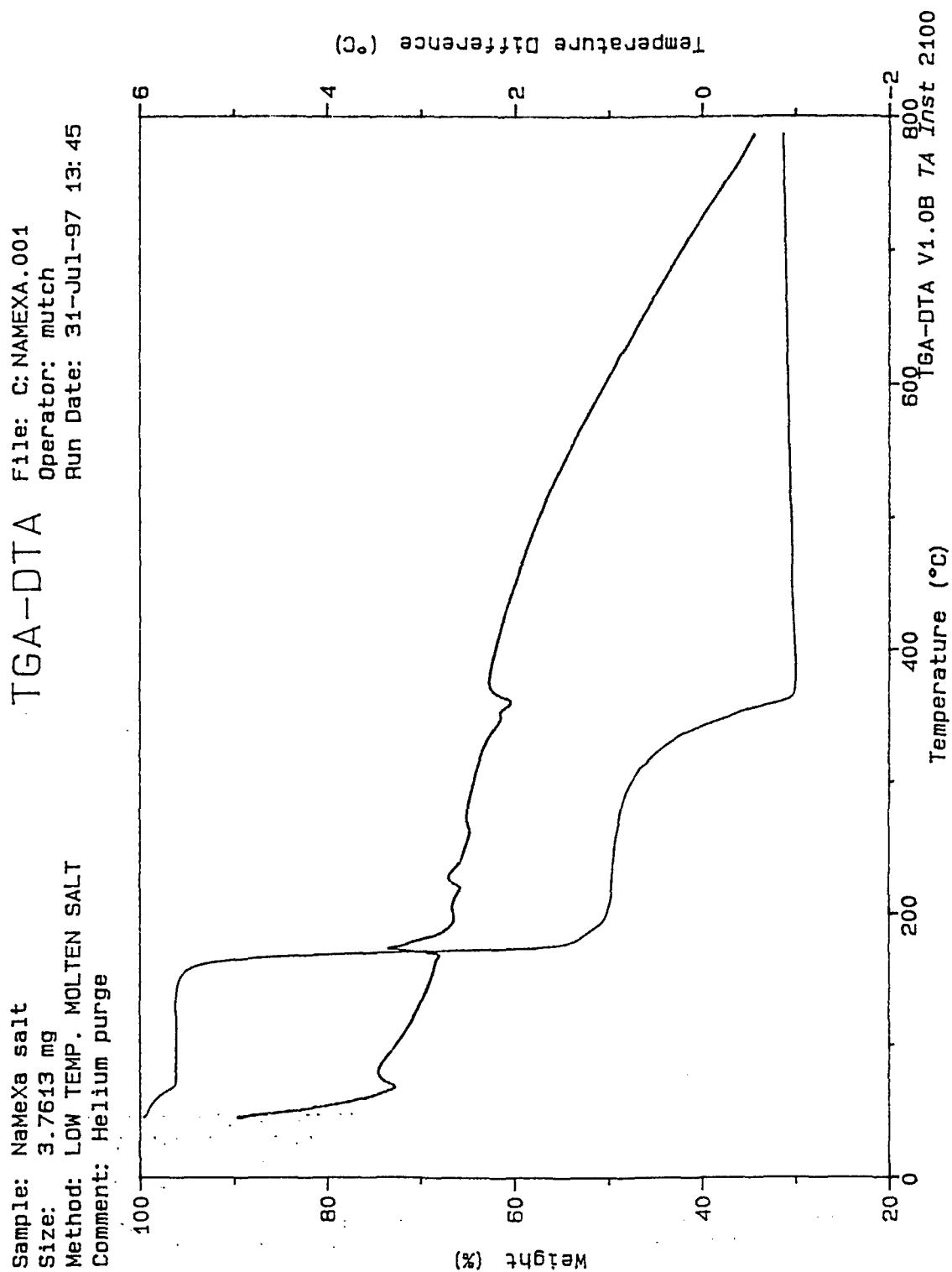
Figure 10: TGA-DTA analysis of O-methylxanthic acid, sodium salt
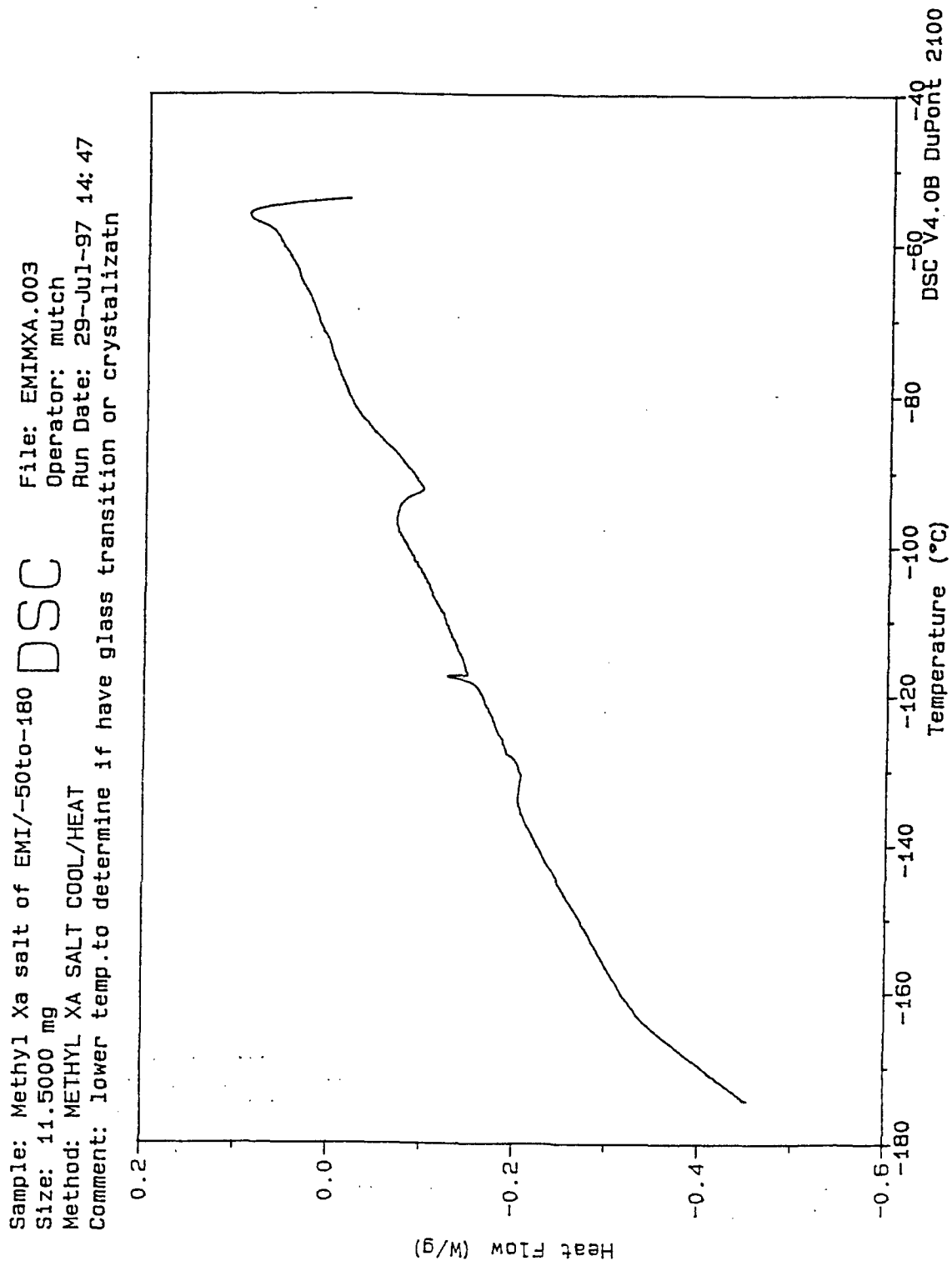
Figure 11: DSC analysis of 1-ethyl-3-methylimidazolium-O-methylxanthate

Raster-To-Vector Conversion of Circuit Diagrams: Software Requirements

Jeffrey M. Bigelow
Associate Professor
Department of Electrical Engineering

Oklahoma Christian University
Oklahoma City, OK 73131

RASTER-TO-VECTOR CONVERSION
OF CIRCUIT DIAGRAMS:
SOFTWARE REQUIREMENTS

Jeffrey M. Bigelow, Ph.D.
Assistant Professor
Department of Electrical Engineering
Oklahoma Christian University

Abstract

This paper presents the requirements needed for Raster-to-Vector Conversion software to effectively convert schematics from scanned raster to vector format. Primary attention is placed on the need for a software package to understand the particulars of the typical format of circuit diagrams. The needs analysis suggests that with such a dedicated software package the entire process could approach the desired goal of fully automated document conversion.

RASTER-TO-VECTOR CONVERSION
OF CIRCUIT DIAGRAMS:
SOFTWARE REQUIREMENTS

Jeffrey M. Bigelow, Ph.D.

Introduction

The capability to convert engineering documents into a useful, revisable electronic format has become highly

desirable. Electronic storage is much more convenient than paper drawings due to the reduction in physical space,

the ease of retrieval and the elimination of the aging process and vulnerability of paper. Electronic forms are also

more transmittable via computer than paper drawings allowing more competitive outsourcing of replacement

parts. Engineering drawings can be stored electronically in two formats: raster or vector. Raster format arranges

graphical data as picture cells (pixels) arranged in uniform rows and columns that recreate the drawing when

viewed on a monitor or printed on paper. The density of pixels measured in dots per inch (dpi) determines the

resolution or clarity of the image. In general, raster files use large amounts of computer memory and thus are

expensive in terms of memory to store. A preferable format is vector. Vector format represents the graphical

data as lines or other geometries drawn from point to point rather than having to store each pixel along the

geometry. For this reason, vector format takes up much less memory than raster format. Text is stored as ASCII

characters and thus information such as engineering notes, parts list, and drawing information can be extracted

and stored in a meaningful and useful data base. Because the data is vectorized, vector formats also easily allow

for scaling and editing of the image. This feature is essential to allow a design to be modified or optimized or a

subset of the design to be used in a different application. Finally, computer simulation has become an essential

part of the engineering design process. This is especially important for electronic circuits. Therefore, it is

imperative for schematic drawings to be in the appropriate vector form for simulation to be possible.

For these reasons, it is not sufficient for engineering drawings to be scanned and stored in raster format. These

documents should also be converted into the appropriate vector format. Automated Document Conversion

System is a Department of Defense program to install and test new technology for Raster to Vector Conversion

(RVC). The primary motivation of RVC is a reduction in cost. Tinker Air Force Base has thousands of electronic

schematic drawings of avionics equipment needing reliability and maintainability improvements. RVC is an

essential part of meeting these needs in a cost-efficient manner. In this paper, the requirements of software are

outlined for the ability to convert schematics from scanned raster to vector format.

Raster-to-Vector Conversion

Electrical schematic diagrams, also referred to as circuit diagrams, are comprised of symbols representing

electronic components connected by straight lines representing wires. Each symbol provides information on the

type of the component (e.g. resistor, capacitor, transistor, etc.). Text is located near the symbol that gives the part number and a unique name for each component. For example, a resistor generally has a unique name that starts with R and a value that indicates its resistance. The text near a transistor, on the other hand, must indicate the part number of the transistor as well as its name. All this information: text, symbols, and wires, must be retained in order to completely define a circuit.

The RVC process is shown in Figure 1. First the schematic must be scanned into a raster form such as Tagged Image File Format (TIFF). The current standard resolutions are 200, 400 or 600 dpi. A sample TIFF file of a common-emitter amplifier is shown in Figure 2. Note the three components: text, symbols and wires.
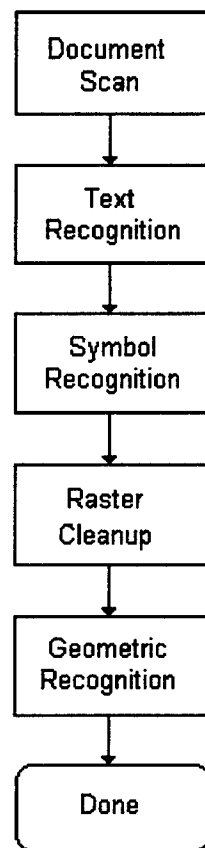
Figure 1. Raster to Vector process

The first conversion step is to recognize and convert the text containing the part numbers, names and values of each components. The raster comprising the text must be replaced with ASCII characters in a font and size that approximates the original. This can be done using automatic character recognition or manual insertion of text. After this, the underlying raster of the text must be completely erased so that it will not be interpreted as symbols

or wires later in the conversion process. Figure 3 shows the TIFF image of Figure 2 after the text has been recognized and removed.
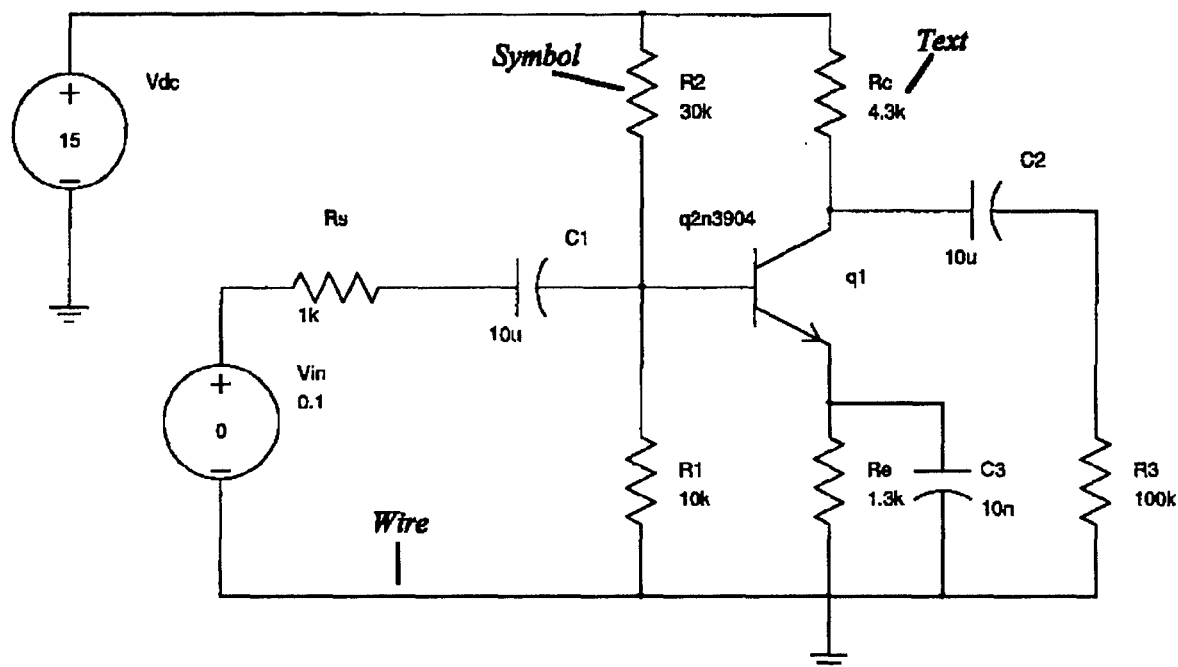
Figure 2. Circuit diagram showing the three layers to be extracted: text, symbols, and wires.

The second conversion step is to recognize and convert the symbols of the schematic. A sample of each symbol used in the collection of schematics is brought together in tabular form in a template file. Automatic Symbol Recognition (ASR) is done by comparing the raster with this template file. Each raster symbol in the template file is linked to the vector symbol in a symbol file which is used in the final vector form of the schematic. These symbols can also be used to manually insert the correct symbol onto a component through Manual Symbol Insertion (MSI). Once again, after the symbol is recognized and extracted, the underlying raster must be erased. In Figure 4, the symbols have been extracted, leaving only the wires.

The next conversion step is to convert the remaining raster to the straight lines that represent the wires of the schematic. This process is called Geometric Recognition (GR), and care must be taken to ensure all text and symbol raster has been removed or else extraneous wires will occur, possibly resulting in short circuits.
At this point the raster representation of the schematic has been converted to a vector representation, and the circuit can be stored, transferred and modified in an electronic format. The remaining step is to convert the vector representation to the format of the target simulation software. In electronics software packages, this is

usually done through converting first to a standard data format. The most common standard is Electronic Design Interchange Format (EDIF) [1]. This format allows communication with a wide variety of Electronic Design Automation (EDA) vendors.
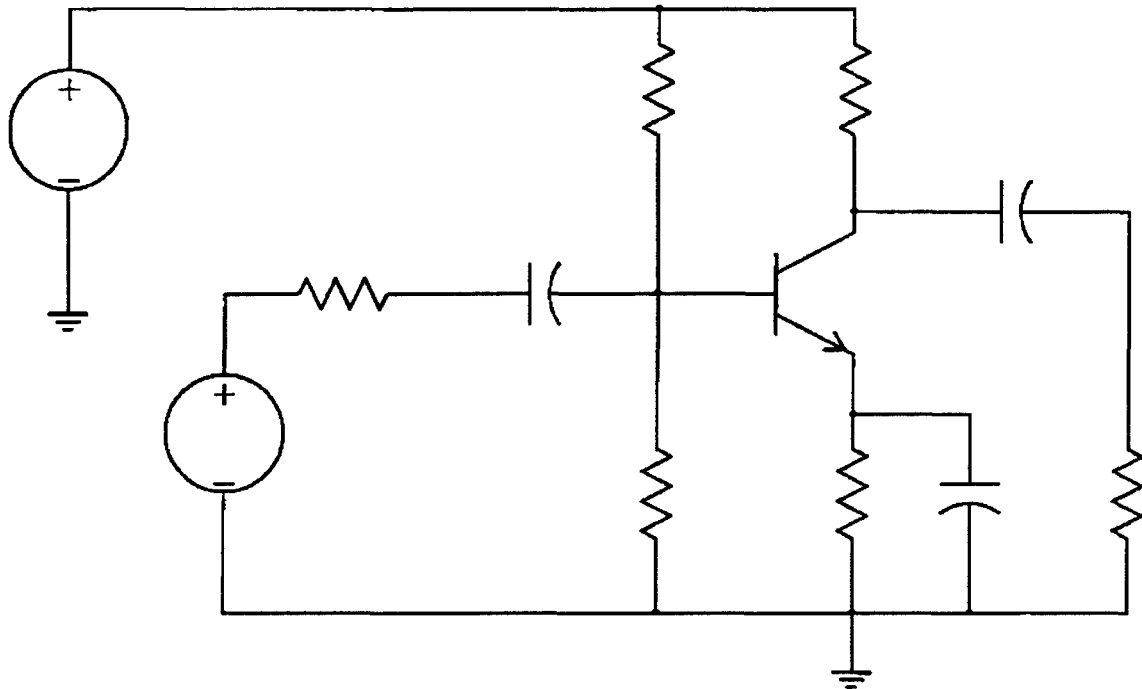
Figure 3. Circuit diagram of Figure 2 after text is extracted.

One important feature of vector formats is that they are stored in ASCII format, which allows great portability of the files. This is true whether considering the output format of the RVC software, EDIF format, or the target simulation software. Figure 5 shows partial file contents of the output format of AUDRE [2], a prominent RVC software package, the same schematic in EDIF form, and the same schematic in the format used by Analogy [3], a highly used analog simulation package, respectively.

Requirements for effective RVC software

The overarching requirement for useful automatic RVC of electrical schematics is a software package that has features dedicated to such diagrams. Circuit diagrams have a particular style that must be considered if the RVC is able to put the results in an appropriate and accurate form suitable for simulation. As stated above, an electrical schematic is a combination of three layers: text, symbols and wires. In this section, the requirements for successful RVC are presented for these three layers.
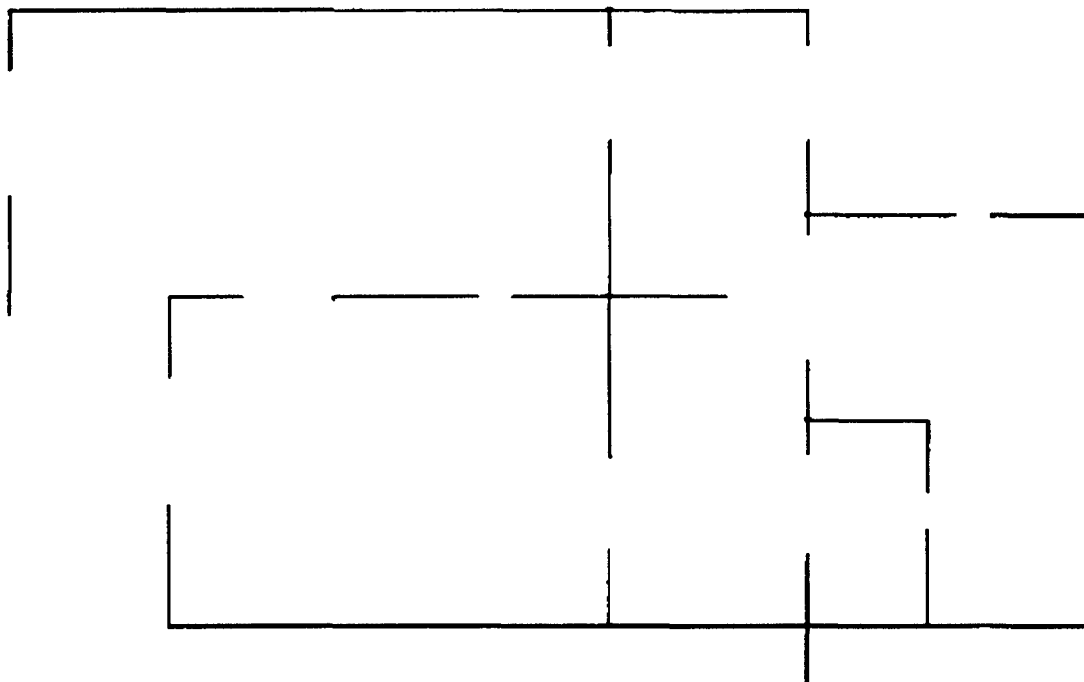
Figure 4. Circuit diagram of Figure 3 after symbols are extracted. Only the wires remain.

(a)	(b)	(c)
1762.000 1275.000\C3\7 1 0 0.000	(commentGraphics	use r_9_ -96 160 0 0 { } r r Default {
1251.000 1275.000\R1\7 1 0 0.000	(annotate	{ref R1 { } string { } { } { } 2 0 { } { } { } { } { } { }
1571.000 1275.000\Re\7 1 0 0.000	(stringDisplay "R1"	{ } 2}
1252.000 1324.000\10k\7 1 0 0.000	(display	{mom 10k { } string { } { } { } { } 0 { } { } { } { }
1571.000 1324.000\1.3k\7 1 0 0.000	(figureGroupOverride TEXT	{ } { } { } 2}
	(color 0 100 0))	} { }
	(origin (pt 568 184))))))	use r_10_ 64 480 0 0 { } r r Default {
	(commentGraphics	{ref Rc { } string { } { } { } 2 0 { } { } { } { } { } { }
	(annotate	{ } 2}
	(stringDisplay "Re"	{mom 4.3k { } string { } { } { } { } 0 { } { } { } { }
	(display	{ } { } { } 2}
	(figureGroupOverride TEXT	} { }
	(color 0 100 0))	
	(origin (pt 728 184))))))	

Figure 5. Partial portions of three different ASCII files representing different ways to describe the circuit of Figure 2: (a) AUDRE, (b) EDIF, and (c) Analogy. In all three files, the string R1 is indicated.

The text found on a circuit diagram varies significantly from other forms of text, but is formatted in typical patterns. First, component names usually follow standard notation. For example, a resistor will usually be labeled with an R followed by a number or letter, sometimes subscripted. Second, component values are numbers sometimes followed by prefixes and units. Thus, 10K, 10KΩ, and 2pF are all recognizable formats for component values. Third, part numbers are a combination of letters and numbers such as 2N3904 for a transistor. In all these cases, character recognition that uses an English dictionary to suggest words is a waste of effort. Instead, a dictionary of typical component designations, values and part numbers should be available that the software consults first. The user must be able to customize the dictionary for a particular set of diagrams.

Symbol recognition is arguably the most important aspect of effective RVC. In ASR, the software matches symbols present in the raster with templates in the user library to extract the symbols. Therefore, the user must be able to quickly build an accurate template library for these components. Once again, the software must be geared towards circuits for successful automatic symbol recognition and take advantage of the expected presence and typical shapes of all common components. The symbols in circuit diagrams fall into three categories. Hollow symbols are components that are outlined, such as the triangle of an amplifier. Solid symbols have no white space inside, such as diodes and resistors. Finally, there are multi-part symbols, such as capacitors and grounds. Each category presents problems for effective ASR. For example, the three most common elements, the capacitor, the node, and the resistor, provide particular problems: the resistor (a solid symbol) is the same width as the wires, the node (another solid symbol) is not much larger than the intersection of two wires, and the capacitor (a multi-part symbol) must be recognized as a single symbol. Hollow symbols present a different problem. The raster image might contain breaks in a hollow symbol which must be repaired before it can be recognized. Successful ASR extracts all but a few components. For those components that cannot be automatically recognized, manual insertion must be quick and accurate. Once a symbol is inserted, two events must occur. First, the symbol must retain its connectivity with the wires. Second, all raster of the component must be cleanly removed.

The last step in extraction is geometric recognition of the wires. While the most straightforward of the extraction steps, there are several requirements for the software. First, as in the case of hollow symbols, broken wires must be repaired. Wires with one or two pixel breaks in them are never intentionally drawn that way. If those breaks can be filled in, the GR process will be more accurate. Secondly, the software must recognize that in circuit diagrams wires connect components together. These wires must connect with the symbols extracted in the previous steps.

The overall extraction process must be easily set up to be completed in background on a computer. This allows large numbers of circuit diagrams to be handled efficiently. Finally, the user must be able to quickly create templates for any new symbols encountered and manually insert symbols the ASR process missed.

10-8

Ideally, the ASR software will automatically convert the raster to a vector format recognized by the target simulation software. If this is not possible, the final step is to convert the ASR software format to the target format. This involves mapping the symbols and their properties of the diagram with the schematic capture portion of the simulation software. The fact that these formats are ASCII texts suggests that conversion from one vector format to another is merely comprised of two parts: (1) understanding both the source and target formats, and (2) writing software that converts text strings.

Conclusion

Raster-to-vector conversion programs were originally designed for geological maps. Converting electrical schematics from raster to vector has been an additional use for such software. Therefore, it should not be surprising that these software packages are not yet optimized for schematics. Until a program becomes available that is fine-tuned for converting circuit diagrams, the effort of using RVC programs to convert such diagrams is not worthwhile: it is quicker to manually re-enter the circuit design directly into the target simulation program. However, it is not difficult to envision that that day is not too far in the future.

Acknowledgment

The author would like to thank Ed Kincaid of the Technology Insertion Branch of OC-ALC at Tinker AFB for his support and assistance on this project

References

[1] A brief overview can be found in: John P. Eurich and Gene Roth, "EDIF grows up," *IEEE Spectrum*, pp 68-72, Nov. 1990.
[2] AUDRE is a registered trademark of AUDRE, Incorporated (San Diego, CA).
[3] Analogy is a registered trademark of Analogy, Incorporated (Beaverton, OR).

A PROBABILISTIC FRAMEWORK FOR THE ANALYSIS
OF CORROSION DAMAGE IN AGING AIRCRAFT

Paul W. Whaley
Professor of Mechanical Engineering

Oklahoma Christian University
2501 East Memorial Road, P. O. Box 11000
Oklahoma City, OK 73136-1100

A PROBABILISTIC FRAMEWORK FOR THE ANALYSIS
OF CORROSION DAMAGE IN AGING AIRCRAFT

Paul W. Whaley
Professor
Department of Mechanical Engineering
Oklahoma Christian University

Abstract

Damage tolerance is the design philosophy used by the Air Force to manage the structural integrity of aircraft. The damage tolerance design philosophy assumes that cracks are always present in all aircraft structures, although they may be so small that they can not be detected. NonDestructive Inspection (NDI) techniques are utilized to measure the crack length. Analysis of crack growth is used to predict the number of fatigue cycles until the longest crack grows to half its critical length. This procedure is deterministic although there is significant uncertainty in the crack growth rate and in the detection of cracks.

The presence of corrosion in aging aircraft makes damage tolerance analysis more difficult. NDI procedures have been developed to detect cracks; detecting and quantifying corrosion is not well understood. The most common way to account for the effects of corrosion is to estimate the equivalent material thickness loss and then adjust the stress distribution accordingly. However, corrosion is nonuniform as verified by inspections of a number of corroded structural members from aircraft undergoing Programmed Depot Maintenance. Although some mild pitting corrosion might be approximated by equivalent material thickness loss, in the case of severe pitting and exfoliation unacceptable errors may be introduced by this approximation.

Sample mean and standard deviation of the crack growth constant were determined from regression analysis of crack growth data for several aluminum alloys and corrosion conditions. The data were collected during US Air Force sponsored round robin testing conducted at four different laboratories. Variability in the crack growth was modeled by variation in the Paris crack growth constant. The crack growth rate was then numerically integrated to estimate the mean and standard deviation of the crack length. The reliability, defined as the probability that the crack length exceeds the critical crack length, was calculated as a function of loading cycles. Results show that although the crack growth rates for corroded material falls within the scatter bands from traditional damage tolerance analysis, corrosion has a very distinctive effect on the reliability function. Reliability analysis is proposed as a suitable design tool for damage tolerance of aircraft structures with hidden corrosion.

11-2

A PROBABILISTIC FRAMEWORK FOR THE ANALYSIS
OF CORROSION DAMAGE IN AGING AIRCRAFT

Paul W. Whaley

Introduction

Damage tolerance is the design philosophy used by the Air Force to manage aircraft structural integrity. Cracks are assumed to be present in all aircraft structures, although they may be so small that they cannot be detected. NonDestructive Inspection (NDI) techniques are utilized to measure the crack length. Analysis of crack growth is used to predict the number of cycles until the longest crack grows to half its critical length. This analysis is used to guide decisions concerning inspection intervals and maintenance requirements. This procedure is deterministic although there is significant uncertainty in the crack growth rate and in the detection of cracks by NDI. NDI procedures have been developed to detect cracks; procedures for detecting and quantifying corrosion are not available.

The presence of corrosion in aging aircraft makes damage tolerance analysis much more difficult. The most common way to evaluate the effect of corrosion is to estimate the equivalent material thickness loss and then adjust the stress distribution accordingly. Based on inspections of a number of corroded structural members from KC-135 aircraft undergoing Programmed Depot Maintenance (PDM), corrosion is clearly non-uniform. Although some mild pitting corrosion might be approximated by equivalent material thickness loss, in the case of severe pitting and exfoliation unacceptable this approximation might introduce unacceotable errors.

Corrosion is assumed to be a random process that affects the initial defect size, random crack growth and fracture toughness. The mechanism of corrosion and its interaction with the grain structure is not well understood. For metals, the grain structure controls the formation of microvoids at the grain boundaries. These "microcracks" grow and coalesce until cracks that can be detected by NDI are formed (Fine, et. al., 1992). The initial crack size is randomly distributed and identifying the probability distribution function is a critical task (Trantina and Johnson, 1983). Crack growth rate curves always include significant scatter, so crack growth is assumed to be a random process. There is uncertainty in the fracture toughness for most alloys and accurate estimates of the mean and standard deviation are not available for all alloys.

Sample mean and standard deviation of crack growth rate data were determined from regression analysis for four aluminum alloys: 2024T3, 2024T4, 7075T6 and 7178T6. Five corrosion conditions were analyzed: dry as received, dry artificially corroded, wet as received, wet artificially corroded and 3.5% salt solution. The data were collected under contract to the Oklahoma City Air Logistics Center, Tinker AFB, OK by round robin testing conducted at four different laboratories (Luzar, 1997). Standard ASTM E647 Middle Tension Specimens were used.

The objective of this report is to devise a probabilistic framework for analysis of corrosion damage in aging aircraft. In the next section the analysis of crack growth sample mean and standard deviation for corroded and uncorroded material is described. That information is used to estimate the mean and standard deviation of the crack length and to estimate the reliability.

Random Crack Growth

Since the data appear to show the existence of a threshold stress intensity factor in many cases, the following form of the crack growth rate equation was used:

$$\frac{da}{dN} = A_o \left[(1-R)^q K_{max} - K_{th} \right]^b \tag{1}$$

Variability of the crack growth rate was modeled as a random variation in the parameter, A_o. Taking the logarithm of equation (1) gives a form that can be easily analyzed by linear regression, including an estimate of the confidence intervals (Bendat and Piersol, 1971). The sample variance of the observed values of $\log\left(\frac{da}{dN}\right)_i$ about the predicted $\log\left(\frac{da}{dN}\right)_p$ is:

$$s_o^2 = \frac{\sum\limits_{i=1}^{N} \log\left(\frac{da}{dN}\right)_i - \log(A) - b \log\left[(1-R)^q K_{max,i} - K_{th}\right]}{N-2}$$

Table I shows the results of the regression analysis for the alloys and corrosion conditions considered in this paper. The mean of the regression analysis is A and the standard deviation is s_o.

In some cases the appropriate threshold stress intensity value, K_{th}, was estimated by engineering judgment, since the threshold corresponding to the minimum error did not always fit the data adequately. In most cases the minimum error did not change very much as K_{th} changed. The dry conditions listed in Table I were usually much less than 15% relative humidity and the wet conditions were usually much greater than 85% relative humidity (Luzar, 1997). The artificially corroded specimens were based on ASTM B368.

The parameter, A_o, was assumed to follow a log-normal distribution (Ostergaard and Hillberry, 1983). This assumption is consistent with all of the crack growth data analyzed in this research. For a log-normal distribution, the 95% confidence intervals are:

$$A\left[(1-R)^q K_{max} - K_{th}\right]^b \Big/ \exp(2 s_o) \leq \frac{da}{dN} \leq A\left[(1-R)^q K_{max} - K_{th}\right]^b * \exp(2 s_o)$$

TABLE 1. RESULTS OF REGRESSION ANALYSIS FOR FOUR ALUMINUM ALLOYS AND FIVE CORROSION CONDITIONS

Alloy	Corrosion Condition	K_{th}	A	b	S_o
2024T3	As Received, <15%RH	4.529	1.346×10^{-7}	1.725	0.400
2024T3	Artificially Corroded, <15%RH	3.472	1.702×10^{-8}	2.566	0.622
2024T3	As Received, >85%RH	2.687	3.260×10^{-8}	2.361	0.225
2024T3	Artificially Corroded, >85%RH	1.997	5.714×10^{-8}	2.199	0.330
2024T3	3.5% Salt Solution	0.0	2.871×10^{-7}	1.433	0.194
2024T4	As Received, <15%RH	3.515	5.027×10^{-8}	2.078	0.487
2024T4	Artificially Corroded, <15%RH	2.691	1.333×10^{-8}	2.540	0.492
2024T4	As Received, >85%RH	2.014	1.827×10^{-8}	2.597	0.272
2024T4	Artificially Corroded, >85%RH	1.344	3.836×10^{-8}	2.252	0.288
2024T4	3.5% Salt Solution	0	2.170×10^{-7}	1.581	0.202
7075T6	As Received, <15%RH	3.846	1.500×10^{-7}	1.974	0.422
7075T6	Artificially Corroded, <15%RH	3.492	2.310×10^{-7}	1.872	0.407
7075T6	As Received, >85%RH	2.899	8.402×10^{-8}	2.525	0.351
7075T6	Artificially Corroded, >85%RH	2.502	2.212×10^{-7}	2.287	0.376
7075T6	3.5% Salt Solution	0.0	6.359×10^{-8}	2.472	0.471
7178T6	As Received, <15%RH	2.555	7.003×10^{-8}	2.229	0.415
7178T6	Artificially Corroded, <15%RH	1.741	3.198×10^{-8}	2.671	0.451
7178T6	As Received, >85%RH	1.449	2.391×10^{-8}	3.355	0.491
7178T6	Artificially Corroded, >85%RH	1.140	2.090×10^{-8}	3.466	0.433
7178T6	3.5% Salt Solution	0.0	2.924×10^{-9}	4.031	0.384

Figures 1-4 summarize crack growth rates for the alloys and corrosion conditions listed in Table I. Figure 1 shows crack growth rate for 2024T3, wet and dry conditions. Figure 1a suggests that for dry conditions, artificial corrosion does not significantly effect the crack growth rate. Even though the as received and artificially corroded cases have different confidence intervals in Figure 1a, the data appear to be superimposed. For the wet condition illustrated in Figure 1b, the effect of corrosion is to increase the mean crack growth rate. Figure 2 shows crack growth rate for 2024T4, wet and dry conditions. In Figure 2a the confidence intervals for as received and artificially corroded, dry 2024T4 specimens are essentially the same. Figure 2b suggests that the effect of corrosion in the wet condition is to increase the mean crack growth rate. Table I suggests that the effect of corrosion in 2024T3 and-T4 is to increase the width of the confidence intervals for the wet condition.

Figures 3 and 4 show crack growth for 7075T6 and 7178T6. The effect of corrosion is to increase the mean crack growth rate in both alloys, wet and dry conditions. Table I suggests that the effect of corrosion is to increase the width of the confidence intervals in 7075T6 and decrease the width of the confidence intervals in 7178T6.
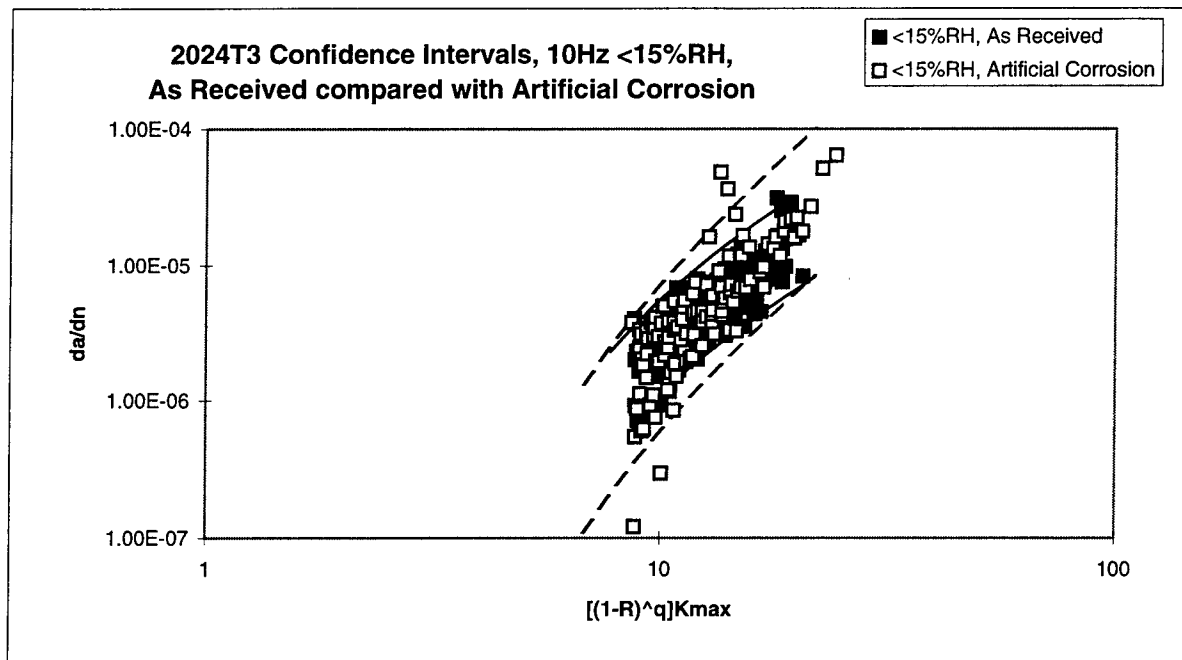
a. As Received and Artificially Corroded Dry Condition
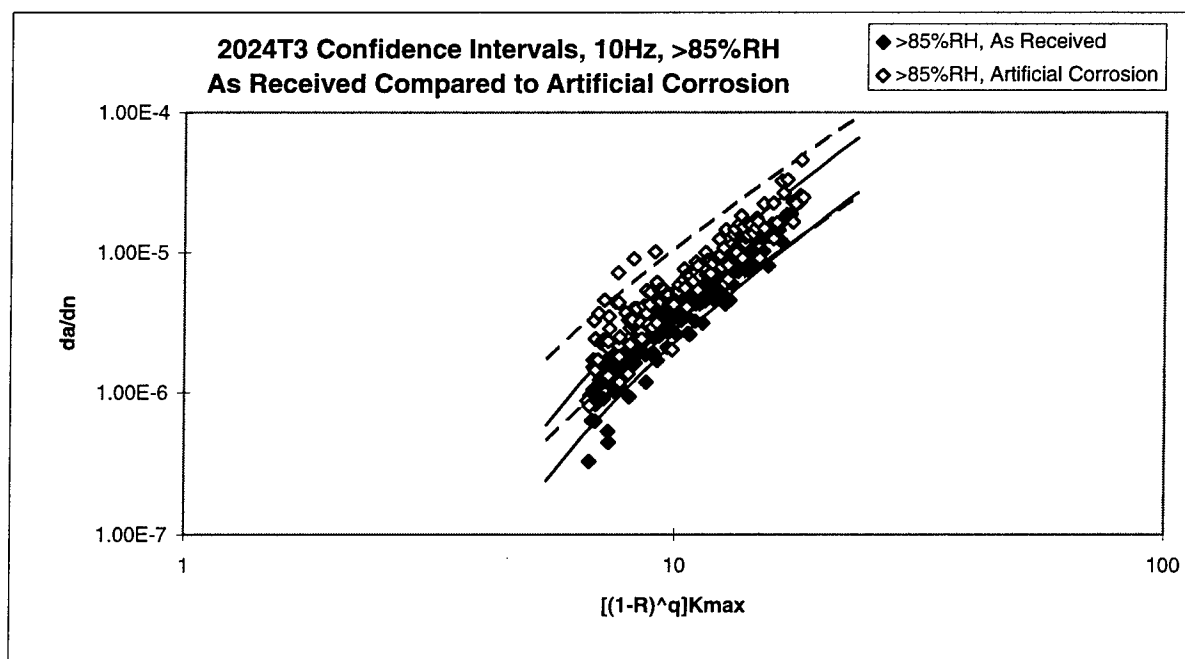
b. As Received and Artificially Corroded Wet Condition

Figure 1. Crack Growth Rate for Corroded and As Received 2024-T3 Aluminum With Confidence Intervals
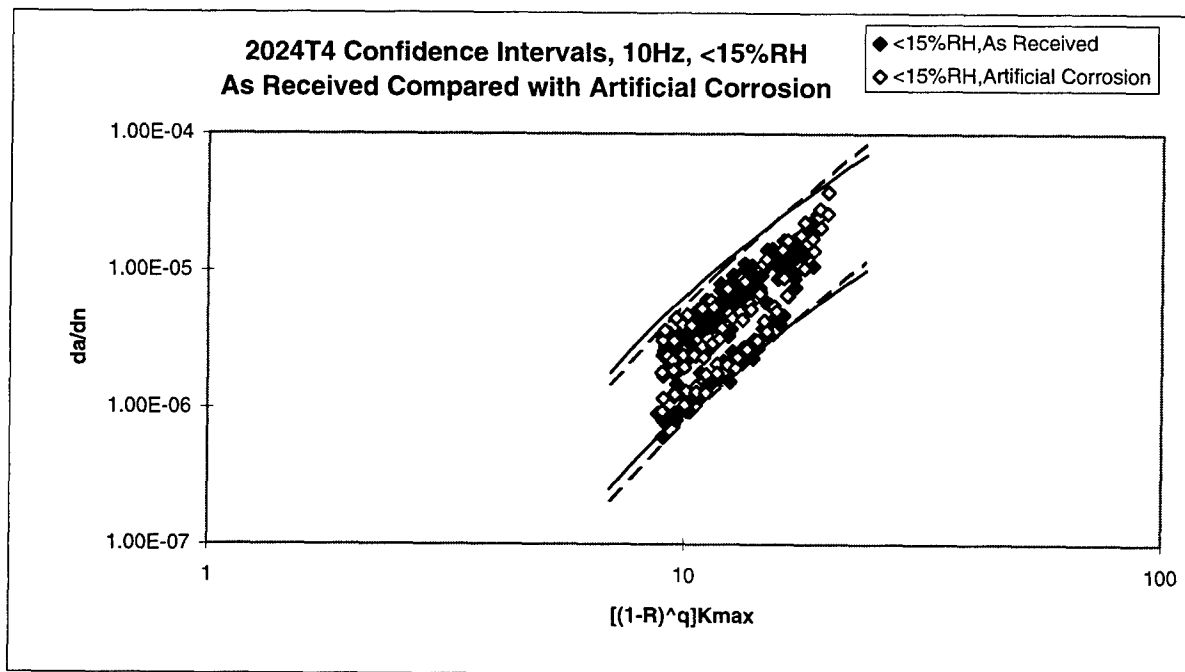
a. As Received and Artificially Corroded Dry Condition
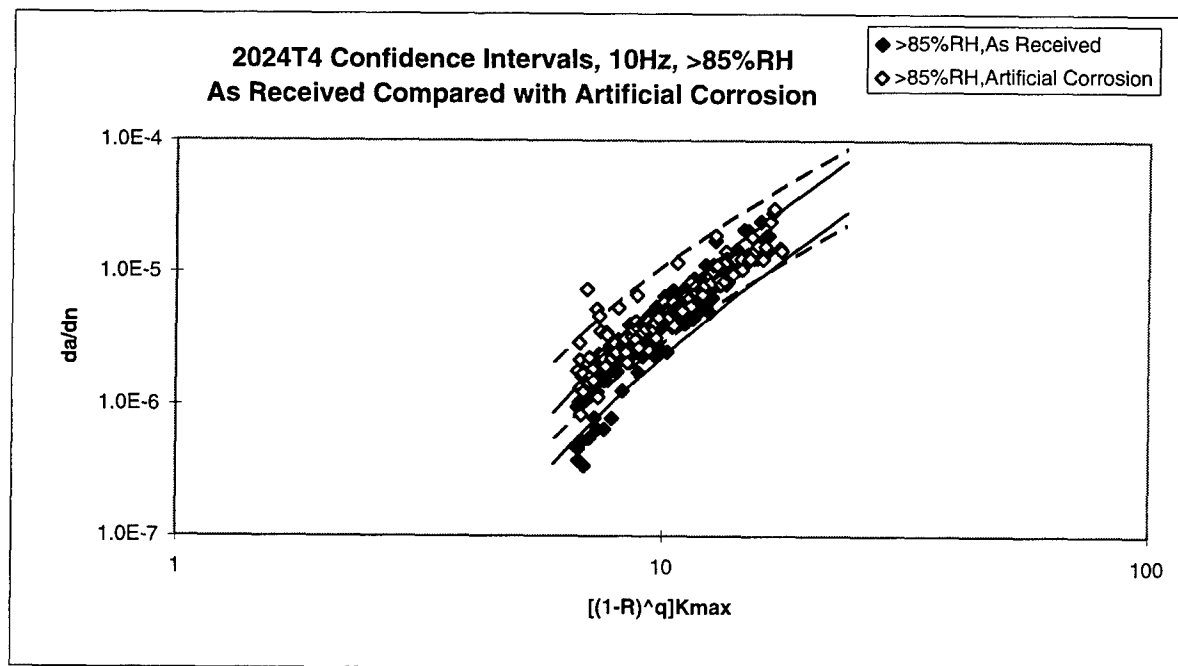
b. As Received and Artificially Corroded Wet Condition

Figure 2. Crack Growth Rate for Corroded and As Received 2024-T4 Aluminum With Confidence Intervals
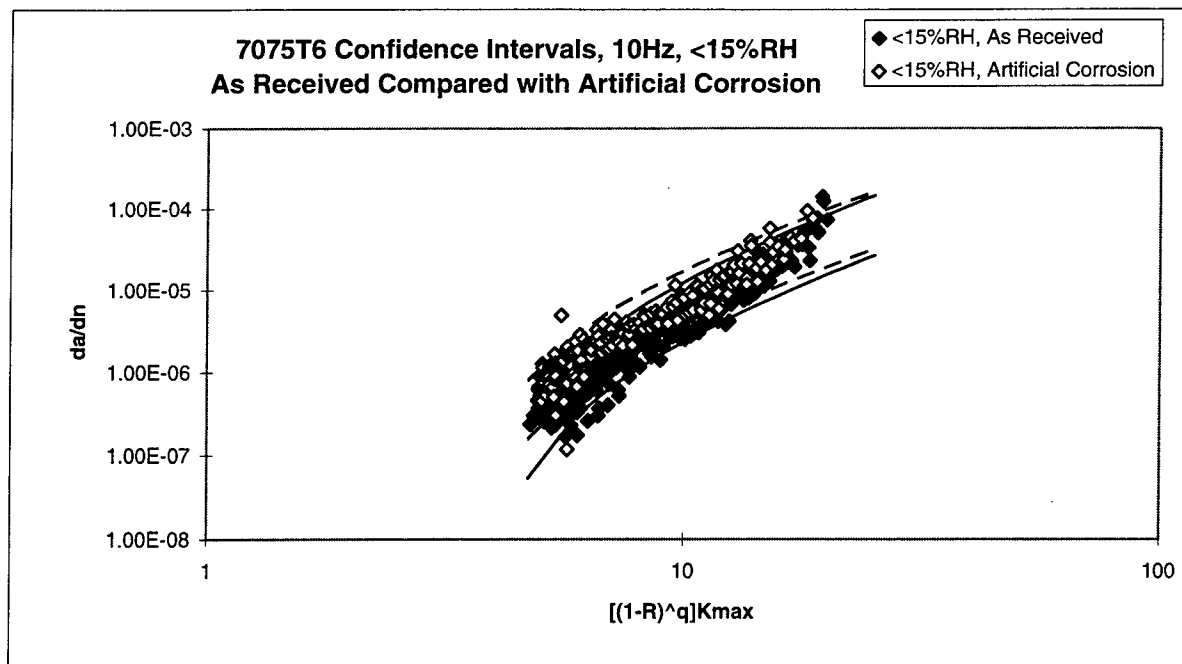
11-7

a. As Received and Artificially Corroded Dry Condition
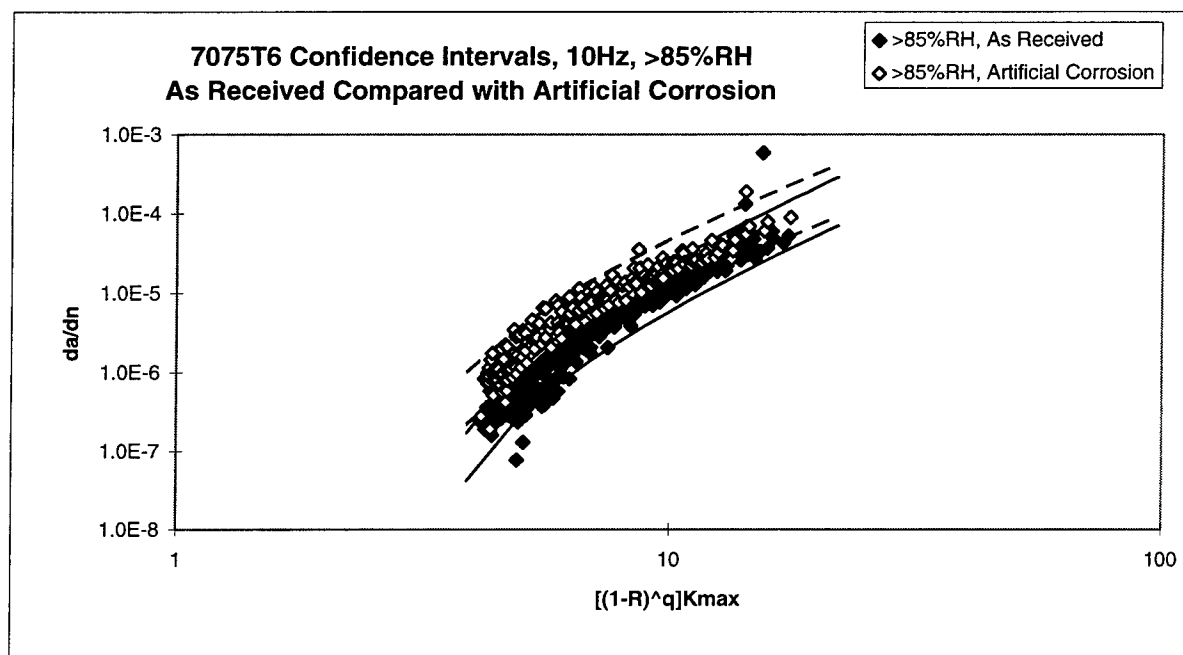
b. As Received and Artificially Corroded Wet Condition

Figure 3. Crack Growth Rate for Corroded and As Received 7075-T6 Aluminum With Confidence Intervals
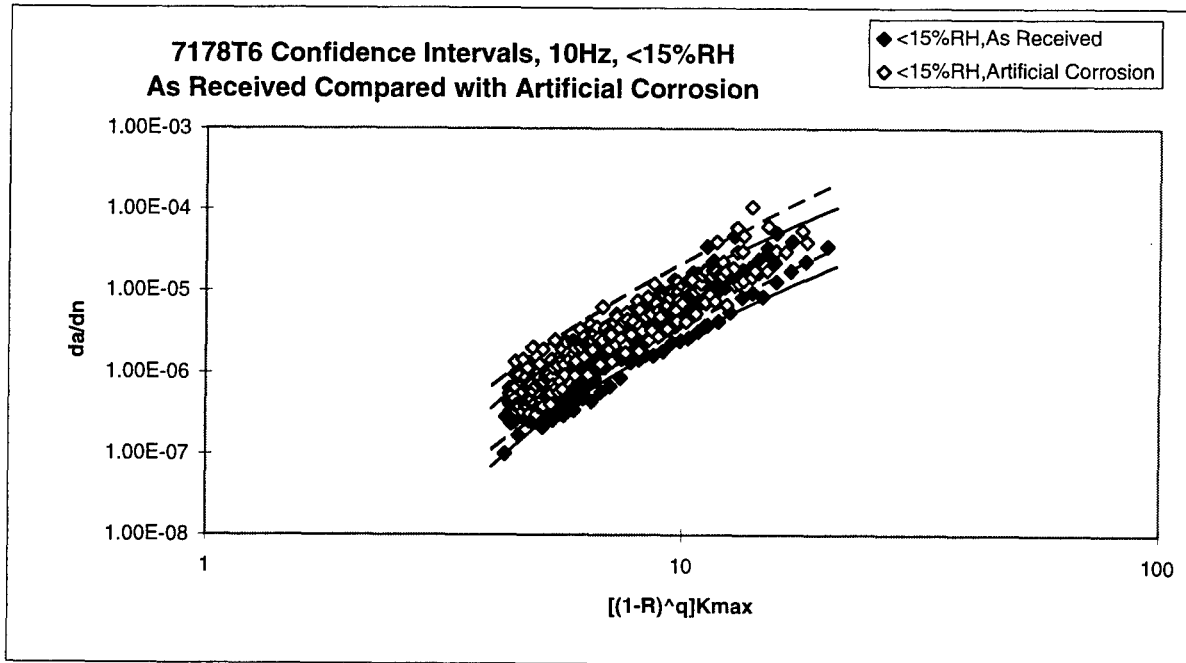
a. As Received and Artificially Corroded Dry Condition
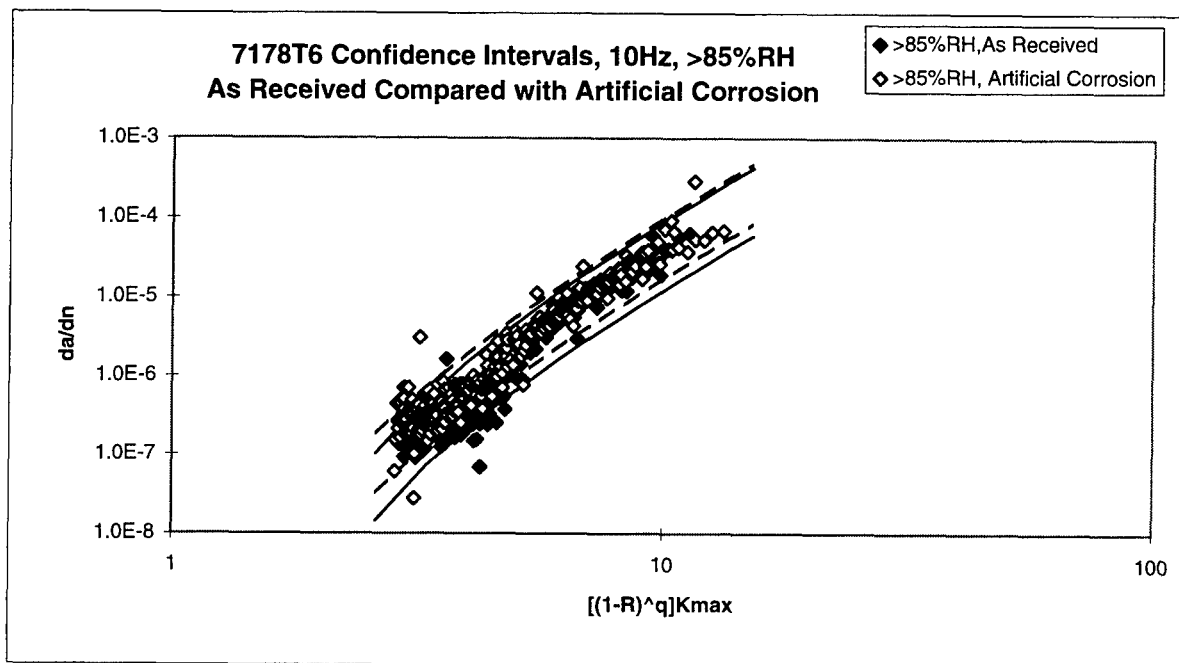
b. As Received and Artificially Corroded Wet Condition

Figure 4.   Crack Growth Rate for Corroded and As Received 7178-T6 Aluminum With
Confidence Intervals

Figures 1-4 indicate that crack growth rate is not strongly dependent on corrosion. There is significant overlap in the data scatter in all cases. Therefore, a probabilistic analysis of crack length as a function of fatigue cycles is needed to investigate the relationship between corrosion and fatigue damage. The mean and standard deviations of crack size are described next.

Probabilistic Analysis of Crack Size

In this section the mean and standard deviation of the crack size are estimated based on numerical integration of the crack growth rate. The crack growth equation is:

$$\frac{da}{dN} = A_o \left((1-R)^q \sigma_{max} \sqrt{\pi a / \cos\left(\frac{\pi a}{W}\right)} - K_{th} \right)^b \tag{2}$$

The classic Runga Kutta algorithm was used to numerically integrate equation 2. Monte Carlo simulation is sometimes used to generate a crack size population after N fatigue cycles. A random number generator provides sample values of A_o and then equation 2 is solved for each sample value to build a population of crack sizes. That approach is very computationally intensive since the number of cycles to the next inspection interval is a variable. Instead of applying Monte Carlo simulation, the mean and standard deviation of the crack size are estimated directly from knowledge of the mean and standard deviation of A_o.

The mean crack size is assumed to be the solution of equation 2 with A_o equal to the mean value, A, column 4 of Table I. The crack size corresponding to the mean plus the standard deviation is the solution of equation 2 with A_o equal to $A \exp(s_0)$. These two curves can be combined to approximate the standard deviation of the crack size, which is a function of the cycles of loading. It is useful to estimate the confidence intervals of the crack size. If the crack size is log-normally distributed, then the 95% confidence intervals are defined by the mean crack size $\pm$ two standard deviations. Typical predicted crack size versus fatigue cycles are summarized in Figures 5-8.

The following notion is used in Figures 5-8: the curve labeled 'a - 2 s.d.' is the crack size from solving equation 2 with A_o replaced by $A/\exp(2 s_0)$; the curve labeled 'a - 1 s.d.' is the crack size from solving equation 2 with A_o replaced by $A/\exp(s_0)$; the curve labeled 'a' is the crack size from solving equation 2 with A_o replaced by A; the curve labeled 'a + 1 s.d.' is the crack size from solving equation 2 with A_o replaced by $A \exp(s_0)$ and the curve labeled 'a + 2 s.d.' is the crack size from solving equation 2 with A_o replaced by $A \exp(s_0)$. The curves labeled 'a - 2 s.d.' and 'a + 2 s.d.' define the confidence intervals. Similar analysis was conducted for all the alloys and corrosion conditions listed in Table I. In all cases the measured crack size data falls within the predicted confidence intervals. These preliminary results suggest that a log-normal probability density function for the crack size gives consistent predictions of the crack growth.
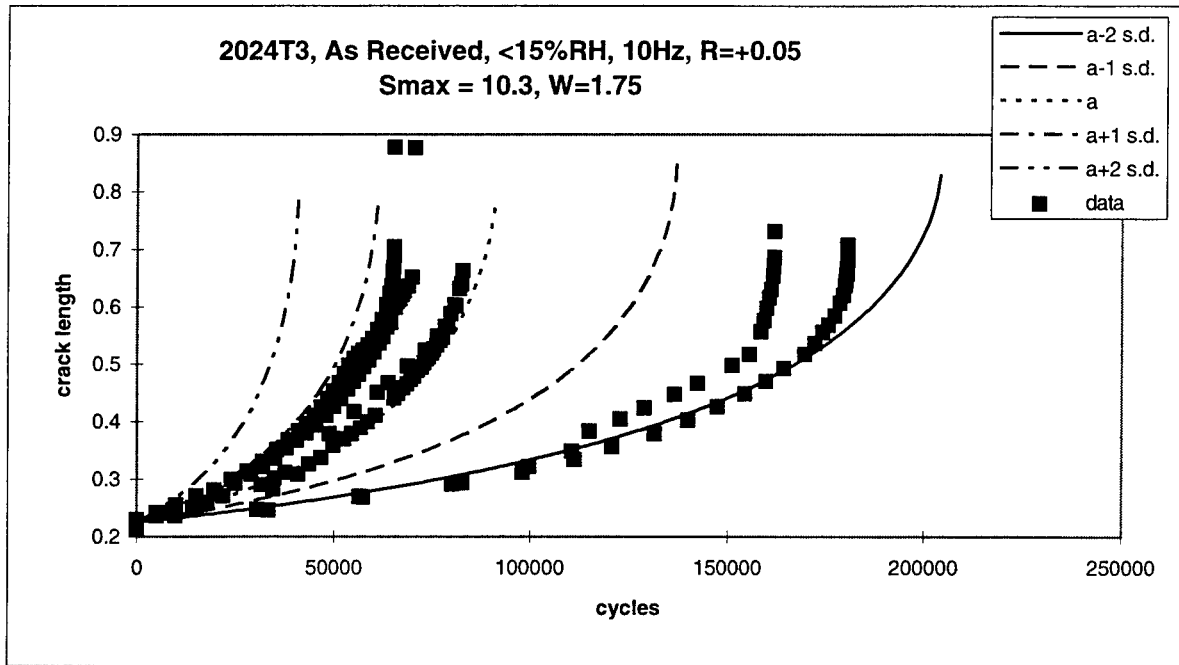
Figure 5. Predicted Crack Growth with Confidence Intervals for 2024T3 Aluminum.
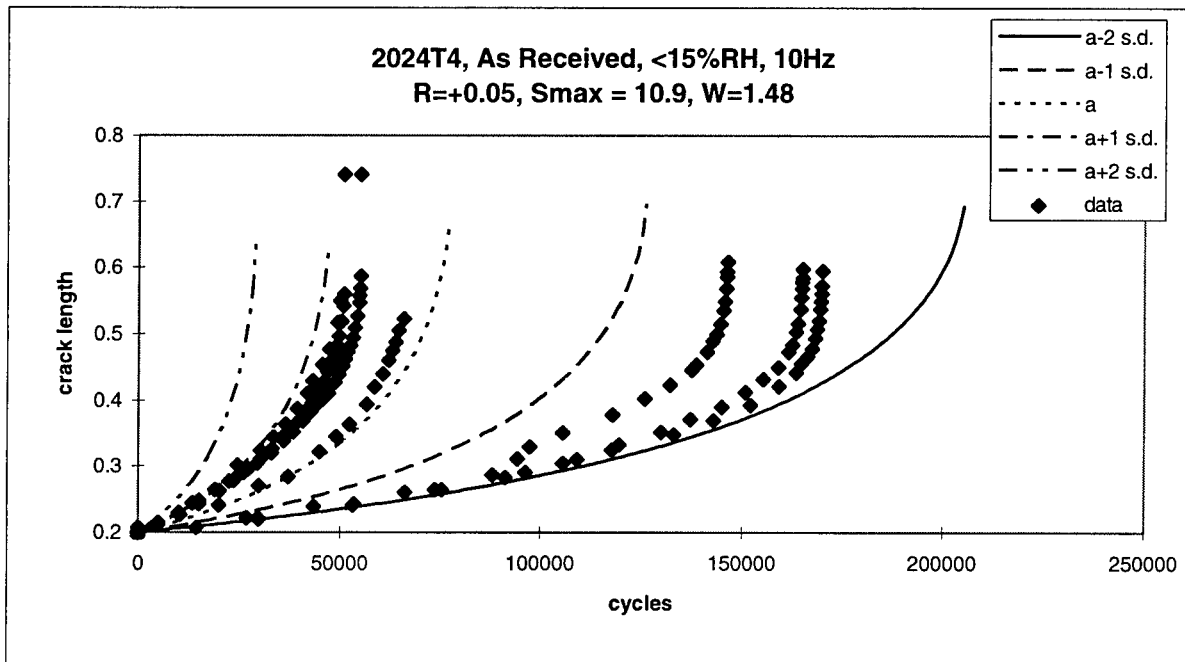
Figure 6. Predicted Crack Growth with Confidence Intervals for 2024T4 Aluminum.
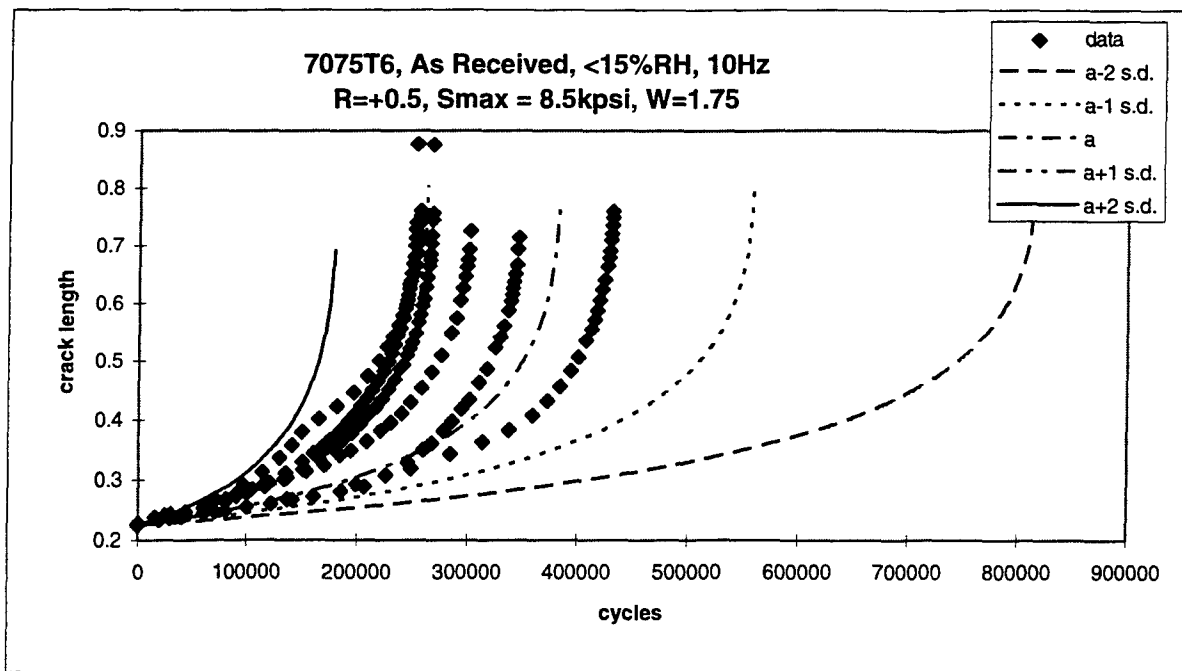
Figure 7. Predicted Crack Growth with Confidence Intervals for 7075T6 Aluminum.
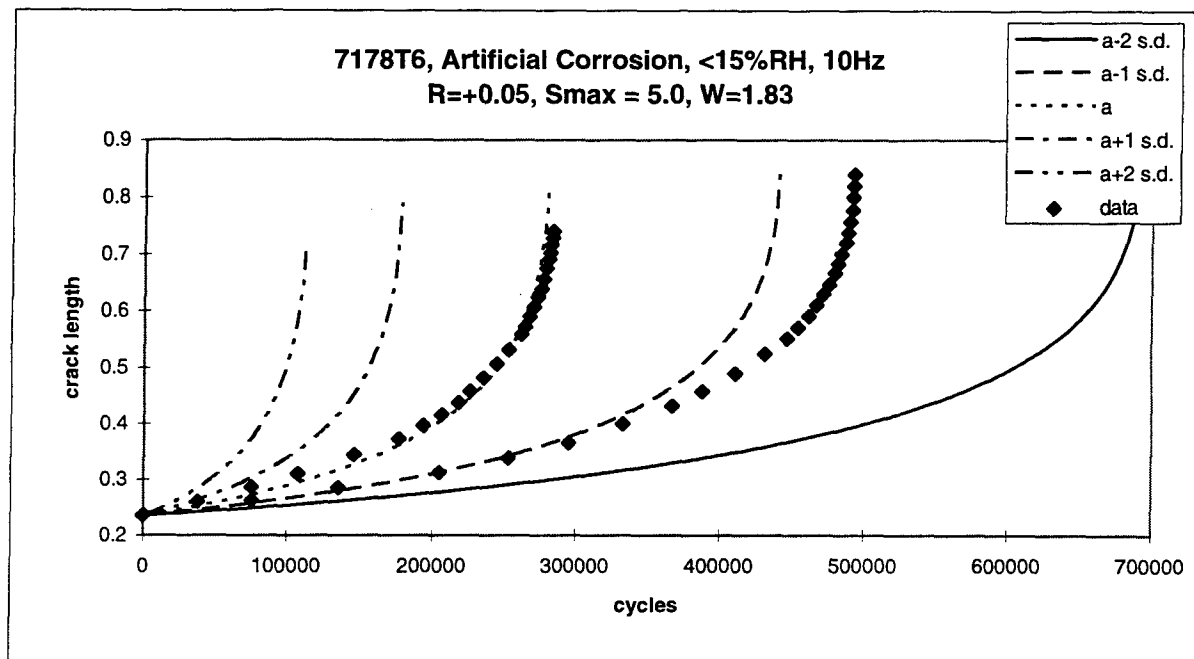
Figure 8. Predicted Crack Growth with Confidence Intervals for 7178T6 Aluminum.

Analysis of Reliability

Figures 5-8 reveal the random character of crack growth in the aluminum alloys considered in this paper. However, this information is not useful to the structural engineer who must predict fatigue life in the presence of corrosion. There is significant uncertainty in the crack size so a probabilistic analysis is needed.

Reliability is defined as the probability that the crack size is greater than the critical crack size. In this section the mean and standard deviation of the critical crack size are determined from the mean and standard deviation of the fracture toughness. The reliability function can be derived by combining the means and standard deviations of the crack length mean and the critical crack length. First, the critical crack length are determined from the fracture toughness. Then the mean and standard deviation of the critical crack length are determined from the mean and standard deviation of the fracture toughness.

Fracture toughness for the Aluminum alloys 2024T3, 2024T4, 7075T6 and 7178T6 were obtained from the Damage Tolerant Design Handbook, as summarized in Table II.

TABLE II. PLANE STRESS FRACTURE TOUGHNESS ESTIMATES
FOR 2024T3, 2024T4, 7075T6 AND 7178T6 ALUMINUM
WITHOUT BUCKLING CONSTRAINTS

Alloy, sheet	Thickness	K_{Ic}, mean $\left(\text{kpsi }\sqrt{\text{in}}\right)$	K_{Ic}, s.d. $\left(\text{kpsi }\sqrt{\text{in}}\right)$
2024T3	0.063	81.8	0
2024T4	0.064	30.0	0
7075T6	0.063	75.4	0.4
7178T6	0.12	38.8	3.8

For 2024T3 and -T4 the net section stress was greater than 80% of the yield strength in most cases. Therefore, the apparent fracture toughness was used for those alloys, resulting in a low estimate of fracture toughness. There was not enough data to calculate the standard deviation. For 7075T6 and 7178T6 the mean and standard deviations are not reliable because they were calculated from small sample sizes. This is not a serious limitation, since a parameter study shows that the standard deviation of the crack length has a stronger effect on the reliability than does the standard deviation of the critical crack length. Data for 7178T6 were were for wing skin 0.175 inch thick (Luzar, 1997). The thickness 0.12 in Table II was the closest from the Damage Tolerant Design Handbook.

The critical crack size is the solution to the transcendental equation:

$$\left(K_{Ic}/\sigma_{max}\right)^2 = \pi a_c \Big/ \cos\left(\frac{\pi a_c}{W}\right)$$

The relationships between the mean and standard deviation of a_c and the mean and standard deviation of K_{Ic} were determined following the development of Shigley and Mischke (1989).

$$\left(\frac{\mu_K}{\sigma_{max}}\right)^2 = \frac{\pi\mu_{a_c}}{\cos\frac{\pi\mu_{a_c}}{W}} \qquad \sigma_{\overline{a}} = \frac{2\sigma_K\mu_K\cos^2\frac{\pi\mu_{a_c}}{W}}{W\sqrt{\cos^2\frac{\pi\mu_{a_c}}{W}+\left(\frac{\pi\mu_{a_c}}{W}\right)^4+\frac{1}{36}\left(\frac{\pi\mu_{a_c}}{W}\right)^8}}$$

The mean critical crack length is easily calculated by Newton-Raphson iteration. The standard deviations of 7075T6 and 7178T6 using the data in Table II were very small.

In the previous section consistent predictions of crack size were based on the assumption that the crack size is distributed log-normally. The critical crack length is also assumed to be log-normally distributed. This assumption is acceptable because the standard deviations of the critical crack size are so small. Reliability was calculated based on log-normal interference following the development by Shigley and Mischke (1989). The mean and standard deviations of the crack length and critical crack length were transformed to a univariate normal distribution with unity variance by the coupling equation:

$$z_c = -\log\left[\frac{\mu_{a_c}\sqrt{1+\left(C_a\right)^2}}{\mu_a\sqrt{1+\left(C_{a_c}\right)^2}}\right]\Bigg/ \sqrt{\log\left[1+\left(C_{a_c}\right)^2\right]+\log\left[1+\left(C_a\right)^2\right]} \qquad (3)$$

The coefficient of variation, C, defined as the ratio of the standard deviation to the mean, was used to simplify equation 3. The reliability was calculated by numerical integration using Gaussian quadrature:

$$R_c = \int_{z_c}^{\infty} p(\zeta)d\zeta$$

Figures 9-12 show reliability predictions of different corrosion conditions for the aluminum alloys considered in this paper. Figures 9 and 10 suggest that for 2024T3 and -T4, the reliability of artificially corroded material is greater than that of the as received material for dry conditions. That counter-intuitive result is not surprising since Figures 1a and 2a show very little difference between these two dry conditions. The dry condition is not a realistic field condition.

For all of the alloys there is a distinctive difference between as received and artificially corroded, wet condition. This suggests that the probabilistic approach described in this paper is a useful tool for the analysis of crack growth in the presence of corrosion. For 2024T3, 2024T4 and 7075T6 the worst corrosion condition was the 3.5% salt solution, as indicated by increased crack growth rates at low stress intensity factors. Figures 9-11 show that the 3.5% salt solution also has the lowest reliability, while Figure 12 suggests that the wet, artificially corroded condition in 7178T6 has lower reliability than the 3.5% salt solution. Figure 13 is the crack growth rate in 7178T6 wet, artificially corroded condition compared with the 3.5% salt solution. The two crack growth rates fall within the same scatter band, giving an explanation for this result.
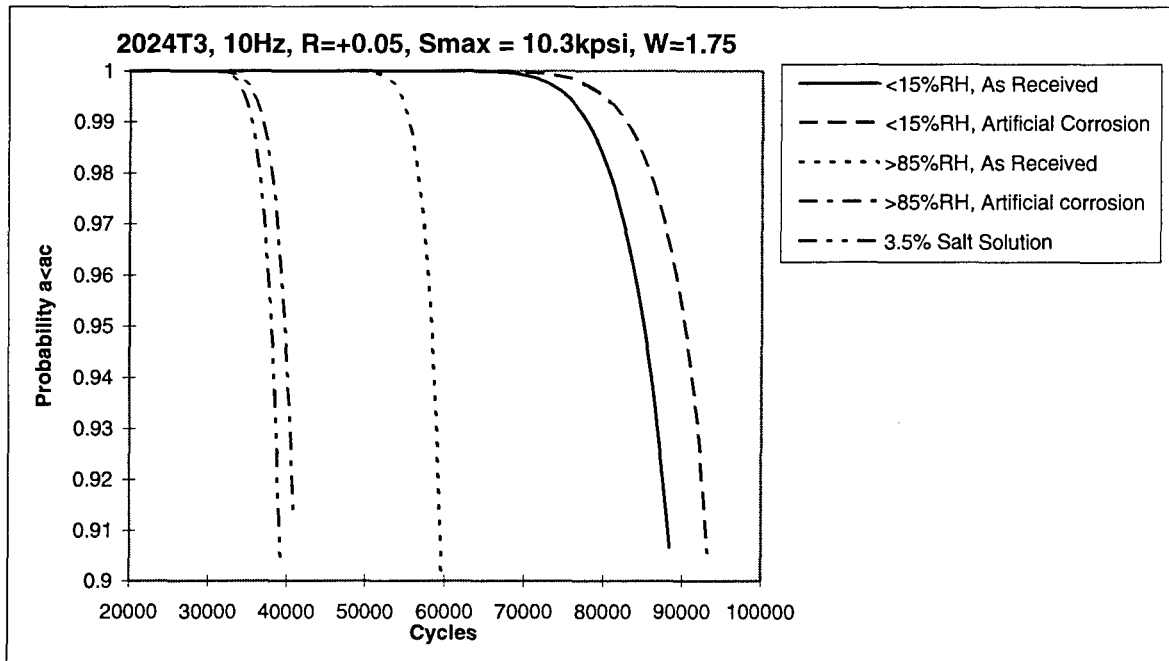
Figure 9. Reliability Analysis of 2024T3 Aluminum, Corroded Conditions
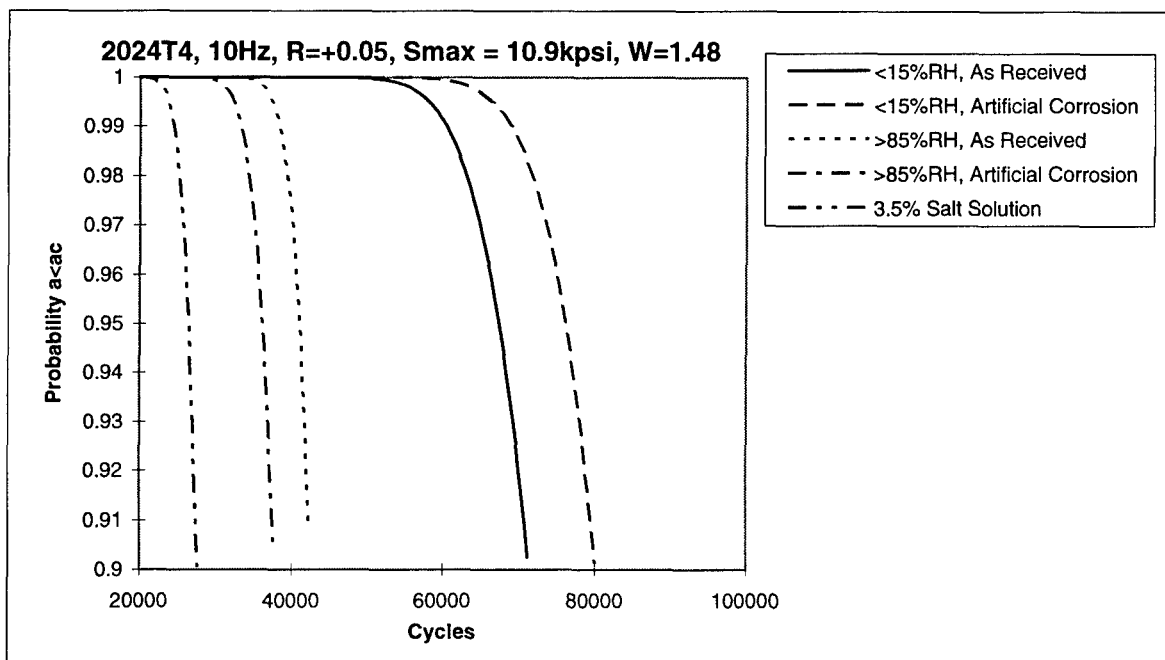
Figure 10. Reliability Analysis of 2024T4 Aluminum, Corroded Conditions
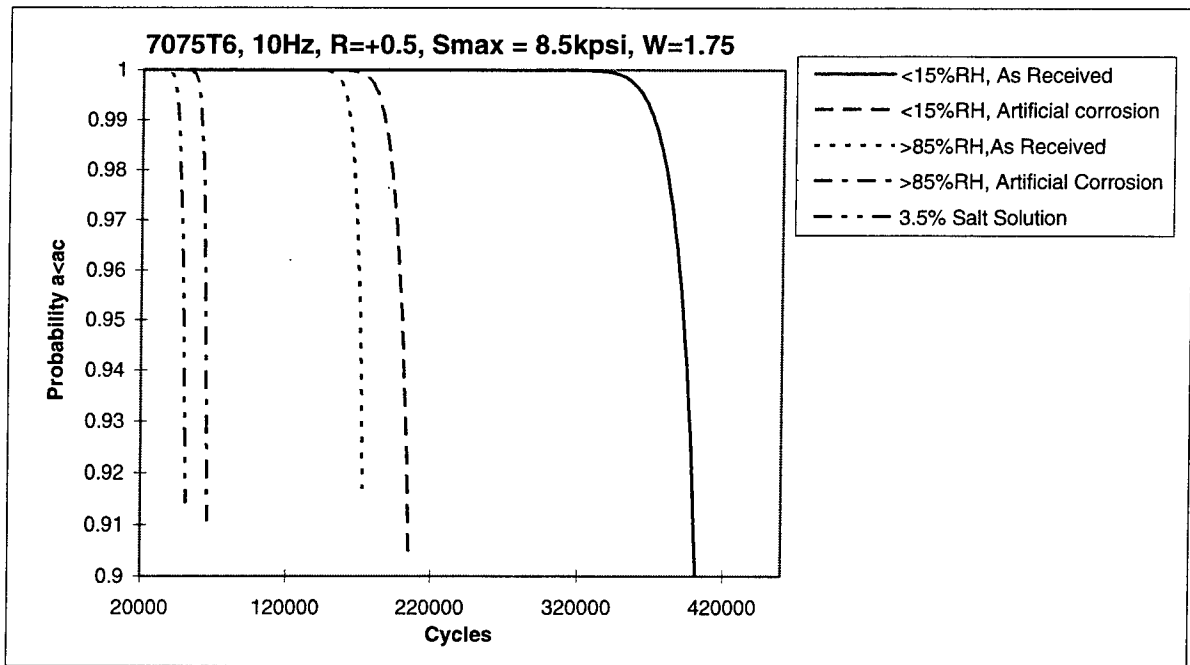
Figure 11. Reliability Analysis of 7075T6 Aluminum, Corroded Conditions
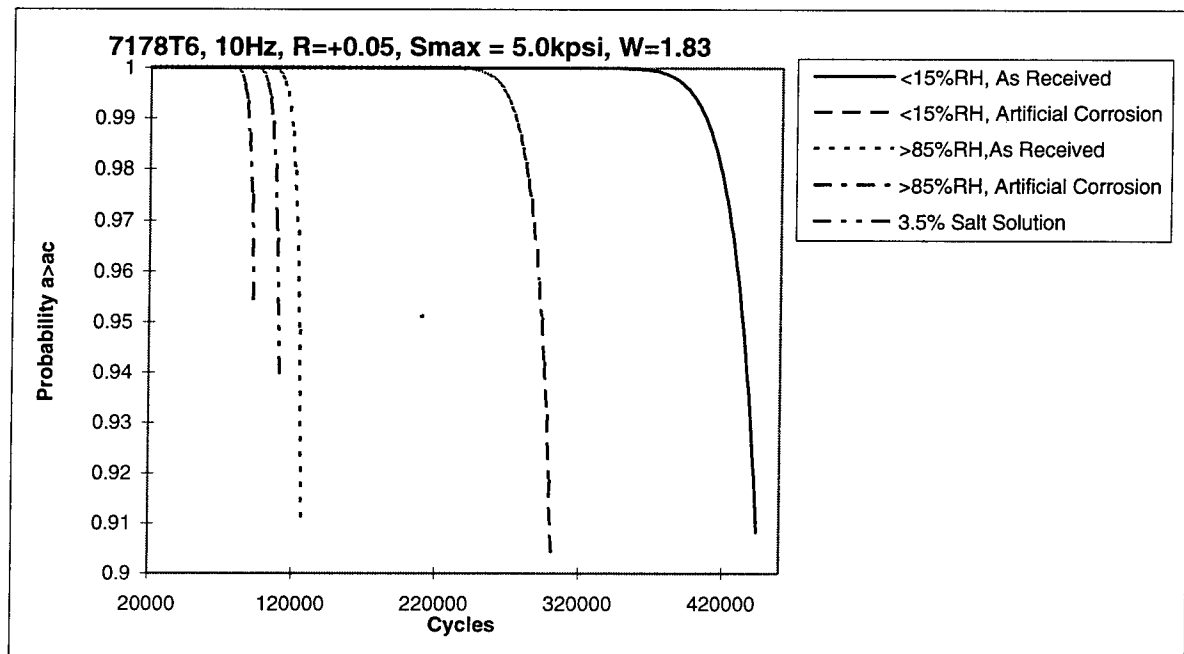
Figure 12. Reliability Analysis of 7178T6 Aluminum, Corroded Conditions

**7178T6 Confidence Intervals, 10Hz
>85%RH, Artificially Corroded
Compared with 3.5% Salt Solution**

Legend:
- ■ >85%RH, Artificial Corrosion
- O 3.5% Salt Solution

y-axis: da/dn (1.00E-3, 1.00E-4, 1.00E-5, 1.00E-6, 1.00E-7, 1.00E-8)
x-axis: [(1-R)^q]Kmax (1, 10, 100)
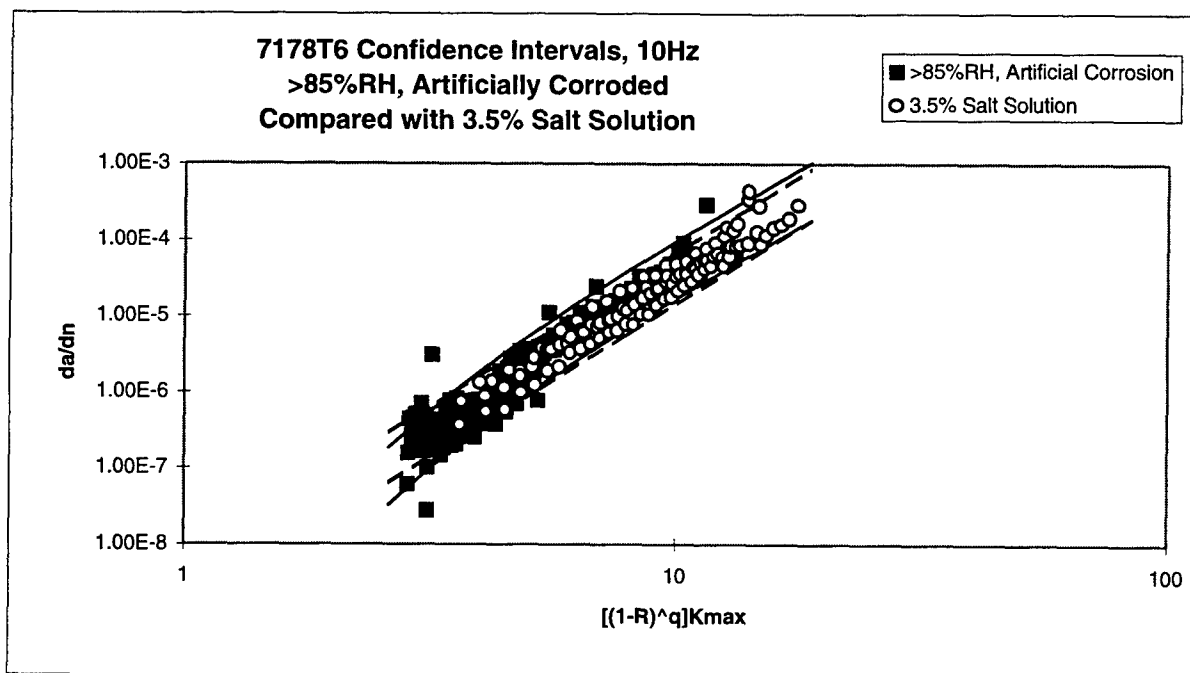
Figure 13. Crack Growth rates for 7178T6 Aluminum, wet Artificially Corroded Condition Compared with the 3.5% Salt Solution

It is clear from Figures 1-4 that there is no general crack growth rate trend in the presence of corrosion. In some stress intensity factor ranges the corroded crack growth is greater than the uncorroded, while in other ranges the corroded crack growth is smaller than the uncorroded. Still other corrosion conditions show no clear distinction in crack growth rate, as illustrated by Figure 13. In the case of 7075T6, artificial corrosion increases the crack growth as compared to the as received condition for both wet and dry conditions. The reliability analysis would have to be repeated for each alloy, corrosion condition, crack length and loading condition in order to predict fatigue life.

One way to account for the effects of corrosion is to calculate the equivalent material thickness loss and adjust the stress distribution accordingly. Figure 14 shows the results of that analysis for 7075T6, wet condition. Using the original material thickness, the artificially corroded condition shows a reduction in the fatigue lifetime by a factor of 2.7 compared to the as received condition. If 15% of the material is assumed uniformly corroded away the fatigue lifetime is reduced by a factor of 1.5, and if 20% of the material is assumed corroded away the factor becomes 1.2. If 25% of the material thickness is assumed corroded away, the artificially corroded reliability prediction is super-imposed on the as received condition. The measured equivalent material thickness loss was between 12% and 16% (Luzar, 1997). This analysis suggests that equivalent material thickness loss does not adequately account for the effects of corrosion.

For PDM applications adjusting the stress distribution using an equivalent material thickness loss does not make sense. Accurate measurements of thickness loss are difficult because corrosion is non-uniform. Also, when corrosion is found during PDM it is repaired. The purpose of fatigue

life predictions in the presence of corrosion is to account for any undetected corrosion that may still be in the aircraft.
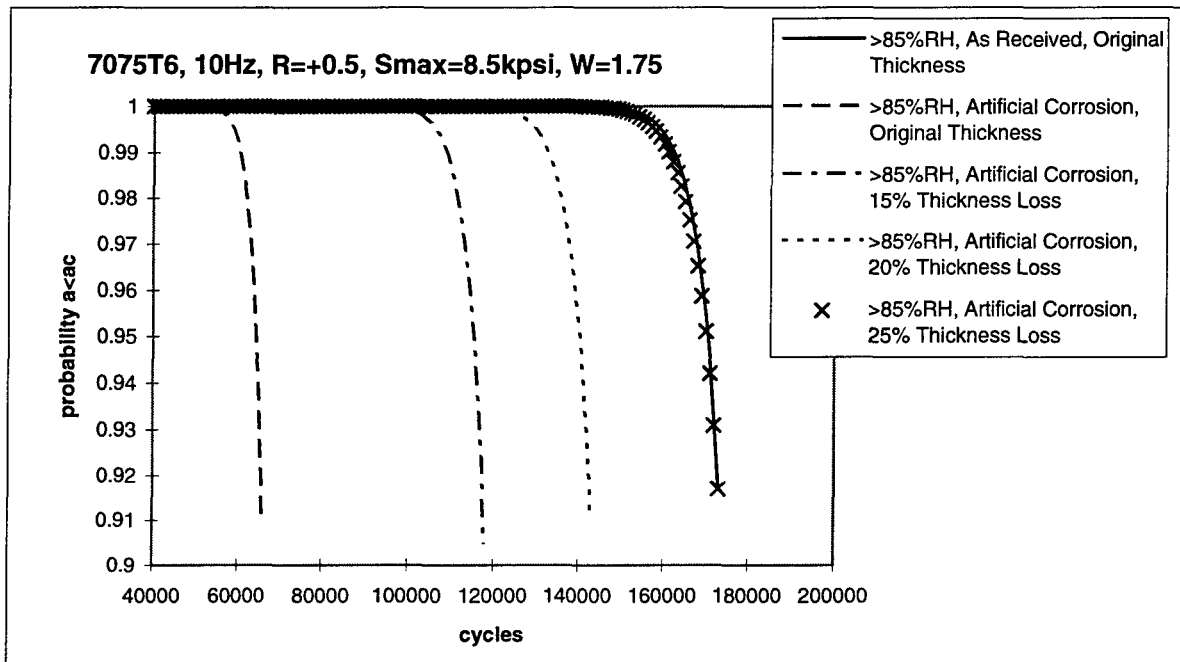
Figure 14. Reliability Analysis of 7075T6 Aluminum Showing the Effects of Equivalent Material Thickness Loss

Recommendations and Conclusions

Crack growth was modeled as a random process in this paper. The mean and standard deviation of the crack growth constant were determined by linear regression analysis. Assuming a log-normal distribution, the 95% confidence intervals can be approximated by the mean crack growth ± two standard deviations. This assumption was consistent with all the measured crack growth rate data.

Crack length was calculated by direct numerical integration of the crack growth rate. The mean and standard deviation of the crack length were estimated directly from the mean and standard deviation of the crack growth constant. The crack size was assumed to follow a log-normal distribution. This assumption was consistent with numerous measured crack growth curves. This analysis suggests that the standard deviation of the crack size increases as fatigue cycles increases. Additional testing is needed to identify the probability density function of the crack size.

The mean and standard deviation of critical crack size were determined from the fracture toughness in order to estimate the reliability. The critical crack size was assumed to follow the log-normal distribution. In this research, the standard deviation of the critical crack size was so small that its effect was secondary. A parameter study suggests that the fracture toughness standard deviation would have to be greater than about 20% of the mean in order to have a

significant effect on the reliability. More fracture toughness data is needed to evaluate this assumption.

This analysis assumed a known initial crack length. In practice, the initial crack length is expected to be random as well. More work is needed to identify the initial defect distribution in common alloys. This task is difficult because of the limitations of NDI and also because of the need to model entire populations of cracks and not just the longest one. The Weibull probability density function has been proposed for initial crack length distribution in nickel-base superalloys (Trantina and Johnson, 1983).

This research suggests that reliability analysis provides a suitable framework for predicting fatigue lifetime reduction due to undetected corrosion in old aircraft. In order to provide a useful design tool, it is necessary to measure crack growth rate and fracture toughness in corroded material removed from old aircraft.

Acknowledgements

References

Bendat, J. S., and A. G. Piersol, *Random Data: Analysis and Measurement Procedures*, Wiley-Interscience, New York, 1971, p. 131.

Fine, M. E., C. Y. Kung, M. I. Fadragas and J. D. Achenbach, "Fatigue Crack Initiation and Microcrack Propagation Data Base in Precipitation Hardened Aluminum Alloys," *Durability of Metal Aircraft Structures*, Proceedings of the International Workshop on Structural Integrity of Aging Airplanes, March 31-April 2, 1992, Atlanta, GA, Atlanta Technology Publications.

Gruff, J. J., and J. G. Hutcheson, "Effects of Corrosive Environments on Fatigue Life of Aluminum Alloys Under Maneuver Spectrum Loading," Proceedings of the Air Force Conference on Fatigue and Fracture of Aircraft Structures and Materials, Miami Beach, Florida, 15-18 December, 1969, AFFDL TR 70-144.

Harmsworth, C. L., "Effect of Corrosion on the Fatigue Behavior of 2024-T4 Aluminum Alloy," ASD Technical Report 61-121, July, 1961.

Johnson, G. O., "Statistical Scatter in Fracture Toughness and Fatigue Crack Growth Rate Data," *Probabilistic Fracture Mechanics and Fatigue Methods: Applications for Structural Design and Maintenance*, ASTM STP 798, J. M. Bloom and J. C. Ekvall, Eds., American Society for Testing and Materials, 1983, pp. 42-66.

Lee, E. U., "Corrosion Fatigue and Stress Corrosion Cracking of 7475-T7351 Aluminum Alloy," *Corrosion Cracking*, Proceedings of the Corrosion Cracking Program and Related Papers presented at the International Conference and Exposition on Fatigue, Corrosion Cracking, Fracture Mechanics and Failure Analysis, V. S. Goel, Ed., 2-6 December 1985, Salt Lake City, UT, pp. 123-128.

Luzar, J., "Corroded Material Crack Growth Rate Test Results," Proceedings, The First Joint DOD/FAA/NASA Conference on Aging Aircraft, 8-10 July, 1997, Ogden, UT.

Ostergaard, D. F. and B. M. Hillberry, "Characterization of the Variability in Fatigue Crack Propagation Data," *Probabilistic Fracture Mechanics and Fatigue Methods: Applications for Structural Design and Maintenance*, ASTM STP 798, J. M. Bloom and J. C. Ekvall, Eds., American Society for Testing and Materials, 1983, pp. 97-115.

Schutz, W., "Corrosion Fatigue - The Forgotten Factor in Assessing Durability," 15th Plantema Memorial Lecture, Proceedings, 18th Symposium of the International Committee on Aeronautical Fatigue, Melbourne, May, 1995.

Shigley, J. E. and C. R. Mischke, *Mechanical Engineering Design*, fifth edition, McGraw-Hill, New York, 1989.

Trantina, G. G. and C. A. Johnson, "Probabilistic Defect Size Analysis Using Fatigue and Cyclic Crack Growth rate Data," *Probabilistic Fracture Mechanics and Fatigue Methods: Applications for Structural Design and Maintenance*, ASTM STP 798, J. M. Bloom and J. C. Ekvall, Eds., American Society for Testing and Materials, 1983, pp. 67-78.

LOGISTICS ASSET MANAGEMENT: MODELS AND SIMULATIONS

Bjong Wolf Yeigh
Assistant Professor
School of Civil and Environmental Engineering

Oklahoma State University
207 Engineering South
Stillwater, OK 74078

12-1

LOGISTICS ASSET MANAGEMENT: MODELS AND SIMULATIONS

Bjong Wolf Yeigh
Assistant Professor
School of Civil and Environmental Engineering
Oklahoma State University

Abstract

The end of the Cold War brought many changes to the U.S. Armed Forces. Today's Air Force must work not only harder than ever before but also smarter and more efficiently. With limited resources, the Air Force of the 1990s and beyond continues to face the increasing challenges of maintaining aging weapons systems while bringing in new weapons systems. In this light, logistics planning and simulation tools are expected to increase productivity and decrease waste. The Oklahoma City Air Logistics Center (OC-ALC) located at Tinker Air Force Base, Oklahoma, is one of five ALCs. Along with San Antonio ALC located at Kelly AFB, Texas, the two ALCs service majority of U.S. Air Force aircraft. Logistics modeling and simulation tools are needed to assist management in decision making at OC-ALC. To address this need, several logistics planning and simulation models including RAMES, CREST, UNIRAM, LCOM, and others were evaluated for their scientific merit and organizational need. Deficiencies in modeling techniques have been carefully studied. Logistics Composite Model (LCOM) is recommended as the optimal choice for OC-ALC based on its versatility, scope, and development. A preliminary study has been completed on implementing LCOM and its role as the forecasting tool at OC-ALC. An order of magnitude analysis determines that adopting LCOM is expected to save $40-60 million for OC-ALC.

LOGISTICS ASSET MANAGEMENT: MODELS AND SIMULATIONS

Bjong Wolf Yeigh

1. Introduction

From base closures to declining Department of Defense (DoD) budgets, today's Armed Forces face numerous challenges at home and abroad. DoD outfits must do more work with fewer resources. This daunting task can be managed if the resource allocations can be optimized to eliminate waste and duplicity.

The Oklahoma City Air Logistics Center (OC-ALC) at Tinker Air Force Base, Oklahoma, and San Antonio Air Logistics Center (SA-ALC) at Kelly AFB, Texas, service a majority of U.S. Air Force aircraft in terms of man-hours. The Air Logistic Centers (ALC) are extremely large, and their logistics resources are immense. Modeling individual resource factors, such as spare parts needed, are important. However, the interactions of multiple logistics resource factors are crucial in establishing the resource requirements. OC-ALC handles not one but hundreds of weapons systems. Modeling one weapon system let alone hundreds of them would be a difficult task. In addition to the sheer number of weapons systems in inventory, the life cycle of each system has many uncertainties. Tying in reliability and maintainability to the overall logistics resources will be even more difficult.

There are several terms used throughout this paper that are defined here. A list of acronyms is provided in Appendix A. The problem addressed in this study was modeling the weapons systems managed or handled by OC-ALC. A system has a set of components (e.g., pieces, parts, elements, etc.) for which there are cause-and-effect relationships among the variables [Close and Frederick, 1978]. Dynamic systems are time-dependent; static systems are not. To understand these variables and their effects, models are used. Models are constructed from physical laws. It is important to remember that

models are only approximate mathematical descriptions (or abstractions) of the system and, therefore, an unavoidable simplification of reality [Morgan and Henrion, 1990].

Since mainframe computers were used by the Air Force in the 1960s, they have chosen computer simulations have been used to forecast logistics needs. Simulation is the process of building a model of the system to generate information useful for decision making [Pritsker, 1995]. Simulation is as good as the model(s) that describe the real problem. According to Pritsker [1995], the simulation process can be divided into eleven individual steps: (1) problem formulation, (2) model formulation and specification, (3) data acquisition, (4) model translation, (5) verification, (6) validation, (7) strategic and tactical planning, (8) model use, (9) analysis and results, (10) implementation, and (11) documentation.

Reliability and maintainability (R &M) software programs are abundant (Caroli, 1991; AFLC, 1992) and increasing! Until these software programs are used to model real weapons systems, there are no models, only the tools which *can* model. This report will distinguish between models and simulation tools.

2. Uncertainty Modeling

If there were no uncertainties, decisions would be easy. Much of aircraft maintenance, however, is full of uncertainties. From unscheduled repair of equipment to unexpected events of war and peace, predicting aircraft maintenance needs continues to be extremely difficult. Thus, the models or simulation methods selected must take into consideration this difficult task of incorporating random variations in maintenance resources. These resources include but are not limited to manpower, spare parts, aerospace ground equipment, type of missions flown, repair facility availability, and duration of repair.

Sources and implications of uncertainties are many. Sources of uncertainties include statistical variation, subjective judgment, linguistic imprecision, variability, inherent randomness, disagreement between information sources, approximation, among others [Morgan and Henrion, 1990]. Accounting for these uncertainties are a difficult but necessary task.

3. Air Force Maintenance Databases

Before delving into the modeling aspects of logistics resource management, it would seem appropriate to discuss the information contained in the Air Force databases. It is reported that USAF spends an estimated $45-60 million per year on maintenance of existing data systems [*Air Force Times*, 1996]. However, the system suffers from data accuracy, interoperability, and reliability. One General Accounting Office report stated that an overall data error rate of 50 percent had been noted on some USAF databases [*Government Computer News*, 1995]. Of over 181 data system designations [OC-ALC Management Handbook, 1996] that OC-ALC currently uses in some form, the three maintenance databases of interest are the Core Automated Management System (CAMS), the Reliability Engineering Management Information System (REMIS), and the Airlift CAMS (G081).

CAMS is the first of three maintenance systems. It began in 1982 by Standard Systems Group at Maxwell AFB, Alabama. CAMS gathers maintenance data on aircraft, electronic equipment and other assets. In 1986, Litton Computer Systems in Dayton, Ohio, delivered REMIS under subcontract. REMIS is designed to process, store, and retrieve performance and readiness information based on data generated by CAMS. REMIS also delivers R & M information. G081 is a Headquarters Air Mobility Command system utilized to track maintenance data for an airlift weapon system.

The Integrated Maintenance Data Systems (IMDS) was initiated to address deficiencies of the legacy maintenance systems by becoming the single, evolving and integrated information technology program that provide all Air Force persons with the maintenance-related information they need to do their jobs [USAF 002-95-I, 1996]. In 1996, Andersen Consulting LLP of Washington, DC, was awarded a $72.5 million development contract for the second increment of IMDS [*Leading Edge*, 1996]. IMDS will be the maintenance data system for the Air Force. It will eventually take over all databases to eliminate duplication. There are six annual increments to the program. Each increment begins in July and ends in June of the following year. Production Management and Data Collection will be the focus of the first three increments. The second half of IMDS will focus on Integrated Weapons System Management.

Through IMDS, CAMS and REMIS will be phased out, but their functionality will be captured in IMDS with two exceptions—support equipment and training records.

4. Modeling and Simulation Tools

Models and simulation tools for reliability, availability, and maintainability (RAM) are abundant. It appears that every major defense contractor has a simulation engine to sell. This section reviews several RAM models and simulation tools.

4.1 Complete Reliability Evaluation and Sensitivity Techniques (CREST)

CREST is an Air Force owned reliability and maintainability (R & M) software for F-16s and C-5s. First developed by Support System Associates, Inc. (SSAI) and then supported by Science Application International Corporation (SAIC), the program takes input from Air Force Product Performance System (PPS). PPS information is "filtered" to create the CREST database. The output is in terms of Work Unit Code (WUC), Mean Time Between Failure Type 1 (MTBM1), Weibull parameters, reliability at 1.8 and 1,000 hours, failure rate, Mean Repair Time (MRT), standard deviation, crew size, and input data. The technique incorporates an actual system model and calculates reliability accurately, using the reliability block diagrams. It also provides "what if" analysis. The program also provides sensitivity graphs for any WUC, which the users felt was very beneficial. The CREST F-16 and CREST C-5 models were delivered in 1992 and in 1994, respectively [AFLC, 1992].

CREST uses Weibull and lognormal distributions for reliability and maintainability, respectively. Input data are compared to the prescribed distributions to extract shape and scale parameters. The Weibull distribution was chosen by the developer to capture decreasing, constant, and increasing failure rates (i.e., the "bathtub" curve). Reliability block diagrams (RBD) are then used to calculate system reliability.

The CREST F-16 model is no longer used by the Air Combat Command (ACC), and the Air Mobility Command (AMC) currently uses CREST C-5 in limited cases. A unique feature of the CREST C-5 model is that it takes G081 (i.e., airlift CAMS) data quickly into Windows™-based format. The CREST C-5 model, however, has not been kept up-to-date since 1995. Currently, the 1994-1995 database is used, and there are no plans to update its existing database or additional models. Recalling that CREST's RBDs reflect actual physical configuration of the system. Developing RBD is a time consuming investment.

4.2 Reliability, Availability, Maintainability Engineering System (RAMES)

RAMES is a reliability, availability, and maintainability modeling tool developed by SAIC. The input data can be taken from any one of the Air Force maintenance data collection systems such as CAMS and REMIS. It is a PC-based analysis and evaluation tool that completes R & M parameter assessment at weapons system, system, subsystem, Line Replacement Unit (LRU), and Shop Replacement Unit (SRU) levels. It also ranks Air Force Work Unit Codes (WUC) with respect to R & M parameters. The objective of this model is to give a System Program Director (SPD) and staff full control of an R & M performance baseline of a weapons system.

RAMES evolved from the CREST F-16 model. The initial RAMES model development was for the Air Force Satellite Control Network (AFSCN) Remote Tracking Stations (RTSs); Defense Meteorological Satellite Program (DMSP) Command, Control, and Communications (C3) Segment; and Global Positioning System (GPS) Satellite Operations Centers (SOCs). Current RAMES models include the Defense Support System (DSP) and PAVE PAWS (phased array radar system for sea launched ballistic missiles) weapon system. RAMES has been developed extensively in space systems and air traffic controls areas.

Like CREST models, RAMES uses Weibull and lognormal distributions to fit reliability and maintainability data. RAMES integrates life cycle costs and return on investment (ROI) analysis,

logistics analysis, software R & M, and weapons system effectiveness evaluation models. It is becoming a total weapon system analysis tool [Ryan, 1997]. There are no models available for aircraft maintenance at the present time.

4.3 Unit Reliability, Availability, and Maintainability (UNIRAM) Model

Developed by ARINC, UNIRAM [ARINC, 1992] is a deterministic R & M tool employing the Markov modeling technique to determine steady state and time-variant events. It has an R & M focus and uses availability block diagrams and subsystem fault trees to compute MTBF and MDT. Only the first order statistics (i.e., expected values) are used in the model. The model utilizes Monte Carlo simulation only statistical uncertainty, which provides 90 percent confidence intervals for availability, equivalent availability, unplanned downtime, and equivalent planned downtime. UNIRAM allows the user to specify the mean and median of MTBF and MDT. Like other RAM models, UNIRAM can provide "what if" analysis for system design in terms of availability and equivalent availability [ARINC, 1997].

The current UNIRAM customers are the hydrocarbon and chemical processing industries, where the model has been applied successfully. The database is generated from generic sources such as discussions with on-site workers and managers. Plant records are also used to extract pertinent data. The manual data extraction process is still limited. Theoretically, the model can accommodate multi-source databases such as those used in the Air Force. ARINC's UNIRAM model appears to be a sound RAM model using availability as a basis for computation. It has the potential to be used in many other industries requiring RAM such as Air Force depots.

Apart from its graphic input feature which could be more user friendly, there is one significant deficiency—interface with the current and planned Air Force database systems. In theory, the model can extract necessary data from Air Force databases such as CAMS, REMIS, PPS, and future IMDS. However, the necessary interface will require significant investment. In addition, it is estimated that 4-6 months development time is required for each weapons system. More elaborate models would be

required for complex systems like E-3. Modeling every weapons system for OC-ALC would total millions of dollars. UNIRAM would be more cost-effective and well suited for smaller systems where data can be extracted manually and the system size is smaller in scale.

4.4 Avionics Reliability and Configuration Access System (ARCAS)

Developed for the Air Force Avionics Item Management (WR-ALC/LYL) at Robins AFB, Georgia, ARCAS identifies problematic avionics system components across multiple platform installations [AFLC, 1992]. ARCAS is an ARINC product. As a DOS-based software intended to run on a 286 PC, ARCAS uses the Air Force Maintenance and Operational Data Access System (MODAS) report listings. MODAS was phased out in 1990. The output provided top-level comparison charts that rank avionics systems across all platforms where the systems are installed. As the name indicates, the R & M model is restricted to avionics and does not extend to other weapons systems. No system updates have been made since the code was turned over to the Air Force in 1990.

4.5 NASA O & S Model

For the proposed space systems, NASA Langley Research Center established operational and support parameters through an Operation and Support (O & S) model [Ebeling, 1994; Ebeling and Donohue, 1994]. This RAM model, written in SLAM II, predicts R & M parameters for conceptual space vehicles from parametric relationships between vehicle design and performance characteristics and subsystem mean time between maintenance (MTBM) and manhours per maintenance action (MHMA). Due to lack of extensive data for space systems, Air Force AFM-66-1 MDC and Navy 3-M data systems were used in the modeling and simulation. Ten tactical aircraft including F-15s and F-16s, three bombers, eight cargo/tanker aircraft, and four Command, Control, Communications and Intelligence (C3I) aircraft were used in the study. The model appears to be robust and scientifically sound. It is government owned and would be an excellent tool to compare Air Force RAM studies.

4.6 Advanced Logistics Program (ALP)

ALP is a three-stage program examining strategic mobility, logistics planning, and logistics research and development [Draper, 1996]. Started in FY1996, this 5-year program is sponsored by the Defense Advanced Research Projects Agency (DARPA). The program is intended to serve the Defense Logistics Agency (DLA) and Depots. AMC is an intended customer of this program although it has had limited involvement in the program. It is a pure logistics program that addresses the wartime needs of DoD. Because it is still in its infancy, its impact on OC-ALC is unknown.

4.7 Logistics Composite Model (LCOM)

Created in the late 1960s, Logistics Composite Model (LCOM) has been the workhorse for establishing the Air Force manpower requirements. This joint product between the Rand Corporation and the Air Force Logistics Command has received much criticism for its archaic structure and difficulty in use. LCOM, however, is flexible and versatile, which is why it is still in use after nearly 30 years. LCOM has met its intended use repeatedly by providing a "policy analysis tool to relate base-level logistics resources with each other and with sortie generating capability" [Boyle, 1990]. LCOM is a Monte Carlo simulation tool for maintenance—an optimization tool.

LCOM can be broken down into three subsystems—a data preparation system, a main simulation engine, and a series of post processors. The input data (e.g., tasks, task times, failure rates, etc.) come from CAMS, REMIS or other Air Force database systems. This preprocessor converts input data into the initialization and exogenous files. The former file prepares the system to be simulated, while the latter file contains the flying schedule and the mission information. The difficulty with the input module is that LCOM recognizes each input by its form number. However, this is now addressed by converting the front-end of LCOM from a series of COBOL programs to a user-friendly Microsoft Access™ format.

The conceptual framework under which LCOM is found is rather simple. LCOM counts. If aircraft are available, they fly. If they are broken, fix them, then fly them. Based on the sorties and performance variables such as activities, personnel, supply, shop repair, aerospace ground equipment used (AGE), aircraft, and facilities provided by the analysts, LCOM evaluates manpower levels and resources to find their optimal levels [Boyle, 1990]. LCOM simulates unscheduled maintenance requirements by taking random samples from equipment failure parameters defined by the analyst from the Air Force maintenance database. This process is iterated until a satisfactory solution is found. Rather than providing an exact number, LCOM determines confidence levels for manpower. The main simulation module produces the Performance Summary Report (PSR) listing statistics in eight categories: mission, activity, aircraft, personnel, shop repair, supply, equipment, and facility. This information is also recorded in the output file, which in conjunction with the change cards is used in the post processor modules.

The final stage of LCOM is the group of post processors. Post processors are preformatted topic specific reports produced from the data created in the main simulation module. They are broken up into five categories: display, matrix, missions, parts, and support. The display post processor provides the condition of aircraft over the mission duration. Manpower by Air Force Specialty Code (AFSC) is reported in the matrix post processor. This is the most detailed report describing manpower utilization, equipment, and shift summaries. Details of chronological sortie history, parts consumed, and support equipment required for maintenance and repairs are given in the next three post processor reports.

Operational audits were performed to validate LCOM simulation. Both ACC and AMC sent audit crews to operating bases. Information received from maintenance data collection (MDC), AFSC crew size, and maintenance procedure were all verified and updated for the LCOM models. Operational audits in conjunction with Air Force MDS provide the necessary input for weapon system modeling and simulation for ACC and AMC.

Both ACC and AMC have modeled most of their major weapons systems using LCOM. The B-2 weapon system model has begun recently. HH-60, U-2, RC-135, and JSTAR are four weapon systems without working LCOM models at ACC. Like ACC, AMC has most of its major weapons systems modeled in LCOM including C-5, C-141, KC-145, C-17, and C-130, among others. AMC reviews its LCOM input by combining the airlift CAMS (G081) database and operational audit.

LCOM is Air Force owned and maintained. Subcontracts have been awarded to Computer Data Systems, Inc. (CDSI) to develop the Microsoft Access™-based input module, which eliminates COBOL-based Front-End. This update is expected to be operational in 1998. The Analytic Sciences Corporation (TASC) is developing a new LCOM engine. Replacing over 100,000 lines of SIMSCRIPT simulation language with the object-oriented C++, TASC expects the new LCOM engine delivery in the next 3-5 years.

4.8 Other Modeling and Simulation Tools

For E-3, an automated weapons system management tool exists. The Analytic Science Corporation (TASC) delivered Decision Support System (DSS) for E-3 [TASC, 1989]. The Air Force owns this software which provides reliability, maintainability, and "what if" analysis. In addition, DSS provides "warstopper" parts, coverage, support system, mobility, manpower, and cost analyses [TASC, 1989]. DSS is not used at the present time and has not been updated since its final delivery.

DRAIR or Deficiency Report Analysis Information Report produces analysis reports for all aircraft engines in the Air Force inventory. Co-developed by OC-ALC/TI and Southwest Research Institute, DRAIR provides supply and cost data in addition to RAM data. OC-ALC, WR-ALC, SA-ALC, HQ ACC, and HQ AMC heavily use the system. A high-impact listing can be generated for any weapons system in the inventory at system, subsystem, or component level.

5. Discussion

The objective of this study was to recommend a simulation tool for OC-ALC so that the organization can best manage its resources and help achieve the Air Force mission. There are many simulation tools available. They are often referred to as "models" by many. Simulation is the process of building a model of the system on a computer and exercising it to generate information useful for decision making. Models on the other hand are mathematical abstractions of the real system(s) in question. There is no one model that describes every system; rather each model is a unique description of a real system. The real systems for which this study was rendered are the Air Force weapons systems. OC-ALC manages or provides support for over 200 mission designator series (MDS). Each MDS is unique and requires a model, and with time, a weapons system can be taken out of service and a new system can be introduced. Therefore, models must be updated to reflect such changes.

In this study, ten models and/or simulation tools have been examined. Each has its own strengths and weaknesses. Based on their strengths and the needs of OC-ALC, LCOM is recommended. Economics, capabilities, and adaptability were three major factors considered in this final recommendation.

5.1 Economics

Model development is by far the most time consuming task in any modeling and simulation endeavors. Within a major weapon system, there are several mission designator series (MDS). Each MDS must be modeled. OC-ALC manages or provides support for over 200 MDS. In addition there are over 181 maintenance data collection systems. Using simulation tools mentioned above to model or tailor them to the current Air Force weapons systems would be impractical. A working model for each MDS is estimated to cost between $200,000-$300,000. Modeling over 200 MDS would cost between $40-60 million. Cost incurred from operational audits could raise this figure even higher, provided there would be a budget for it. Certainly, LCOM is not the most user-friendly model available to OC-ALC. It does,

however, have an economic advantage over all other simulation tools investigated in this study. LCOM is supported internally by AFMC. Both ACC and AMC have developed LCOM models for most of the major weapons systems in the Air Force inventory.

Adopting LCOM is not free either. As it was discussed earlier, LCOM usage is difficult. There needs to be a dedicated LCOM analyst at OC-ALC, who will maintain the up-to-date models as well as maintain the necessary database systems. This cost, however, would be a fraction of the cost of creating models for over 200 MDS.

5.2. Capabilities

Effective resource management includes the ability to predict supply and demand at all levels in an organization. This was where the "what if" capabilities become important. CREST, RAMES, E-3 DSS, LCOM, ALP, and NASA O&S all have "what if" capabilities. The simulation tool, however, must be flexible in modeling the weapons systems. CREST and RAMES use a Weibull distribution for reliability. Exponential, Weibull, and gamma distributions are very popular lifetime models. None of the distributions by themselves can take decreasing, constant, and increasing failure rates (i.e., bathtub curve) into account unless a piecewise model over segments of the lifetime were used [Leemis, 1995]. In fact, the choice of distribution should be left to the analysts. Curve fitting maintenance data or audit of a specific distribution would not be prudent in accounting for inherent uncertainties in the weapons systems.

LCOM does not specify a specific distribution; rather it allows the analyst to do that task. In this light, LCOM is versatile and flexible. As the maintenance requirements change from peacetime to wartime, the simulation tool can appropriately accommodate such changes. As the aging aircraft are phased out, the model can oblige because the simulation tool was not restricted to one specific distribution. This feature is necessary.

While CREST and RAMES have R & M focus, LCOM has a manpower focus. This focus area is not all that important as long as the simulation provides the necessary information for the organization. LCOM does not explain why a part failed but it counts how many fail over a certain period of time. For design engineers, LCOM would not be helpful since it only provides a relative measure of part reliability. To examine the mechanics of failure, finite element analysis or failure analysis software should be used.

5.3. Adaptability

One of the many challenges that large maintenance facilities such as OC-ALC faces is using the existing data, which are both massive in volume and complicated in structure, with the new model. In theory, all models can be adapted to interface with any extant database. Only ALP, UNIRAM, and ARCAS do not use or have not been updated to use the current Air Force Maintenance Data System. NASA O&S uses REMIS and CAMS data because spacecraft data were not available during its development phase. LCOM is the only platform that can be introduced without changing the current database and maintenance record systems in place at OC-ALC. It would certainly provide additional capabilities since no other system at OC-ALC provides "what if" analysis.

6. Recommendations

Based on OC-ALC's planning and reporting needs, LCOM is recommended for adoption. Although LCOM possesses many difficulties in usage, it contains features that OC-ALC needs to accomplish its mission. No model development is necessary, as there are existing models for most major weapons systems. Comparisons are given in Table 1 below. Coordination with ACC and AMC will be necessary to establish model validation and updates. Details for LCOM adoption include the following:

- Assign one dedicated LCOM analyst/SPM at OC-ALC for coordination and analysis.
- Establish a common model classification system with ACC, AMD, AFCQMI, and ASC for all major weapons systems managed or supported by OC-ALC.

- Use HP750 at AFCQMI, Randolph AFB, Texas, for analysis (short-term solution) until a new LCOM simulation engine is released.

- Perform parametric analysis to determine robustness of LCOM simulation.

The decision-makers should be cognizant of the fact that LCOM has a rather steep learning curve in the beginning. It would take several months to fully integrate LCOM into any organization. As LCOM allows the analyst to control it, the analyst must be fully aware of its intricacies, which the analyst will develop with time. LCOM is a maintenance tool, and it will help OC-ALC optimize resource allocations and eliminate waste and duplicity. It will also help OC-ALC save $40-60 million in developing models, which no other Air Force commands can use.

Table 1: Model Capability Matrix

	Manpower	Mobility Readiness Spares Pack	Support Equipment	Mobility	Sustain-ability	R & M Analysis	"What if" Analysis
CREST		X				X	X
RAMES		X				X	X
UNIRAM		X				X	X
ARCAS						X	X
NASA O&S	X	X	X			X	X
ALP	X	X		X			X
LCOM	**X**	**X**	**X**	**X**	**X**	**X**	**X**
E-3 DSS		X				X	X
DRAIR						X	

7. Acknowledgment

8. References

_____, "Defense Trends," *Air Force Times*, August 5, 1996.

_____, "IMDS," *Leading Edge*, October 1996.

_____, "Integrated Maintenance Data System (IMDS)," *TechTap*, September 27, 1996.

_____, "Many Data Problems," *Government Computer News*, September 18, 1995.

_____, *Reliability Software Tools*, AFLC, OO-ALC/TIETM, March 1992.

ARINC, *Quantitative Approach to Reliability Improvement*, 1997.

ARINC, *UNIRAM*, 1992.

Benjamin, J.R. and C.A. Cornell, *Probability, Statistics, and Decision for Civil Engineers*, McGraw-Hill, 1970.

Boyle, E., *LCOM Explained*, AFHRL Technical Paper 90-58, 1990.

Caroli, J.A., *A Survey of Reliability, Maintainability, Supportability, and Testability Software Tools*, RL-TR-91-87, Rome Laboratory, 1991.

Close, C.M. and D.K. Frederick, *Modeling and Analysis of Dynamic Systems*, Houghton Mifflin, 1978.

Draper, S., *Advanced Logistics*, TAMS FPI Presentation, The MITRE Corporation, 1996.

Ebeling, C.A., *Integrating O&S Models During Conceptual Design--Part II*, NASA NAG1-1-1327 Annual Report, 1991.

Ebeling, C.A. and C.B. Donohue, *Integrating O&S Models During Conceptual Design--Part III*, NASA NAG1-1-1327 Annual Report, 1991.

Gnedenko, B.V. and I.A. Ushakov, Probabilistic Reliability Engineering, Wiley Interscience, 1995.

Leemis, L.M., *Reliability, Probabilistic Model and Statistical Methods*, Prentice-Hall, 1995.

Lewis, E.E., *Introduction to Reliability Engineering*, Second Ed., Wiley, 1994.

Morgan, M.G. and M. Henrion, *Uncertainty, a Guide to Dealing With Uncertainty in Quantitative Risk and Policy Analysis*, Cambridge, 1990.

Patterson, A. and B. Edson, *Reliability, Availability, Maintainability Engineering System (RAMES) Software Reliability Modeling and Analysis Tool Set (SORTS)*, SAIC, 1994.

Pritsker, A.A.B., *Introduction to Simulation and SALM II*, Fourth Ed., Wiley, 1995.

Ryan, J., *A White Paper on the Reliability, Availability, and Maintainability Engineering System (RAMES)*, September 1997.

TASC, *E-3 Automated Weapon System Master Planning/Program Management Information System Decision Support System*, SP-5469-3-1, 1989.

USAF, *Final Operational Requirement Document USAF 002-95-I for Evolutionary Acquisition of the Integrated Maintenance Data System*, HQ USAF/LGMM, 1996.

World Wide Web

IMDS (http://lgm.ssg.gunter.af.mil/IMDS/IMDS.HTM)

RAMES (http://www.afmc.wpafb.af.mil/organizations/MSG/orgs/SQ/)

Appendix

A. Glossary of Terms

ACC	Air Combat Command
AFCQMI	Air Force Center for Quality and Management Innovation
AFLC	Air Force Logistics Center
AFSC	Air Force Specialty Code
AFSCN	Air Force Satellite Control Network
AGE	Aerospace Ground Equipment
ALP	Advanced Logistics Program
AMC	Air Mobility Command
ARCAS	Avionics Reliability and Configuration Access System
C3I	Command, Control, Communications and Intelligence
CAMS	Core Automated Maintenance System
CDSI	Computer Data Systems, Inc.
CREST	Complete Reliability Evaluation and Sensitivity Technique
DARPA	Defense Advanced Research Project Agency
DLA	Defense Logistics Agency
DMSP	Defense Meteorological Program Command
DoD	Department of Defense
DRAIR	Deficiency Report Analysis Information Report
DSP	Defense Support Program
DSS	Decision Support System
GAO	General Accounting Office
GPS	Global Positioning System
IMDS	Integrated Maintenance Data System
LCOM	Logistics Composite Model
LRU	Line Replaceable Unit
MDC	Maintenance Data Collection
MDS	Maintenance Data System
	Mission Designator Series
MDT	Mean Down Time
MHMA	manhours per maintenance action
MODAS	Maintenance Operational Data Access System
MRT	Mean Repair Time
MTBF	Mean Time Between Failures
MTBM	Mean Time Between Maintenance
MTBM1	Mean Time Between Maintenance Type 1
MTTR	Mean Time To Repair
O&S	Operation and Support
PAVE PAWS	Phased array radar system for sea launch ballistic missiles
PPS	Product Performance System
PSR	Performance Summary Report
R & M	Reliability and Maintainability
RAM	Reliability, Availability, and Maintainability
RAMES	Reliability, Availability, and Maintainability Engineering System
RBD	Reliability Block Diagram
REMIS	Reliability and Maintainability Information System
ROI	Return On Investment

RTS	Remote Tracking System
SM	Single Manager
SOC	Satellite Operation Center
SPD	System Program Director
SPM	System Program Manager
SRU	Shop Replaceable Unit
SAIC	Science Applications International Corporation
SSAI	Support System Associates, Inc.
TASC	The Analytic Sciences Corporation
UNIRAM	Unit Reliability, Availability, and Maintainability
WUC	Work Unit Code

B. Interviews

ALP
Mike Donovan and Stu Draper, The MITRE Corporation (618-256-5109)
James Jamieson, LTC, AMCSC (618-256-8749)

CREST F-16 and C-5
Carl Bargar, OO-ALC/LMEI (DSN 777-6948)
Rubin Bell, CAPT and Christina Vilella, CAPT, AMC (DSN 576-6487)
Bruce Edson and John Ryan, SAIC (916-974-8800)

DRAIR
Gayle Davis, TIET/OC-ALC (DSN 336-5567)

IMDS
Phillip Denyse, ASC, Hanscom AFB (DSN 478-6141)

LCOM
David Albers, HQ AMC (DSN 576-6364)
Gene Brown, SMSgt and Fred Juarez, AFCQMI (DSN 487-4690)
Dr. Charles Ebeling, Univ. of Dayton
David Jennings and Mike York, CDSI (DSN 487-4690)
Anthony Scheidt, CAPT, HQ ACC (DSN 574-3292)
Alan Wallace, ASC-XR (DSN 785-8060)
Eric Zahn, TASC (937-426-1040)

RAMES
Bruce Edson and John Ryan, SAIC (916-974-8800)

UNIRAM and ARCAS
Robert Rennell, ARINC (405-739-0939)
Dr. Yupo Chan, AFIT (DSN 986-4943)

Delisting of Hill Air Force Base's Industrial Wastewater Treatment Plant Sludge

Michael J. McFarland
Associate Professor
Department of Civil and Environmental Engineering

Utah State University
River Heights, UT 84321

Final Report for:
Summer Faculty Research Program
Ogden Air Logistic Center

DELISTING OF HILL AIR FORCE BASE'S
INDUSTRIAL WASTEWATER TREATMENT PLANT SLUDGE

Michael J. McFarland
Associate Professor
Department of Civil and Environmental Engineering
Utah State University

Abstract

The US Air Force has set an environmental goal of 50% reduction of hazardous waste disposal by December 31, 1999. To meet this objective, Hill Air Force Base (Hill AFB) desires to delist its industrial wastewater treatment plant (IWTP) sludge which represents the largest single hazardous waste stream generated by the base.

The first step towards IWTP sludge delisting involved a legal review of both the code of federal regulations (CFR) and recent court rulings pertaining to industrial hazardous wastes. The results of this effort indicated that, since the IWTP sludge is produced by treatment of commingled wastewaters, it is not a "listed" hazardous waste under the Resource Conservation and Recovery Act (RCRA).

The second step toward sludge delisting involved conducting a metal material balance around the IWTP to identify the major sources of cadmium and chromium which impart the hazardous "characteristics" to the IWTP sludge. Results of the material balance indicated that the major sources of cadmium and chromium to the IWTP sludge were the following flows: 1) Building 505 Main Flow, 2) Building 505 Cyanide Flow and 3) Carboys from Buildings 505 and 507. To demonstrate to the regulatory authorities that delisting of the IWTP sludge does not diminish protection of human health and the environment, future pollution prevention (P2) efforts at Hill AFB should be directed towards reducing the metals loadings from these flows.

DELISTING OF HILL AIR FORCE BASE'S
INDUSTRIAL WASTEWATER TREATMENT PLANT SLUDGE

Michael J. McFarland

Introduction

Hill Air Force Base (Hill AFB) annually disposes of approximately 102,400 kilograms (225,300 lbs - dry weight) of industrial waste treatment plant (IWTP) sludge at a cost of nearly $500,000. This waste flow represents the single largest hazardous waste stream from the base and is presently managed as a listed hazardous wastes under Subpart D of the Resource Conservation and Recovery Act (*i.e.*, RCRA - 40 CFR Part 261). Hill Air Force Base desires to delist this material as a hazardous waste which will enable the base to attain a US Air Force goal of 50% reduction of hazardous waste disposal by December 31, 1999.

To accomplish IWTP sludge delisting, several tasks were identified by Hill AFB as critical to this effort. These include the following items:

- development of a regulatory "road map" for the sludge delisting petition

- performance of a materials balance to identify the source of those constituents which now requires the sludge to be managed as a hazardous waste

- provide recommendations of additional efforts that will assist Hill AFB in its delisting petition

Background

The regulatory requirements which define a hazardous wastes are described in the RCRA hazardous waste rules (*i.e.*, 40 CFR Part 261.3). In general, a solid waste is a hazardous waste if it is not excluded from the regulations as a hazardous waste under 40 CFR Part 261.4 and meets any one of the following three criteria:

- it exhibits any of the characteristics of hazardous waste identified in 40 CFR Part 261 Subpart C (*i.e.*, "characteristic hazardous waste")

- it is a "listed" hazardous waste as described in 40 CFR Part 261 Subpart D

- it is a mixture of a solid waste and a hazardous waste that is listed in Subpart D and it exhibits one or more of the characteristics of a hazardous waste identified in 40 CFR Part 261 Subpart C

In order to develop a plausible regulatory "road map" for the sludge delisting petition, it is important that the regulatory definitions of a hazardous waste be thoroughly understood. Moreover, those aspects of the regulatory definition that directly impact the delisting process, namely 40 CFR Part 261 Subparts C and D, must be logically integrated into the delisting argument for the effort to be successful. Each of these Subparts will now be introduced together with a description of how each impacts the hazardous waste delisting process.

40 CFR Part 261 Subpart C

Under 40 CFR Part 261 Subpart C, a solid waste is a hazardous waste if it is not excluded as a hazardous waste under 40 CFR Part 261.4 and if it exhibits any one of the following four characteristics:

1. Characteristic of ignitability (40 CFR Part 261.21)
2. Characteristic of corrosivity (40 CFR Part 261.22)
3. Characteristic of reactivity (40 CFR Part 261.23)
4. Characteristic of toxicity (40 CFR Part 261.24)

Historical analyses of Hill Air Force Base's IWTP sludge have demonstrated that it does not possess the characteristics of ignitability, corrosivity or reactivity as defined in 40 CFR Part 261. Therefore, the sludge is not a characteristic hazardous waste based on any of these first three characteristic criteria.

The characteristic of toxicity, which is determined using the Toxicity Characteristic Leaching Procedure (TCLP - Test Method 1311 - EPA SW846), involves analysis of a dilute acetic acid extract of the sludge. If the acid extract contains any of the contaminants

listed in Table 1 at a concentration equal to or greater than the given regulatory level, the solid waste is designated as a "characteristic" hazardous waste based on its toxicity.

TABLE 1.—MAXIMUM CONCENTRATION OF CONTAMINANTS FOR THE TOXICITY CHARACTERISTIC

EPA HW No.[1]	Contaminant	CAS No.[2]	Regulatory Level (mg/L)	EPA HW No.[1]	Contaminant	CAS No.[2]	Regulatory Level (mg/L)
D004	Arsenic	7440-38-2	5.0	D033	Hexachlorobutadiene	87-68-3	0.5
D005	Barium	7440-39-3	100.0	D034	Hexachloroethane	67-72-1	3.0
D018	Benzene	71-43-2	0.5	D008	Lead	7439-92-1	5.0
D006	Cadmium	7440-43-9	1.0	♦ RCRA—275			
D019	Carbon tetrachloride	56-23-5	0.5	D013	Lindane	58-89-9	0.4
D020	Chlordane	57-74-9	0.03	D009	Mercury	7439-97-6	0.2
D021	Chlorobenzene	108-90-7	100.0	♦ RCRA—211			
D022	Chloroform	67-66-3	6.0	D014	Methoxychlor	72-43-5	10.0
D007	Chromium	7440-47-3	5.0	D035	Methyl ethyl ketone	78-93-3	200.0
D023	o-Cresol	95-48-7	[4]200.0	D036	Nitrobenzene	98-95-3	2.0
D024	m-Cresol	108-39-4	[4]200.0	D037	Pentachlorophenol	87-86-5	100.0
D025	p-Cresol	106-44-5	[4]200.0	D038	Pyridine	110-86-1	[3]5.0
D026	Cresol		[4]200.0	D010	Selenium	7782-49-2	1.0
D016	2,4-D	94-75-7	10.0	D011	Silver	7440-22-4	5.0
D027	1,4-Dichlorobenzene	106-46-7	7.5	D039	Tetrachloroethylene	127-18-4	0.7
D028	1,2-Dichloroethane	107-06-2	0.5	D015	Toxaphene	8001-35-2	0.5
D029	1,1-Dichloroethylene	75-35-4	0.7	D040	Trichloroethylene	79-01-6	0.5
D030	2,4-Dinitrotoluene	121-14-2	[3]0.13	D041	2,4,5-Trichlorophenol	95-95-4	400.0
D012	Endrin	72-20-8	0.02	D042	2,4,6-Trichlorophenol	88-06-2	2.0
D031	Heptachlor (and its epoxide)	76-44-8	0.008	D017	2,4,5-TP (Silvex)	93-72-1	1.0
D032	Hexachlorobenzene	118-74-1	[3]0.13	D043	Vinyl chloride	75-01-4	0.2

[1]Hazardous waste number.

[2]Chemical abstracts service number.

[3]Quantitation limit is greater than the calculated regulatory level. The quantitation limit therefore becomes the regulatory level.

[4]If o-, m-, and p-Cresol concentrations cannot be differentiated, the total cresol (D026) concentration is used. The regulatory level of total cresol is 200 mg/l.

[55 FR 11862, Mar. 29, 1990, as amended at 55 FR 22684, June 1, 1990; 55 FR 26987, June 29, 1990; 58 FR 46049, Aug. 31, 1993]

Taken from: RCRA Regulations and Keyword Index (1997) Elsevier Publishing

Historical data indicate that the TCLP extract of Hill Air Force Base's IWTP sludge has routinely exceeded the regulatory level for cadmium and, in some cases, chromium. Therefore, the IWTP sludge would be considered a "characteristic" hazardous waste under 40 CFR Part 261 Subpart C. However, recent IWTP process modifications, namely the installation of batch treatment filter presses, have been successful in removing significant quantities of cadmium and chromium from the IWTP process stream and presumably from the sludge. If, due to these process changes, the resulting cadmium and chromium concentrations in the TCLP sludge extract were shown statistically to be below the regulatory levels, the IWTP would no longer be designated as a "characteristic" hazardous waste.

Although the historical TCLP analyses had indicated that Hill Air Force Base's IWTP sludge was a "characteristic" hazardous wastes as defined by 40 CFR Part 261 Subpart C, the sludge has been managed by Hill AFB as a "listed" hazardous waste according to 40 CFR Part 261 Subpart D.

40 CFR Part 261 Subpart D

Under 40 CFR Part 261 Subpart D, a solid waste is a hazardous waste if it is listed in this Subpart and has not been excluded through a delisting petition as described under 40 CFR Part 260.20 and Part 260.22. Currently, Hill Air Force Base manages its IWTP sludge as a listed F006 hazardous waste which is defined as follows:

F006 - Wastewater treatment sludges from electroplating operations except from the following processes: 1) sulfuric acid anodizing of aluminum, 2) tin plating on carbon steel, 3) zinc plating (segregated basis) on carbon steel, 4) aluminum or zinc-aluminum plating on carbon steel, 5) cleaning/stripping associated with tin, zinc and aluminum plating on carbon steel, and 6) chemical etching and milling of aluminum.

Since Hill Air Force Base's IWTP sludge is generated from the processing of wastewater discharged from, among other sources, electroplating operations, it would appear that the IWTP sludge would meet the requirements of the F006 category. However, given recent court rulings regarding this type of "listed" waste, the decision to manage the IWTP sludge as a F006 hazardous waste requires reevaluation.

Legal Challenges to the F006 Listed Waste Category

In an unsuccessful legal challenge, the USEPA filed a law suit against the Bethlehem Steel Corporation on grounds that the company knowingly violated RCRA (40 CFR Part 261 Subpart D) by mismanaging their industrial wastewater treatment sludge and neglecting to obtain the required legal permits (US Government vs. Bethlehem Steel Corp. US Court of Appeals Seventh Circuit Sept. 26, 1994). It was government's contention that Bethlehem Steel Corporation's industrial wastewater treatment plant sludge was a listed F006

hazardous waste under the RCRA rules and that the company was legally required to manage the waste accordingly. Moreover, the USEPA claimed that, since they had regulatory authority under RCRA to declare a waste as hazardous by rule making, these particular industrial wastes remained hazardous wastes until the company received government approval for its delisting.

Bethlehem Steel Corporation argued that its wastewater treatment sludge was not an F006 listed waste since it was generated through treatment of wastewaters discharged from electroplating operations *together* with wastewaters from a variety of other sources. Moreover, the regulatory language in 40 CFR Part 261 implied to the company that the F006 listed hazardous wastes pertained to those sludges generated from the treatment of electroplating wastewater exclusively.

The US Court of Appeals disagreed with the USEPA and ruled in favor of Bethlehem Steel Corporation citing the following reasons:

- The F006 listing lacks the phrase "mixtures/blends" or any mention of a threshold concentration percentage (*e.g.*, 10% or more electroplating wastewater). The preceding F001 through F005 listings (which cover wastes generated from organic solvent use) contain both descriptions. Therefore, a facility may therefore reasonably infer that when the USEPA intends to include waste mixtures in its listings, it knows how to do so and that in the F006 listing, such mixture language is conspicuously absent.

- F006 listed wastes do not explicitly include sludges produced from treating wastewater that comes only in part from electroplating operations.

- The industrial wastewater sludges generated by Bethlehem Steel Corporation were produced from wastewater mixtures and not solely electroplating wastewaters.

- The USEPA lacked authority under RCRA to bring enforcement action based on the claim that mixtures of hazardous and nonhazardous waste mixtures are themselves hazardous wastes regulated under 40 CFR Part 261.

In a related court case (USEPA vs. Shell Oil 950 F.2d. 741 34 ERC 1049 D.C. Circuit 1991), the USEPA argued that the regulation of waste mixtures (*i.e.*, combinations of listed hazardous wastes and nonhazardous wastes) is a result of a more general principle and provision contained within the RCRA rules. Namely, that RCRA's "continuing jurisdiction" principle held that a hazardous wastes portion of a mixture is subject to the hazardous waste regulations (*i.e.*, that one cannot hide wastes or change its hazardous characteristic by mixing it into a pile of non-hazardous wastes).

The District of Columbia Circuit Court disagreed with the USEPA arguing that the principle of "continuing jurisdiction" applied not to mixtures of hazardous and nonhazardous wastes but to hazardous wastes and environmental media such as soil and groundwater. The court concluded that the USEPA's contention of a mixing rule was neither implicit in nor a logical outgrowth of the RCRA regulations.

In light of the court rulings, it seems clear that Hill Air Force Base's IWTP sludge has been incorrectly categorized as a F006 listed hazardous waste since it is generated from a *combination* of electroplating wastewater together with a variety of other wastewaters. The F006 category, as interpreted by the courts, pertain to sludges generated from the treatment of electroplating wastewater exclusively. Therefore, the IWTP sludge is not legally a listed hazardous waste under 40 CFR Part 261 Subpart D.

Given the court interpretations of the RCRA rules, it appears that Hill Air Force Base has a reasonably strong legal position to proceed with efforts to remove the IWTP sludge as a "listed" hazardous waste. However, its removal as a "listed" hazardous waste does not mean that the IWTP sludge could be managed as a nonhazardous waste. Its hazardous characteristics, namely the high cadmium and chromium content, must be sufficiently reduced for a successful sludge delisting petition. To accomplish this second task, the principal sources of these metals were identified through an IWTP metals materials balance.

<u>Methodology for IWTP Metals Material Balance</u>

The industrial wastewater treatment plant (IWTP) at Hill AFB currently treats approximately 360,000 gallons per day of industrial wastewater generated from a variety of base facilities and processes. After review of both the IWTP collection system drawings and the electroplating bath discharge records, six wastewater input flows were identified which were suspected to contain significant concentrations of cadmium and chromium. These consisted of the following flows:

- Base Main Flow

- Building 505 Main Flow

- Building 507 Main Flow

- Operable Unit 1 and 2 Groundwater Flows

- Cyanide Pipeline from Building 505

- Building 505 and 507 Carboys

Figure 1 depicts the major metal flow inputs to the IWTP together with the two metal output flows (*i.e.*, IWTP sludge and the treated wastewater).
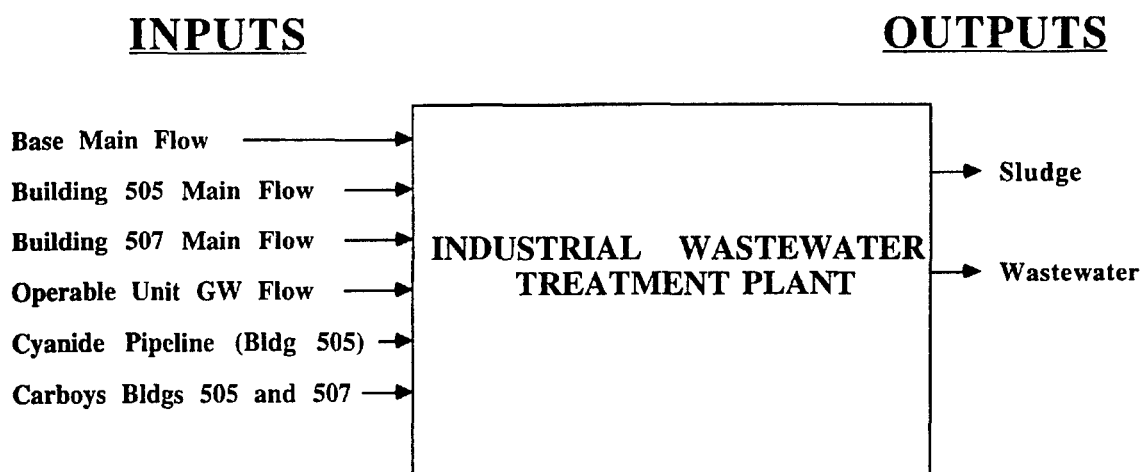
Figure 1.   Schematic Diagram of Flow Inputs and Outputs to the IWTP

The mass flows of cadmium and chromium coming into and leaving the IWTP are evaluated in the following sections.

Base Main Flow

The Base Main Flow includes all the continuous industrial wastewater flows at Hill AFB excluding those from Buildings 505 and 507. These flows include the industrial wastewater generated from the Aircraft Directorate (LA), Missile Directorate (LM), Base Laboratories (TI), 388th Fighter Wing, 419th Fighter Wing *etc.* Although each of these flows enters the Hill AFB industrial sewer collection system at various points, they all ultimately flow through Lift Station 3 prior to being transferred to the IWTP flow equalization tank (T1). Therefore, to estimate the volumetric flow rate of the Base Main Flow, it was decided to quantify the wastewater flow transferred through Lift Station 3 to the IWTP.

Lift Station 3 is equipped with two (2) 400 gallon per minute wastewater pumps which are activated by a float switch located in the wet well. Because of time considerations, a one week analysis of the flow rate discharged by the two pumps was conducted. Pump no #1 ran for 19.7 hours over the seven day analysis period while Pump no #2 ran for 2.9 hours. Extrapolation of this one week period of flow to the entire year gave an annual flow estimate of 28.3 million gallons.

To evaluate the chromium and cadmium mass loadings from the Base Main Flow to the IWTP, one (1) grab sample was obtained from Lift Station 3 for chemical analysis at the beginning of the flow rate analysis test. The analysis (performed by the base laboratory - TIELC) indicated that the chromium and cadmium levels in the sample were 0.25 and 0.02 mg/liter, respectively. The metal mass loadings were then estimated by the following procedure:

Chromium

$$\frac{\text{lbs}}{\text{year}} = 0.25 \text{ mg/liter} \times 28.3 \frac{\text{MG}}{\text{year}} \times 8.34 = 59.0 \text{ lbs/year}$$

Cadmium

$$\frac{\text{lbs}}{\text{year}} = 0.02 \text{ mg/liter} \times 28.3 \frac{\text{MG}}{\text{year}} \times 8.34 = 4.7 \text{ lbs/year}$$

Building 505 Main Flow

Building 505 Main Flow consists of wastewater discharged from electroplating operations and floor drains. The electroplating wastewater consists primarily of plating rinse waters together with mild acidic or basic cleaning solutions. The floor drains, both in the electroplating and machine shop areas, are commingled with the electroplating rinse waters prior to leaving Building 505. The volumetric flow rate is monitored continuously using a Palmer Bowlus Flume equipped with an open channel meter (OCM) located in a concrete vault on the east side of Building 505.

Although installed over six months ago, operation of the continuous flow monitoring system was not fully activated until the mid June 1997. Because of time considerations, a one week flow rate analysis was conducted at the end of June 1997 to estimate the annual Building 505 main flow. The one week analysis indicated an average flow rate of 73 gallons per minute or 38.5 million gallons per year.

To evaluate the chromium and cadmium mass loadings from the Building 505 Main Flow, one (1) grab sample was obtained from the Palmer Bowlus flume for chemical analysis at the beginning of the flow rate analysis test. The analysis (performed by the base laboratory - TIELC) indicated that the total chromium and cadmium levels in the sample were 53.1 and 0.04 mg/liter, respectively. The metal mass loadings were then estimated by the following procedure:

Chromium

$$\frac{lbs}{year} = 53.1 \text{ mg/liter} \times 38.5 \frac{MG}{year} \times 8.34 = 17,050 \text{ lbs/year}$$

Cadmium

$$\frac{lbs}{year} = 0.04 \text{ mg/liter} \times 38.5 \frac{MG}{year} \times 8.34 = 12.8 \text{ lbs/year}$$

Building 507 Main Flow

Building 507 Main Flow consists primarily of rinse waters and dilute cleaning solutions generated from the stripping of various metal coatings (*e.g.*, topcoat, primer and anodize seal). This flow also includes wastewater discharged from the various floor drains in Building 507. All of the rinse waters, dilute cleaning solutions and floor drain wastewater discharged from Building 507 Main Flow is collected in a sump. The level in the sump is continuously controlled by a pump which is activated by a float switch. Inside the discharge side piping is a paddle type flow meter which continuously records the flow removed from the sump.

Although there have been some questions regarding the accuracy of the paddle flow meter, due to time constraints, it was decided to use this meter to estimate the volumetric flow rate. The Building 507 Main Flow was monitored over a four day period during which time a flow rate of 5000 gallons per day or 1.83 million gallons per year was estimated. It should be noted that recommendations to independently validate the flow rates from Building 507 are addressed later in this report.

To evaluate the chromium and cadmium mass loadings from the Building 507 Main Flow, one (1) grab sample was obtained from the Building 507 sump for chemical analysis at the beginning of the flow rate analysis test. The analysis (performed by the base laboratory - TIELC) indicated that the total chromium and cadmium levels in the sample were 0.41 and 0.14 mg/liter, respectively. The metal mass loadings were then estimated by the following procedure:

Chromium

$$\frac{\text{lbs}}{\text{year}} = 0.41 \text{ mg/liter} \times 1.83 \frac{\text{MG}}{\text{year}} \times 8.34 = 6.3 \text{ lbs/year}$$

Cadmium

$$\frac{\text{lbs}}{\text{year}} = 0.14 \text{ mg/liter} \times 1.83 \frac{\text{MG}}{\text{year}} \times 8.34 = 2.1 \text{ lbs/year}$$

Operable Unit 1 and 2 Groundwater Flows

Operable Units 1 and 2 are located on the east side of Hill AFB. Historically, these sites were used for disposal of wastes from industrial operations (*e.g.*, spent paint and cleaning solvents). Due to its contamination, several efforts have been initiated to remediate the contiguous groundwater aquifer beneath Operable Units 1 and 2. Since the treated water in not legally permitted to be discharged back to the aquifer, water from these remediation treatment efforts are transferred to the IWTP for further processing.

Average flow data records indicate an annual total Operable Unit groundwater discharge to the IWTP of 32.5 million gallons. This average flow was estimated using data records from 1994, 1995 and 1996 (data records obtained from P. Betts - Hill AFB, 1997).

With regard to the chromium and cadmium concentrations in the Operable Unit groundwater, the data obtained indicate that were no detectable background level concentrations of either metal (*Revised Interim Draft Final Feasibility Study Report for Operable Unit 1, HAFB, Utah* - March 1996). However, in the risk assessment report for the same aquifer, a chromium and cadmium concentration of 10 and 8 µg/liter were used, respectively. These concentrations presumably represent a limited number of detectable "hits" on both metals from the contaminated aquifer. The use of these risk assessment numbers in the present material balance effort, therefore, represents a conservative estimate of actual metal loadings to the IWTP.

To evaluate the chromium and cadmium mass loadings, the total chromium and cadmium concentrations in the transferred Operable Unit groundwater will be assumed to be 0.01 and 0.008 mg/liter, respectively. The metal mass loadings are then estimated by the following procedure:

Chromium

$$\frac{lbs}{year} = 0.01 \text{ mg/liter} \times 32.5 \frac{MG}{year} \times 8.34 = 2.7 \text{ lbs/year}$$

Cadmium

$$\frac{lbs}{year} = 0.008 \text{ mg/liter} \times 32.5 \frac{MG}{year} \times 8.34 = 2.2 \text{ lbs/year}$$

Cyanide Pipeline from Building 505

Cyanide rinse waters are discharged from Building 505 intermittently to a cyanide wet well located within the IWTP facility. A float switch within the wet well activates a pump that transfers the wastewater to either of two cyanide/cadmium batch treatment tanks where the cyanide is oxidized by treatment with chlorine.

Radian International, LLC (Radian) was contracted by Hill AFB to analyze the cyanide/cadmium rinse water flows from Building 505. By monitoring the pumping rate at the wet well over a 2.5 week period, Radian estimated a flow rate of 30,000 gallons per week or 1.56 million gallons per year for the cyanide rinse waters.

To evaluate the chromium and cadmium mass loadings from the cyanide rinse water flow, one (1) grab sample was obtained by Radian from the cyanide/cadmium wet well for chemical analysis. The analysis (performed by Radian using a Hatch™ Kit) indicated that the total chromium and cadmium concentrations in the sample were 2.0 and 13.0 mg/liter, respectively. The metal mass loadings were then estimated by the following method:

Chromium

$$\frac{lbs}{year} = 2.0 \text{ mg/liter} \times 1.56 \frac{MG}{year} \times 8.34 = 26.0 \text{ lbs/year}$$

Cadmium

$$\frac{lbs}{year} = 13.0 \text{ mg/liter} \times 1.56 \frac{MG}{year} \times 8.34 = 169.0 \text{ lbs/year}$$

Building 505 and 507 Carboys

The concentrated plating and stripping bath solutions are transferred from Buildings 505 and 507 to the IWTP in 400 gallon carboys. These carboys are used to transfer concentrated solutions associated with: 1) chrome plating, 2) chrome stripping, 3) chromic acid anodizing 3) anodize stripping, 4) cadmium plating, 5) cadmium stripping, 6)

dichromate sealing, 6) alodining *etc.* The frequency at which these solutions are transferred to the IWTP depend on their level of contamination. Once a plating tank solution is deemed unsuitable for further use by the Commodities (LI) directorate laboratory personnel, the IWTP is called to empty the tank into one or more 400 gallon carboys. The filled carboy(s) are then transferred back to the IWTP where the contents are discharged into the appropriate batch reactor for treatment.

Depending on its chemical composition, the contents of the carboys are emptied into one of several batch treatment bays (*e.g.*, alkali sump, acid sump, cyanide rinse water sump, concentrated cyanide sump) at the IWTP. Once in the batch treatment bays, the solutions are treated (*e.g.*, neutralized) after which they are trickled into the IWTP equalization tank (T3) for further processing.

To estimate the annual mass loadings of chromium and cadmium to the IWTP from the concentrated plating solutions, carboy data from the Waste Inventory Tracking System (WITS) was reviewed. WITS data from 1994, 1995 and 1996 were aggregated and then averaged to estimate an annual metals loading to the IWTP from the carboys. Based on the WITS data, the chromium and cadmium loadings from the carboys were estimated to be 4,455 lbs ± 1,560 and 176 lbs ± 17, respectively.

IWTP Sludge

Analysis of IWTP sludge production for calendar year 1996 (obtained from the WITS database) indicated an average annual production rate of 102,400 kilograms or 225,300 lbs (dry weight). With regard to its chemical composition, one (1) composite grab sample of IWTP sludge was obtained and analyzed in June 1997. The analysis (performed by the base laboratory - TIELC) indicated that the total chromium and cadmium levels in the sample (on a dry basis) were 57,000 mg/kg and 2,040 mg/kg, respectively. The metal mass loadings were then estimated by the following method:

Chromium

$$\frac{lbs}{year} = \frac{57,000\ mg}{kg} \times \frac{102,400\ kg}{year} \times \frac{1\ gm}{1000\ mg} \times \frac{1\ lb}{454\ gms} = 12,860\ lbs/yr$$

Cadmium

$$\frac{lbs}{year} = \frac{2,040\ mg}{kg} \times \frac{102,400\ kg}{year} \times \frac{1\ gm}{1000\ mg} \times \frac{1\ lb}{454\ gms} = 460\ lbs/yr$$

Treated Wastewater Discharged From the IWTP

Analysis of the wastewater discharged from the IWTP was conducted over the period of May and June 1997. During that time period, an average volumetric flow rate of 357,400 gallons per day (± 18,100) was recorded. This daily average flow was equivalent to an annual flow of 130.5 million gallons.

Inspection of the daily cadmium and chromium data indicated that over the two month period, the effluent concentrations exceeded the pretreatment local limits for chromium on nine (9) separate occasions. In spite of the noncompliance occurrences, a monthly average chromium concentration was estimated to be 0.7 mg/liter. The monthly average cadmium concentration was estimated to be 0.024 mg/liter. The metal mass flow rates from the IWTP from the discharged wastewater were then estimated by the following method:

Chromium

$$\frac{lbs}{year} = 0.7\ mg/liter \times 130.5\ \frac{MG}{year} \times 8.34 = 762\ lbs/year$$

Cadmium

$$\frac{lbs}{year} = 0.024\ mg/liter \times 130.5\ \frac{MG}{year} \times 8.34 = 26\ lbs/year$$

<u>Discussion</u>

In analyzing the metal inputs to the IWTP, the major source of chromium was found to be the Building 505 Main Flow which accounted for approximately 79% of the chromium input. The other major source of chromium to the IWTP were the carboys from Buildings 505 and 507. The carboys accounted for an additional 20% of the chromium input to the IWTP. The other input flows added negligible amounts of chromium to the IWTP (*i.e.*, less than 1%).

With regard to cadmium, both the cyanide pipeline from Building 505 and the carboys from Building 505 and 507 each contributed approximately 46% of the cadmium loading. The other sources of cadmium were insignificant relative to these sources. Table 2 summarizes the inputs of chromium and cadmium metal to the IWTP from the various sources.

Table 2   Summary of Chromium/Cadmium Inputs to the IWTP

	Chromium (lbs/year)	Cadmium (lbs/year)
Base Main Flow	59	4.7
Building 505 Main Flow	17,050	12.8
Building 507 Main Flow	6.3	2.1
OU 1 and 2 Groundwater	2.7	2.2
Cyanide Pipeline (Bldg 505)	26	169
Carboys Bldg 505 and 507	4,455	176
TOTALS	21,600	367

Table 3 summarizes the outputs of cadmium and chromium metal from the IWTP. As anticipated, the major source of cadmium and chromium leaving the IWTP occurs through disposal of the IWTP sludge. The sludge accounts for approximately 95% of the cadmium and chromium removed from the IWTP annually.

Table 3	Summary of the Cadmium/Chromium Outputs from the IWTP	
	Chromium (lbs/year)	Cadmium (lbs/year)
IWTP Sludge	12,860	460
Treated Wastewater	762	26
TOTAL	13,622	486

Table 4 summarizes the overall cadmium and chromium materials balance for the IWTP. Under ideal conditions, the values in each column should have been reasonably close to one another ($\pm$ 10%) indicating acceptable data quality. Due to the large discrepancies in the input and output metal mass flows, the data quality in this materials balance clearly needs significant improvement. For example, the difference between the IWTP influent and effluent chrom. mass flow rates was approximately 8,000 lbs per year. In other words, nearly 60% of the influent chromium loading to the IWTP could not be accounted for in the output IWTP sludge and treated wastewater flow using the given data.

Table 4	Summary of the Overall Cadmium/Chromium IWTP Material Balance	
	Chromium (lbs/year)	Cadmium (lbs/year)
INPUT	21,600	367
OUTPUT	13,622	486

On the other hand, the difference between the IWTP influent and effluent cadmium mass flow rate was approximately 119 lbs per year. In this case, the output cadmium mass flow was nearly 25% greater than the estimated influent cadmium mass flow.

There were several major flaws in the present approach to conducting the metals material balance including the lack of replicate chemical sampling and extrapolation of short term flow rate analysis to estimate annual flows. Statistically, the larger the number of chemical samples taken and the longer the period over which the volumetric flow rate measurement

is evaluated, the closer the sampling data becomes to reflecting average conditions at the IWTP. Although increasing sample collection and flow rate measurement frequencies are sometimes costly, the need for directing additional resources for improved data quality must be weighed against data quality objectives.

Conclusions and Recommendations

The evaluation of IWTP sludge delisting at Hill AFB was characterized by two independent efforts that were conducted simultaneously. The first included a preliminary legal review of the regulatory issues pertaining to a sludge delisting petition while the second focused on identifying the source of metals which imparted a hazardous "characteristic" to the IWTP sludge.

The principal result of the legal review was that there is no longer any regulatory requirement to manage the IWTP sludge as a F006 "listed" hazardous waste. The court interpretation of the RCRA rules indicate that the F006 category only applies to sludge generated from the treatment of electroplating wastewaters exclusively. In spite of the fact that the IWTP may no longer be a "listed" waste, it may still possess hazardous characteristics. The sludge should be reevaluated in light of recent process changes at the IWTP (*i.e.*, installation of batch treatment filter presses) to determine whether or not the sludge can be eventually delisted.

The metal materials balance identified the major sources of chromium to be Building 505 Main Flow and the carboys from Buildings 505 and 507 while the major sources of cadmium were found to be the cyanide pipeline from Building 505 and the carboys from Buildings 505 and 507. In order for IWTP sludge delisting to be successful, it must be demonstrated to the regulatory authorities that delisting does not diminish protection of public health and the environment. This can be accomplished by directing future pollution prevention (P2) resources to those flows which contribute the largest amounts of cadmium and chromium to the IWTP.

In spite of its value in directing future P2 resources, the metal materials balance has limited utility with regard to improving operational performance of the IWTP. This limited utility stems directly from the fact that the data quality varied significantly among the various influent and effluent flows. Without a statistically verifiable materials balance it is difficult, if not impossible, to gauge the effectiveness of the IWTP to treat these metals. Moreover,

use of the materials balance to track "unauthorized" dumping of chromium/cadmium solutions into the IWTP wastewater collection system requires increasing both the number of chemical samples taken and the sampling frequency.

Due to these limitations, it is recommended that at a minimum, triplicate samples for chemical analyses should be taken over various times. Evaluation of a larger chemical data set would allow estimation of a "true" average metal concentration while providing insight into the range of metals (*i.e.*, mean ± standard deviation) expected in a particular flow. Moreover, due to possible short term flow aberrations (*e.g.*, significant increases/decreases in industrial activity, storm events *etc.*), average monthly rather than average weekly flow rates should be used in estimating annual flow rates.

References

1. RCRA Regulations and Keyword Index (1997) Elsevier Publishing

2. US Environmental Protection Agency "Guides to Pollution Prevention - The Metal Finishing Industry" EPA/625/R-92/011

NUMERICAL MODELING OF PHYSICAL CONSTRAINTS ON IN-SITU COSOLVENT FLUSHING AS A GROUNDWATER REMEDIAL OPTION FOR OPERABLE UNIT 1, HILL AIR FORCE BASE, UTAH

William E. Sanford
Assistant Professor
Department of Earth Resources

Colorado State University
322 Natural Resources Building
Fort Collins, CO 80521-1482

NUMERICAL MODELING OF PHYSICAL CONSTRAINTS ON IN-SITU COSOLVENT FLUSHING AS A GROUNDWATER REMEDIAL OPTION FOR OPERABLE UNIT 1, HILL AIR FORCE BASE, UTAH

William E. Sanford
Assistant Professor
Department of Earth Resources

ABSTRACT

In-situ cosolvent flushing has been suggested and tested as a possible method of remediation of the groundwater within Operable Unit 1 (OU1) at Hill Air Force Base, Utah. Small-scale tests were performed at the lab and the results of those results were used in numerical models to determine the constraints placed on scaling the treatment method up to the field-scale. Some of the physical limitations include the limited thickness of the saturated zone beneath OU1. The modeling package FEMWATER included as part of the Department of Defense Groundwater Modeling System was used. This package allows for the modeling of transport of solutes with variable densities. The modeling results suggest that the areal size of the field-scale system would be limited to approximately 0.09 hectares, requiring the need for over 30 such systems to treat the entire area of OU1.

NUMERICAL MODELING OF PHYSICAL CONSTRAINTS ON IN-SITU COSOLVENT FLUSHING AS A GROUNDWATER REMEDIAL OPTION FOR OPERABLE UNIT 1, HILL AIR FORCE BASE, UTAH

William E. Sanford

INTRODUCTION

Several small-scale investigations of *in situ* remediation of light non-aqueous phase liquid (LNAPL) contamination within Operable Unit 1 (OU1) of Hill Air Force Base (HAFB), Utah, have been conducted over the past 2 years (e.g., Sillan et al., 1997). These tests were performed within hydraulically-isolated, 15-m^2 rectangular test cells. Several of the methods proved successful at the scale of the test cells where flow was contained within the walls of the cell. The LNAPL contamination in OU1 occurs over an areal extent greater than 2.8 hectares in an aquifer which has a typical saturated thickness of less than 2 m (Montgomery Watson, 1995). Therefore, it is highly desirable from a cost point-of-view to scale-up the remediation to cover as much area as possible. However, the relatively thin saturated zone places a hydraulic constraint on the areal extent over which the large-scale tests can be performed. The purpose of this study was to model the hydraulic considerations for the scaling-up of the *in situ* remediation tests using the Department of Defense Groundwater Modeling System (GMS) Version 2.0 as an interface with the numerical flow and transport model FEMWATER (Lin et al., 1996).

Background

The hydrostratigraphy of the portion of OU1 that lies within the boundaries of HAFB can be summarized as consisting of two distinct sedimentary units (Montgomery Watson, 1995):

- An upper sand and gravel unit that consists of fine to coarse, clean to silty sand interbedded with gravel. This unit ranges in thickness from 0 to 19 m, with an average thickness of 9 m. The saturated thickness of the unit ranges from 0 to 3 m, with 1 m being the average.

- A lower silty-clay unit that consists primarily of clays and silts containing thin stringers of very fine sand. The lower unit is approximately 60 m thick and is saturated over most of its depth.

See Montgomery Watson (1995) for a more complete description of the hydrogeology of OU1. The LNAPL contamination, which floats on the groundwater, occurs within the upper sand and gravel unit, producing a layer up to 0.30 m in some locations within OU1.

Several studies funded by the Strategic Environmental Research & Development Program (SERDP) and the Environmental Restoration and Management Division, HAFB, were carried out to test various methods of subsurface characterization and LNAPL remediation within OU1. Data used in this report are from tests performed by the University of Florida in which in-situ co-solvent flushing was investigated as a type of remediation for the LNAPL (Sillan et al., 1997). The test was performed within a 4.3 m x 3.5 m test cell that was hydraulically isolated by the use of interlocking sheet pile and had 4 injection wells, 3 extraction wells, and numerous monitoring wells. The work performed within that study involved collection of soil cores and running tracer tests, including partitioning tracers (see Jin et al., 1995, for information on partitioning tracer tests), to identify NAPL distribution, estimate NAPL volume and to characterize the subsurface hydraulic properties, and flushing with a co-solvent mixture consisting of ethanol, *n*-pentanol, and water. The analysis of the co-solvent flushing indicated that on average, greater than 85% of the NAPL was removed.

The results from the co-solvent study raised several concerns that need to be addressed before the process can be scaled-up. First, the alcohol-water mixture is less dense than the groundwater, which results in an over-riding effect during the initial period of injection, and an under-flow of water during the initial period of flushing of the co-solvent mixture (Sillan et al., 1997). Second, about 40,000 L of the co-solvent mixture was flushed through the test cell. All recovered fluids with alcohol concentrations greater than 1% were sent off-site for incineration. A larger-scale system would produce much more waste water that would require expensive disposal. And third, the thin saturated thickness (~1 m) of the aquifer will limit the areal extent of a larger scale flushing with an injection/extraction (line-drive) type configuration. It was with these concerns in mind that the numerical modeling presented within this report was performed.

METHODS

The investigation into the hydraulic concerns of scaling-up the *in situ* remediation schemes from the test cell size was undertaken with the Department of Defense Groundwater Modeling System (GMS), which is used as a pre- and post-processor interface for several popular groundwater flow and transport modeling packages. The package used here was FEMWATER, a three-dimensional finite element model for simulating density dependent flow and transport through porous media (Lin et al., 1996). Three scales of investigation were undertaken: 1) the 4.3 m x 3.5 m test cell used by Sillan et al. (1997); 2) a field-scale line-drive injection/extraction well configuration with a spacing of 25 m; and 3) a small scale simulation (2 m in length) to investigate the applicability of FEMWATER to model the effects

flushing with tracer free water. Tracer distribution throughout the cell and breakthrough curves of the tracer at the extraction wells were determined and compared with the results from Sillan et al. (1997).

Field-Scale Modeling

For the field-scale modeling, several combinations of grid size, well spacing, number of wells, and pumping rates were tried until one configuration provided reasonable results (see Discussion section, below). The field-scale model presented here was a region 100 m wide by 100 m long by 4 m depth and used the same aquifer material as described for the test cell model. This grid consisted of 10780 nodes and 9180 rectangular elements with the grid spacing closer around the wells (Figure 3). The model used a line of 3 injection wells and 3 extraction wells located 25 m apart (Figure 3). In order to force all tracers injected to flow towards the extraction wells, 3 hydraulic control wells were located 5 m directly behind the 3 injection wells (Figure 3). The total flow rates were 3.71 m^3/hr for the hydraulic control wells, 2.48 m^3/hr for the injection wells, and 6.19 m^3/hr for the extraction wells. Again, because of the constraints of the modeling package, the hydraulic control wells were represented by 2 nodes each, the injection wells by 3 nodes each and the extraction wells by 1 node each (located at the bottom of the aquifer; Figure 4).

As for the test cell model, the steady-state flow conditions were established first, starting with an initial saturated thickness of 2 m and a constant head boundary condition around the entire domain of 2.1 m. The tracer simulation was performed with a 24-hour injection of a constant concentration followed by a 456-hour flush of tracer-free water to recover the injected tracer.

the experiment). This is interpreted as a function of how the injection is modeled and that the natural heterogeneity of the aquifer at OU1 was not incorporated into the modeling. The important result is that the shape of the breakthrough curve is nearly identical to that of the actual tracer test. For both the model and the test, the peak arrival time occurred at about 1.5 pore volumes and, after 4 pore volumes of flushing, the concentration of the tracer was well below 0.1. This indicates that the modeling package can accurately model the flow and transport of non-reactive solutes through the test cell using the hydraulic parameters given in Sillan et al. (1997).

Field-Scale Modeling

The spacing of the wells and the pumping rates used in the modeling of the field-scale system were decided on by running various different scenarios. It was found that with larger spacing, it would be difficult to get all the injected fluid to be recovered in the extraction wells. This was deemed important to minimize costs of injection chemicals and well installation and to prevent the contamination of other regions of the aquifer. The pumping rates were chosen based on rates which prevented wells from drying out during the tests. The well configuration that was decided upon is described above.

Once the configuration was decided upon, it was necessary to determine if the tracers injected will all arrive at the extraction wells and to determine an estimate of how long it would take to flush the system. The results of the tracer experiment are shown in Figures 6 and 7. Figure 6 is the breakthrough curve of the tracer for the middle well of the extraction set. Figure 7 illustrates the areal coverage of the tracer in the space between the injection and extraction wells. As can be seen in Figure 6, the maximum concentration in the extraction well was a $C/C_0 = 0.033$. Even after flushing the aquifer with tracer-free water for 19 days, there was still some remaining in the aquifer. Figure 7 shows that there is good coverage of the area between the wells. A balance on injection mass versus extracted mass of tracer

indicates that 98% of the tracer was recovered. This indicates that with this well configuration, all the co-solvent mixture that would be injected can be recovered.

The first recovery of tracer occurred about 1 day after the injection ended, and continued until the end of the modeling period (19 days). Approximately 60 m^3 of tracer was injected over the 1-day period. With the extraction wells pumping at 6.19 m^3/hr, greater than 2800 m^3 of water was extracted that would need treatment and disposal. There was still tracer remaining when the modeling stopped, so the actual amount will be greater.

It is estimated that the area that can be treated by this configuration is 0.09 hectare. Since OU1 comprises 2.8 hectares, it would require over 31 such treatment systems. One advantage of the line-drive configuration is that the various wells can be reused as extraction, injection, and hydraulic control wells as needed as the neighboring area is to be remediated.

Density Effects

Shown in Figure 8a are the results of modeling the displacement of a fluid by another with a relative density of 0.77. It is clearly seen that the less dense fluid rides over top of the fluid with greater density originally in the medium. Figure 8b shows the displacement of a fluid with relative density of 0.77 by a fluid that has a relative density of 1. Significant stratification developed with time, with the greater density being at the bottom. Both cases illustrate that significant stratification can arise when displacing a fluid of one density by one with a different density.

In Figure 8c are the results of a two-dimensional laboratory experiment presented in Sillan et al. (1997) in which the effects of density contrasts were illustrated. As can be seen, there was indeed a development of stratification under conditions similar to those modeled above. The results of the numerical modeling when compared to the 2-D experiment illustrate that the model used, FEMWATER, is adequate for simulating the effects of density contrasts arising during co-solvent flushing.

SUMMARY AND IMPLICATIONS

The results of this study are: 1) FEMWATER (with the GMS interface) can be used to reproduce the hydraulic performance of the cell-scale tracer tests and, therefore, provide accurate predictive capabilities for examining the hydraulic limitations of scaling-up the co-solvent flooding to the field scale; 2) FEMWATER is capable of modeling the flow conditions resulting from the density contrast between the alcohol used in the flooding and the groundwater; 3) To order to maintain a saturated system at all times, the injection and extraction wells can be located no farther than 25 m apart; and 4) Hydraulic control wells are needed to force all the injected co-solvent to flow to the extraction wells.

The last 2 results listed above place constraints on the application of co-solvent flooding at the field-scale in OU-1.

REFERENCES

Jin, M., M. Delshad, V. Dwarakanath, D. C. McKinney, G. A. Pope, K. Sepehrnoori, C. E. Tilburg, and R. E. Jackson, 1995. Partitioning tracer test for detection, estimation, and remediation performance assessment of subsurface nonaqueous phase liquids. *Water Resources Research* 31(5), 1201-1211.

Lin, H. C., D. R. Richards, G. T. Yeh, J. R. Cheng, H. P. Chang, and N. L. Jones, 1996. FEMWATER: A three-dimensional finite element computer model for simulating density dependent flow and transport. U.S. Army engineer Waterways Experiment Station Technical report, 129 pp.

Montgomery Watson, 1995. Final comprehensive remedial investigation report for Operable Unit 1, Hill Air Force Base, Utah. Montgomery Watson, Salt Lake City, Utah.

Sillan, R. K., M. D. Annable, and P. C. S. Rao, editors, 1997. In-situ cosolvent flushing for enhanced solubilization of a complex NAPL: Comprehensive field-scale evaluation. SERDP Draft Final Report.
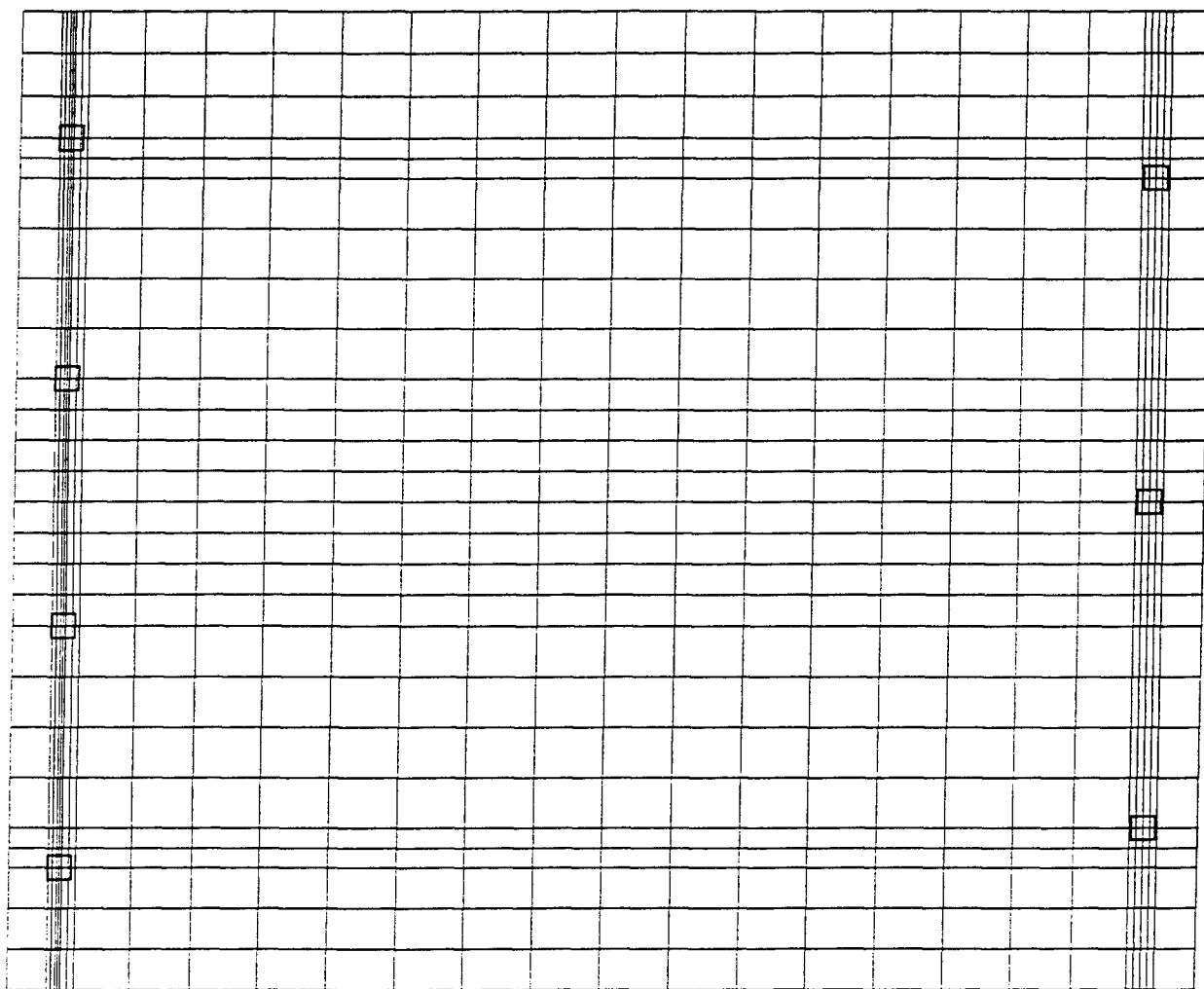
Figure 1. Plan view of the grid used to model the test cells. Domain size is 4.3 m X 3.5 m. Squares on the left denote injection modes and squares on the right denote extraction nodes.
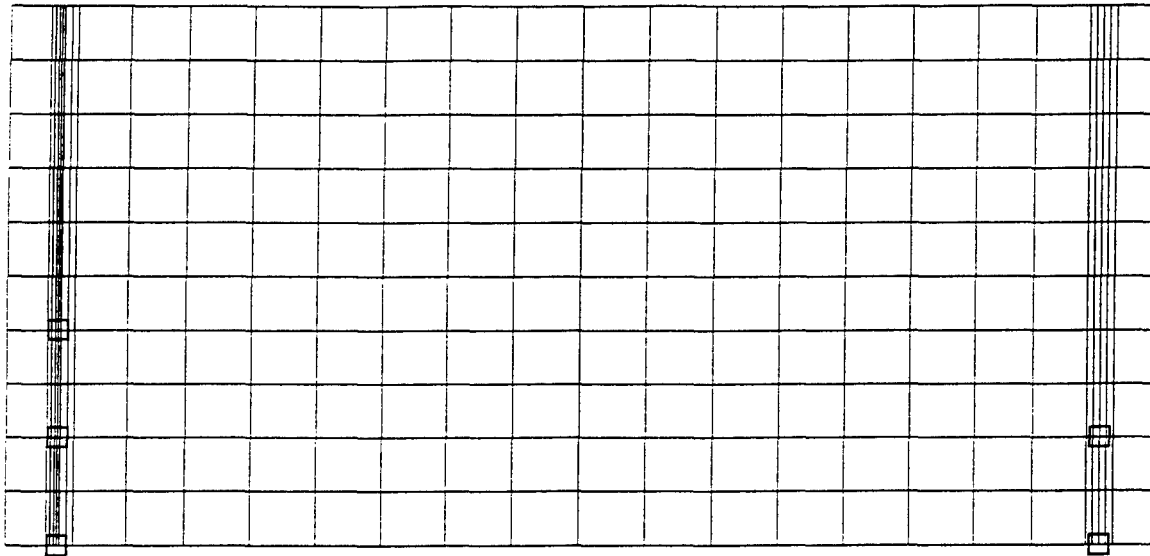
Figure 2. Cross-section view of grid used to model the test cells. Domain size is 4.3 m by 2 m deep. Locations of injection and extraction nodes as in Figure 2.
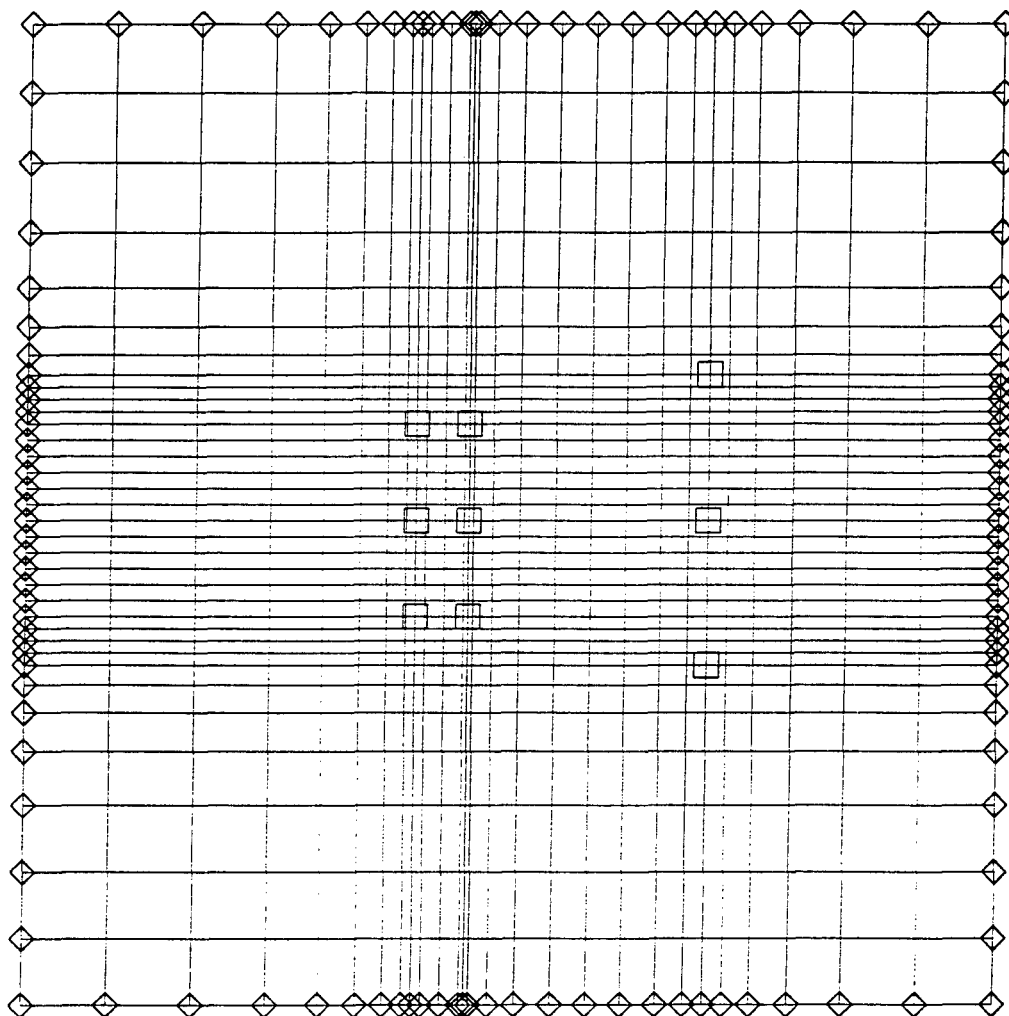
Figure 3. Plan view of grid used to model the field-scale system. Domain size is 100 m X 100 m. The squares located farthest to the left denote hydraulic control nodes. Squares immediately to the right of the hydraulic control nodes denote injection nodes. Squares to the right denote extraction nodes.
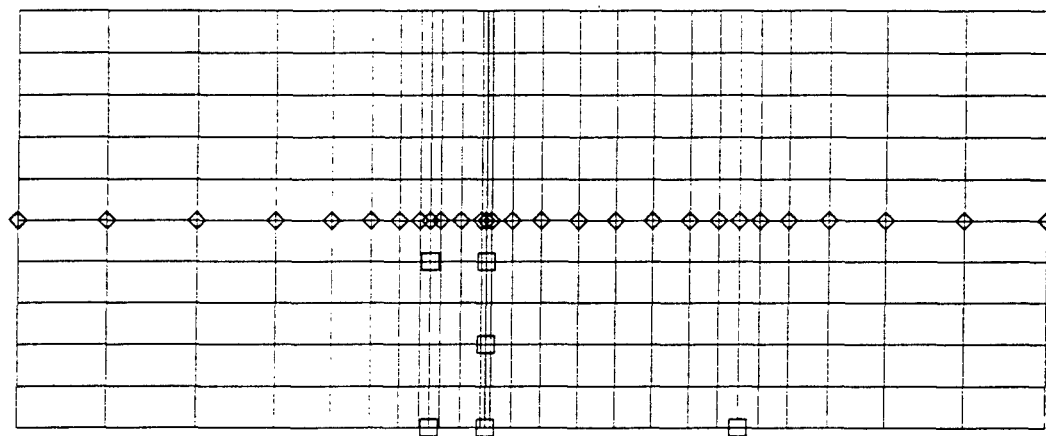
Figure 4. Cross-section view of grid used to model the field-scale system. Domain size is 100 m X 4 m. Locations of hydraulic control, injection, and extraction nodes as in Figure 3.
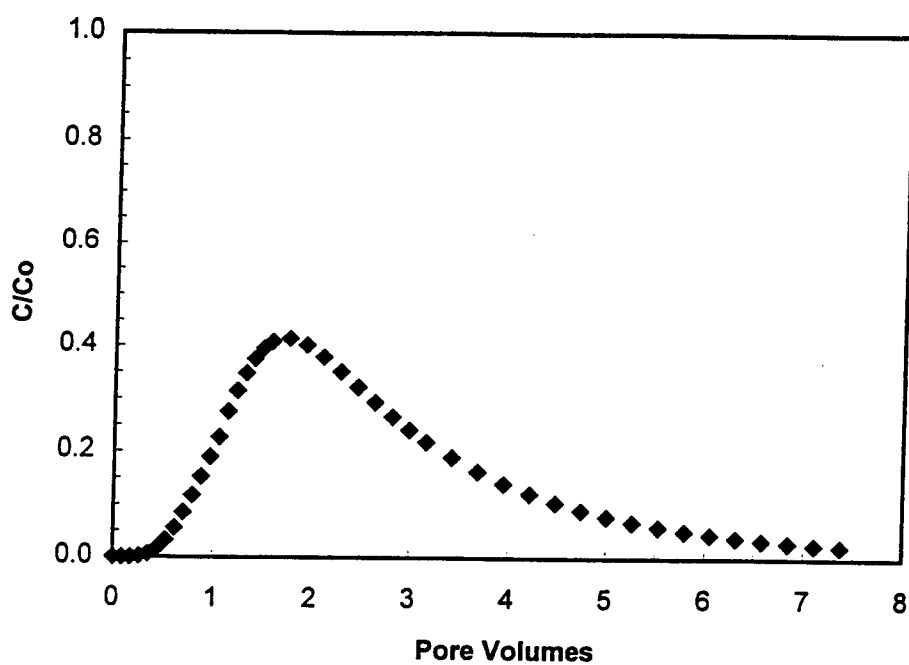
Figure 5. Breakthrough curve of tracer from center extraction well of the test cell model.
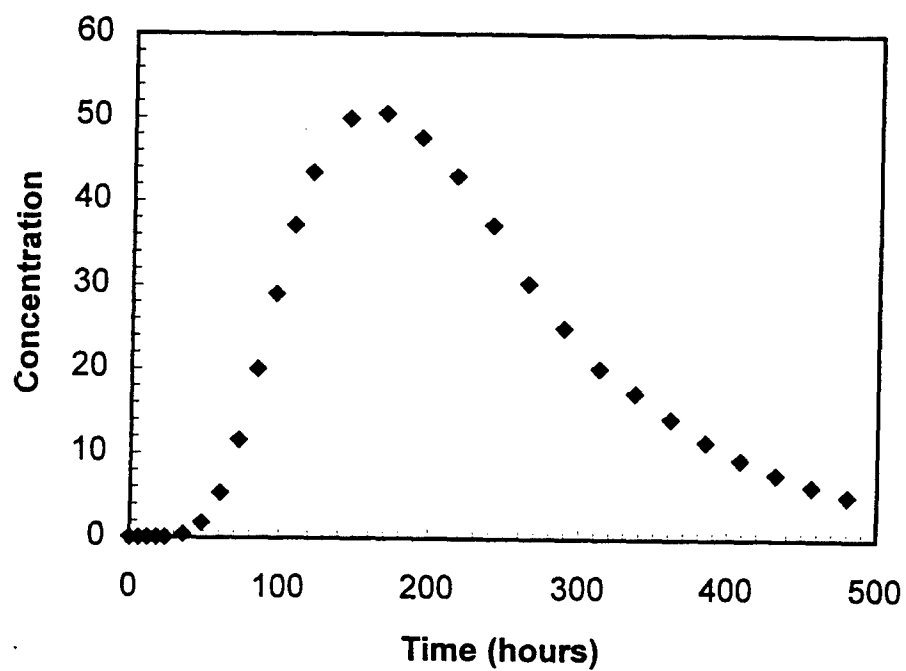
Figure 6. Breakthrough curve of tracer from center extraction well of the field-scale model. Initial injection concentration was 1000 units.
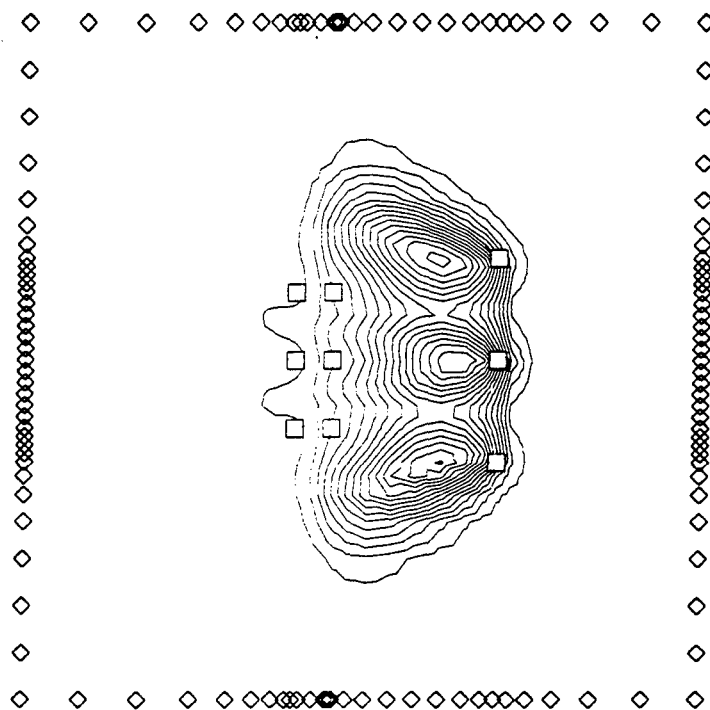
Figure 7. Plan view of field-scale model illustrating areal coverage of aquifer by the tracer 7 days after a 24-hour injection of tracer.
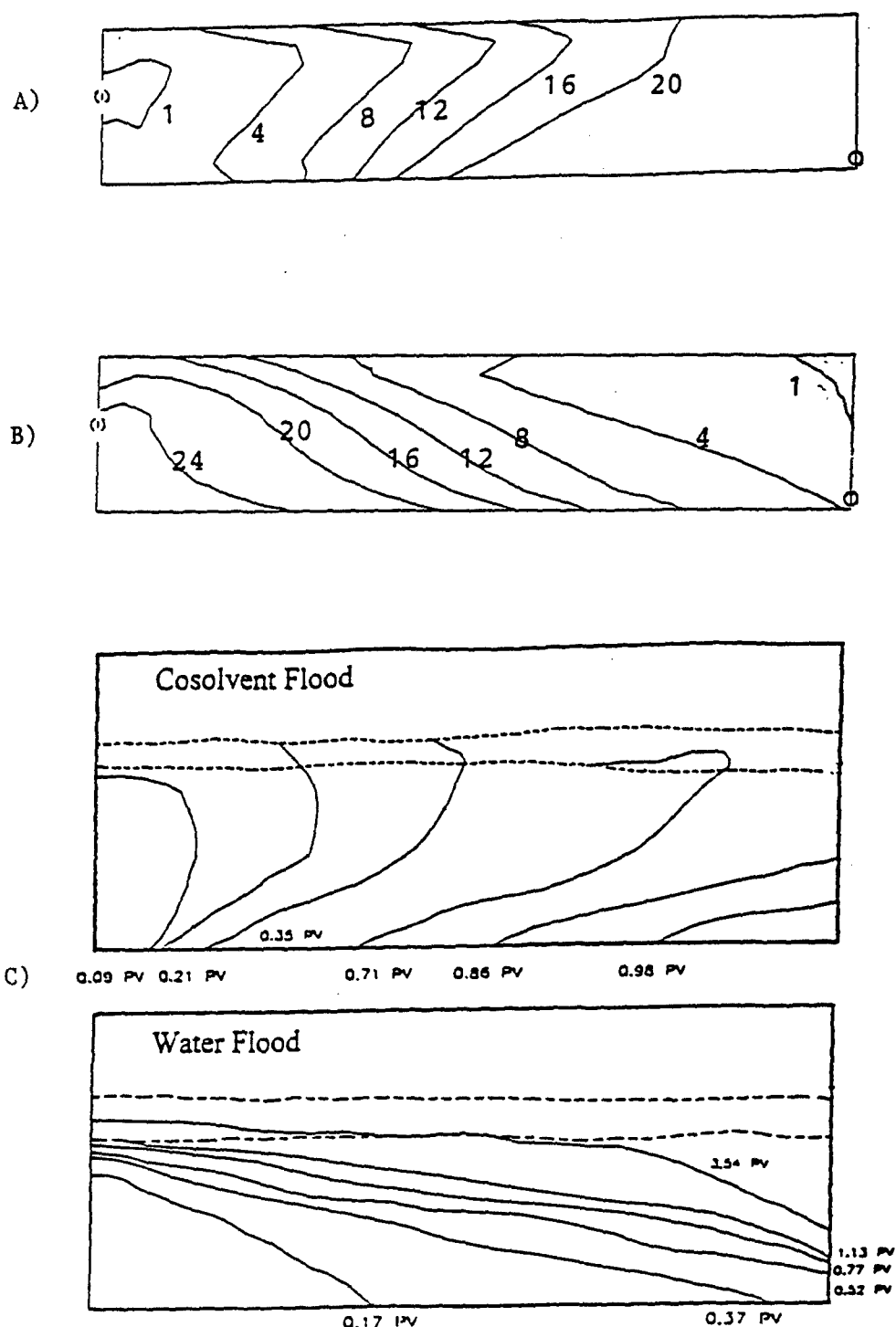
Figure 8. Plots illustrating the effectiveness of using FEMWATER to model density dependent flow. **A)** Modeled transport of a less dense fluid injected into a more dense fluid after 12 hours. **B)** Modeled transport of a more dense fluid injected into a less dense fluid after 12 hours. Contours in **A** and **B** are g/L of concentration. **C)** Results presented in Sillan et al. (1997) of cosolvent and water flooding of a laboratory test column. Contours represent leading edge of plume after the injection of the labeled number of pore volumes.

Fracture Analysis of the F-5, 15%-SPAR Bolt

Sophia Hassiotis
Assistant Professor
Department of Civil Engineering

University of South Florida
Tampa, FL 33620-5350

FRACTURE ANALYSIS OF THE F-5, 15%-SPAR BOLT

Sophia Hassiotis
Assistant Professor
Department of Civil Engineering
University of South Florida

Abstract

The 15%-spar bolt that connects the wing to the fuselage of the F-5 airplanes was observed to fail in fatigue at a relatively short flight-time. Bending stresses developed at the base of the bolt allow for a fast crack growth. The bending stresses can be reduced if the tolerance between the bolt and the bushing is decreased. A boundary-element program was used to investigate the stress concentration factors and crack growth at the base of the bolt. The results of this analysis were fed into a damage-tolerance code to find time-to-failure for different tolerances between the bolt and the bushing. It was found that if the tolerance decreases from 0.012 inches to 0.003 inches the hours-to-failure increase from 1,800 to 50,000.

FRACTURE ANALYSIS OF THE F-5, 15%-SPAR BOLT

Sophia Hassiotis

Introduction

This report contains the damage tolerance assessment for the 15%-Spar Bolt of the F5 aircraft. This trunnion bolt is designed to carry the primary shear between the wing and the fuselage. Many of these bolts have been observed to fail in fatigue at fewer than 1200 flight hours, which adds considerably to the maintenance cost and affects the safety of the aircraft. The objective herein is to assess the durability of the bolt under a different design.

The original design of the bolt provides a maximum clearance of 0.012 inches between the bolt and the bushing. Such clearance allows for the development of bending stresses that were not taken into account in the original design, thus resulting in an un-conservative design of the bolt. In-service inspection has indicated that the bending stresses developed due to the clearance have contributed to crack growth at the base of the bolt threads, resulting in ultimate failure due to fatigue fracture.

The objective is to investigate the damage tolerance of the bolt, allowing for smaller clearances between the bolt and the bushing. Namely, we arrive at the hours-to-failure of the bolt when the clearance is reduced to 0.009, 0.006, or 0.003 inches.

Fracture Analysis

The fracture analysis for the bolt was carried out in two steps. First, the stress-intensity factors at the base of the threads were calculated using a crack-growth computer code . These factors were then used as an input into a life-prediction computer code.

Stress Intensity Factors Using FRANC3D

The boundary element code FRANC3D [1] was used to obtain the stress intensity factors at the base of the bolt. A detailed analysis was deemed necessary because the stress field at the point of interest is complicated due to the presence of (1) the interface of the steel bolt with the aluminum fuselage, (2) the tensile pre-stress applied to the bolt, and (3) the spectrum of shear forces the bolt is designed to carry.

A rough sketch of the bolt dimensions (in inches) is shown in Figure 1. The shear stress developed at the wing/fuselage interface passes through section A-A. It is this shear force that results in the bending stresses that contribute to the fracture of the bolt. Figure 2 shows the modeling of the shear force. In addition, the torque applied to the bolt during assemblage was modeled by the tensile force, P. It was assumed that the torque produces a stress of 50ksi at the base of the bolt. The crack shown at the base of the bolt was grown using a "far-field" shear traction, V=0.5ksi.

The meshed model is shown in Figure 3. Since the bolt/fuselage connection as modeled by a continuous section, it became necessary that the steel and aluminum regions be separated artificially. FRANC3D does not supply contact elements. Therefore, in order to model the steel and aluminum surfaces, which are expected to separate when subjected to tension, a crack was introduced at the interface, as shown in Figure 3.

Figure 4 shows the crack developed at the base of the bolt. The crack was input as a 3-dimensional penny-shaped crack with original size of 0.05 in (in the c-direction). The analysis was
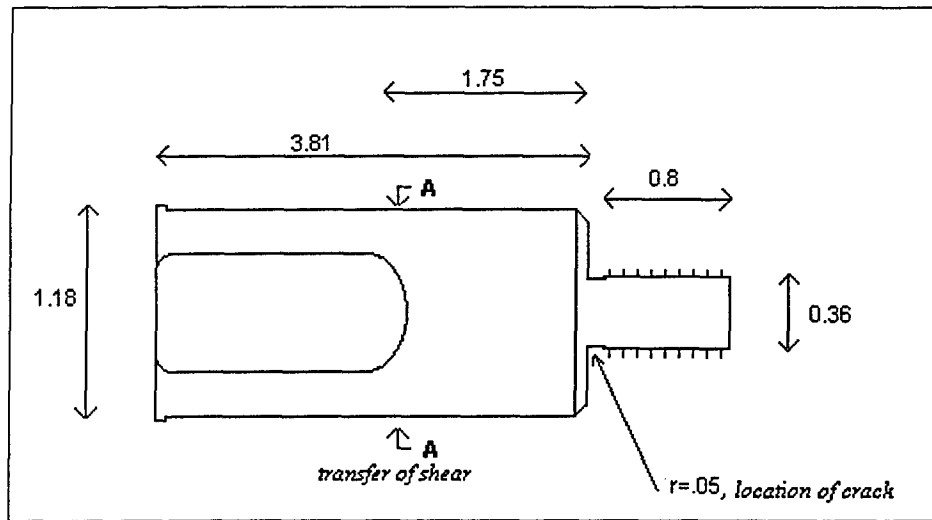
Figure 1: 15%-Spar, trunnion bolt.
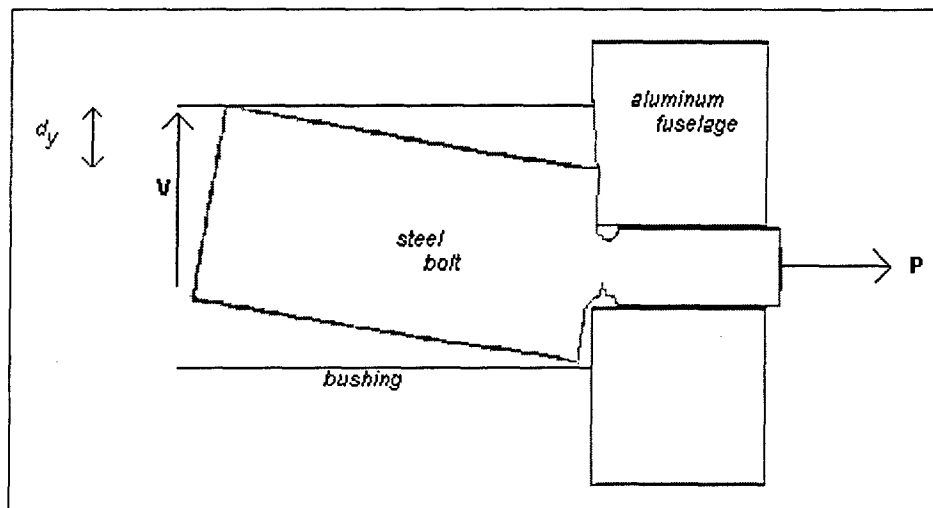
Figure 2: Pre-stress, P, and shear force, V, on the bolt ith crack shown at the base. Magnification 1:100.
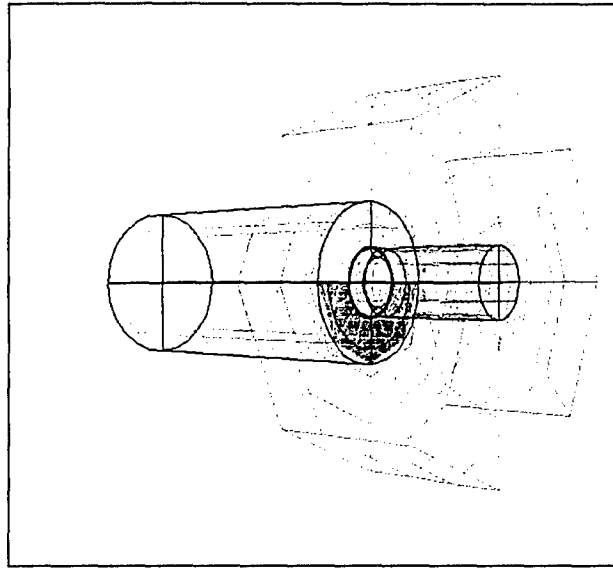
Figure 3: Crack introduced to separate the steel and aluminum.

repeated for several crack sizes to obtain a relationship between the crack length and the stress intensity factors. A table of beta factors was derived and summarized in Table 1.

Table 1. Beta factors

Crack length, c, (in)	beta factors
0.054	61.0
0.08	56.0
0.12	48.8
0.17	49.3
0.20	58.0
0.24	60.0
0.30	82.0

Damage Tolerance Using CRACKS95

The beta factors calculated in the boundary element analysis were used as an input in the CRACKS95 code to obtain hours-to-failure under the load spectrum registered in the F5. The following data was used:

Material for the Bolt: PH13-8Mo, H1000 Forg., Lo Humid Air, L-T. This material has a fracture toughness of $98ksi\sqrt{in}$ and a yield stress of 200ksi.

Initial Crack Size: 0.05in.(in the c-direction).

Final Crack Size: 0.3in. At crack sizes larger than 0.3in the net area cannot support the loading and yielding will occur.

Retardation Model: Generalized Willenborg (shut-off ration 2.3)

Spectrum: The spectrum used was created by SwRI's Aircraft Sequence Development Program. F-5E DACT data was scaled to the limit load for falling wing at sea level, 5.22 g at 0.95 Mach. The spectrum was clipped at the stress required to deform the bolt to the tolerance of the bushing, since the maximum stress on the loading spectrum that contributes to the fracture of the bolt base is a
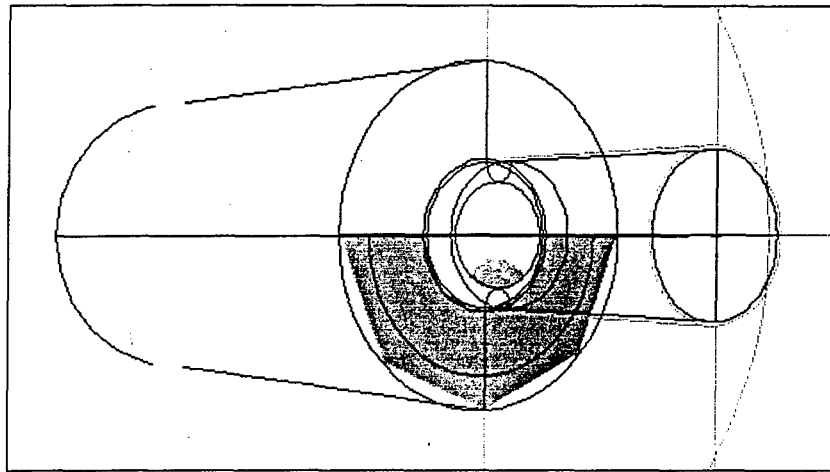
Figure 4: Crack growth at the base of the bolt.

function of this clearance. FRANC3D was used to calculate the shear traction V that contributes to displacing the end of the bolt an amount d_y(see Figure 2). Damage analysis calculations were conducted for four different bolt-to-bushing clearance tolerances: 0.012, 0.009, 0.006, and 0.003 inches. These tolerances allow for a vertical displacement d_y of 0.006, 0.0045, 0.003, and 0.0015 inches respectively, assuming that the bolt is perfectly centered within the bushing. It was found that shear tractions between 0.90 to 0.35 ksi are enough to close the gap between the bolt and the bushing. Table 2 is a summary of the maximum stress used to clip the stress spectrum for the different tolerances.

Table 2. Maximum stress required to close the bolt/bushing gap.

Tolerance (in)	Vertical Disp. (in)	Max Stress (ksi)
0.012	0.006	0.90
0.009	0.045	0.72
0.006	0.003	0.54
0.003	0.0015	0.35

Results

The hours-to-failure depend on the tolerance between the bolt and the bushing, and as expected a smaller tolerance will lead to a more conservative design. Table 3 summarizes the hours-to-failure vs the bolt/bushing clearance. These values are plotted on Figure 5. On the same figure, results found from a simplified analysis using AFGROW [3] are also presented.

Table3. Hours-to-failure

Tolerence (in)	hours-to-failure
0.012	1,820
0.009	2,320
0.006	6,600
0.003	49,900
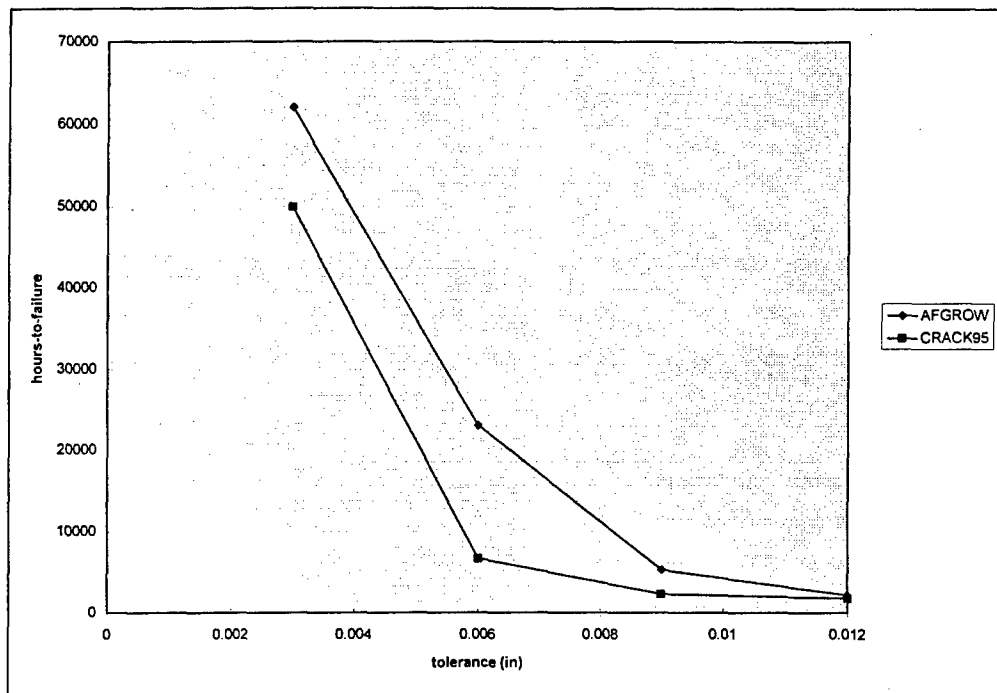
Figure 5: Hours-to-failure vs tolerance

Conclusion

The stress intensity factors due to a crack at the base of the bolt were calculated using a boundary-element program. These factors were used to calculate the life expectancy of the bolt. It is evident by this calculation that the hours-to-failure increase significantly if the clearance between the bolt and the bushing is reduced. Therefore, reducing the tolerance between the bolt and the bushing to the smallest amount required for assemblage constitutes an improved design.

Other factors that would affect the life of the bolt include the possible presence of torsional or axial forces in addition to the shear forces. Such additional stresses would decrease the hours-to-failure significantly. The modeling herein does not provide for the existence of such forces because data that could support this claim has not been collected. Therefore, the hours-to-failure that were calculated in this analysis provide an un-conservative life expectancy if tractions other that the shear exist.

Additional areas of possible change in design that could increase the life of the bolt include the use of materials with higher fracture toughness, a smoother transition at the base of the bolt (an increase of the r=0.05in), or a tapered design to eliminate any spacing between the bolt and the bushing. A detailed analysis of such changes could be the subject of further analyses.

Acknowledgement

References

[1]FRANC3D, Boundary element program for crack propagation, Cornell University, Ithaca, NY.

[2]CRACKS95, Damage Tolerance Analysis Program, University of Dayton, Dayton, OH.

[3]AFGROW, Air Force Crack Propagation Analysis Program, Version 3.82 (1997)

A SIMPLE, MULTIVERSION CONCURRENCY CONTROL PROTOCOL FOR INTERNET DATABASES

Devendra Kumar
Associate Professor
Department of Computer Sciences

The City College of New York
Convent Avenue at 138th Street
New York, NY 10031

A SIMPLE, MULTIVERSION CONCURRENCY CONTROL PROTOCOL FOR INTERNET DATABASES

Devendra Kumar
Associate Professor
Department of Computer Sciences
The City College of New York

Abstract

We present a concurrency control protocol that would be particularly suitable for large, distributed databases accessible via the internet. The main function of a concurrency control protocol is to make sure that the database remains consistent and coherent in spite of concurrent access to it by different users who are reading and updating it. Internet based distributed database systems have special requirements, e.g., having a large number of read-only transactions, and an expectation that such transactions should almost always be given a consistent set of data; letting a read-only transaction read an inconsistent set of data or only part of the data it needs, and later rolling it back is unreasonable--- especially when it is an interactive transaction involving a human user, since it is difficult and annoying for a human user to ``unlearn'' what he read previously. Many traditional protocols do not satisfy this expectation. Also, a read-only transaction expects a fast response time, so it should not be normally delayed or rolled back simply due to the peculiarities of the concurrency control algorithm being employed. Unfortunately, most traditional protocols would do just that ---. delaying a read-only transaction due to unavailability of certain locks or otherwise rolling it back for a variety of reasons. Also, some of the read operations by an update transaction may be ``useless'' in the sense that they won't influence the computation of write values by the transaction. For example, a user may surf other parts of the database just to get a general idea of the contents. Traditional protocols do not take advantage of these reads being ``useless'' (i.e., insignificant) in optimizing performance; they simply assume that all reads are used in computing the write values in ways unknown to the system. Our concurrency control protocol handles all these issues effectively. Moreover, our protocol is conceptually simple, and easy to implement. It is quite flexible, allowing several variations that can be used for performance enhancements or user convenience. For example, a transaction is free to check if a new version of the database has been created after its previous read operations, and hence it may wish to discard the older version and may read again from the newer version. On the other hand, sometimes a read-only transaction may indeed prefer to read an older version of the database even though newer versions are available. In terms of common classification of concurrency control protocols, ours is a *multiversion, optimistic* protocol.

A Simple, Multiversion Concurrency Control Protocol For Internet Databases

Devendra Kumar

1. Introduction

In a distributed database, the data is distributed over several sites. Typically, different sites are geographically widely distributed, but sometimes they may also be part of a local area network. Different transactions, originating at various sites, operate on this data concurrently. The database is assumed to be in a consistent state initially. (Informally, consistent state means that the data is coherent and meaningful and is ready to be used by new users; it is not the case that part of data is old and part is new or still not written yet, and therefore the data as it exists does not make sense together.) Also, it is assumed that each transaction satisfies the following property: starting from any consistent state, if the transaction is executed alone, then the resulting database state at the end of this execution will also be consistent. Clearly, starting from the initial consistent state, any serial execution of one or more transactions will leave the database in a consistent state. However, if the transactions were to run concurrently, with arbitrary interleaving of their operations (such as reading or writing values to variables in the database), the resulting state may not be consistent. There are additional problems caused by uncontrolled, concurrent execution of transactions. Dealing with these issues is the job of the *scheduler* which uses a *concurrency control algorithm* to do this job.

Also, various kinds of failures may occur in the system, e.g., a transaction may explicitly ask to *abort* itself due to some undesirable conditions, or the operating system might crash, or the database management system may abort the transaction due to a deadlock, or the main memory may fail, or the disk may crash, etc. The *recovery manager* deals with such failure conditions; in particular it makes sure that the database remains consistent in spite of such failures. The recovery manager does its job in coordination with the scheduler. The issues of concurrency control and recovery have been well studied; we refer the reader to any of several excellent works [Bernstein87, Ceri84, Korth95, Bell92].

A variety of approaches for concurrency control have been presented in the literature. For a given application, performance of various approaches may vary significantly. Several different factors might influence the choice of an appropriate approach for a given application, e.g., the degree of concurrency, the degree of conflict among transactions in terms of updating the same data items, etc.

In this paper, we consider concurrency control in a special environment--- a large, distributed database, accessible to a large number of users via the internet. Examples of such applications include military information systems, healthcare databases, multinational business databases, etc. We expect the number of such databases to grow over the coming years, for a variety of reasons. First, most large organizations seem to have accepted the internet technology as an important technological tool to conduct their business, and are therefore providing internet access to their databases to their employees, customers, and other interested parties. Moreover, a growing population across the globe is finding the internet, and computers in general, to be interesting and useful. This encourages more people to access a given database. In such an environment, there are special requirements and expectations from the database management system. Below we present a set of such requirements. In Section 2, we present our concurrency control algorithm, called the Internet Database Protocol (IDP), that addresses these requirements. To keep the algorithm and its discussion simple, we will assume that the scheduler is located in one central site, even though the database itself is distributed. The algorithm can be easily extended to the case of distributed scheduler using the well known idea of two phase commit protocol. Essentially, in this protocol a commitment is made to an update only after all the sites have agreed to commit to that update; this requires a synchronization among the sites. Focusing on the central scheduler allows us to look at the fundamental characteristics of the protocol. Similar approach is commonly taken in the literature, e.g., in [Bernstein87, Ceri84, Korth95, Bell92], a protocol is discussed in the centralized case first, and is later extended to the distributed case. Our algorithm is quite simple and flexible. In Section 3, we will suggest several simple variations to the algorithm.

Our algorithm is essentially an *optimistic, multiversion protocol* [Bernstein87, Korth95] with one important difference --- we maintain version numbers for the database as a whole, not for the individual items of the database. Each version of the database is a consistent version, and hence can be read by a transaction without causing a retrieval inconsistency [Bernstein87, Bell92]. This simple view results in tremendous benefits in terms of handling special requirements of internet databases, and the resulting algorithm is also very simple and flexible. In traditional multiversion protocols, the latest version numbers of different data items is different; so at any given moment, it is difficult to take a snapshot of one consistent, coherent database while other users are updating it. In our case, one simply has to read all data items corresponding to the same version number. In Section 4, we compare our protocol with the multiversion protocol discussed in [Korth95] in terms of performance, flexibility, and simplicity.

General Characteristics and Requirements of a Class of Internet Databases

Here we consider a large, distributed database that is accessible to a large number of users via the internet, intranet, or similar technology. As the number, size, and capabilities of these systems grow, a set of common characteristics and common requirements regarding their usage and system performance expectations seem to be emerging that

are relevant to the database designer. In this Section, we list some of the main such characteristics and requirements that are over and above those commonly accepted for the more traditional distributed database systems. In other words, we do not repeat here what is generally known for common distributed database environments. Of course, exact usage patterns and requirements among various internet databases will vary; therefore, for a specific system, some of these observations might not apply, or there may be additional requirements.

1. The system will have a large number of users, many of whom are simply read-only users, i.e., they do not have an interest in, or the authorization to, update the database. These users will generate a large number of concurrent, *read-only transactions.* A read-only transaction is one which does not update the database. For example, a test engineer may wish to find out what test equipment and test procedures are needed to test a particular aircraft. In contrast, an *update transaction* is one that updates the database; it may or may not read data from the database.

2. A read-only transaction will not always be able to declare at its beginning that it is indeed a read-only transaction. For example, a user may read some data with the intention of updating the database, may even write some updates in his own workspace, but later may decide not to update the database after all. Thus it turns out to be a read-only transaction in the end. The database designer should not insist that a read-only transaction declares itself to be as such at its beginning.

3. Most read-only transactions expect the system to give them a *consistent* set of data *in the first instance.* (We say that the data read by a transaction is *consistent* if it came from a consistent version of the database that existed at some point in time.) It is generally not acceptable for the system to first let them read *inconsistent* data and then later tell them that the data was inconsistent and therefore they need to start over again. Several protocols do not satisfy this requirement. For example, in the multiversion protocol discussed in [Korth95], an update transaction T1 may be aborted and as a result a read-only transaction T2 that previously read data written by T1, may also have to be aborted. There are several reasons for this requirement. First, it is sometimes difficult for a human user to unlearn what he has learnt from the data he has already read. It also causes delays which may be annoying to the human user. Moreover, note that in such an environment, *external writes* [Bernstein87] are common. A user may read the inconsistent data and may use it in some other application, fax it to somebody, or take decisions based on it--- all this has to be reversed when the transaction is rolled back. Sometimes an external write may be irreversible; in such instances the user has to delay the external write until he is guaranteed by the system that the data he read is consistent (in general, this is true when the transaction is finally *committed* by the system; in special cases a concurrency control protocol may

16-5

guarantee that all data that is being read by the user will be consistent). Indeed, in some applications, a system that rolls back a read-only transaction may be considered *incorrect,* rather than just very slow. However, we will not take this extreme view in our discussions in this paper.

4. It is also generally not acceptable to give the user a *consistent* set of data, but then ask him to roll back and start over again before he asks for further data (the data to be read after the roll back would normally be inconsistent with the one read before the roll back). For example, a locking protocol may result in a deadlock involving such a transaction and hence it may have to be aborted. The reasons for this requirement are similar to above.

5. Most read-only transactions expect a fast response time since they involve an interactive human user. There are several implications of this:

    (a) The concurrency control algorithm should not unduly delay such transactions. For example, in the two-phase locking protocol [Bernstein87], a read operation on a data item requires getting a lock on the item, and sometimes this may cause a large delay if the lock has been granted to an update transaction.

    (b) Similarly, rolling back read-only transactions should be kept to a minimum since that causes large delays. Examples of rollbacks of read-only transactions have been mentioned above. Now suppose a user has read some data and has spent several minutes looking it over, and then that transaction gets rolled back. Then these wasted minutes have to be included in determining the response time. Similarly, when a user has to delay an external write because the data is not yet guaranteed to be consistent, this delay also becomes part of the response time. In general, we define the *response time* to be the time interval from the moment the transaction issues a read operation first time (over many lifetimes due to rollbacks) and the moment it knows that the data it has received, or has started receiving, is guaranteed to be consistent. Thus rolling back a transaction increases its response time.

6. Many users will generally accept getting a somewhat older version of the database. For example, if a user is   trying to read test equipment data regarding an aircraft, he will generally be comfortable reading data that is just a few weeks old and is not necessarily the most up to date version of the data. This is more true for non-critical data such as finding the phone number of an individual. The database design should try to take advantage of this in its attempts to minimize response time or to increase system throughput.

7. Some *update* transactions also involve interactive, human users and therefore they also expect fast response times. Comments above regarding read-only transactions involving humans also apply here.

8. Many transactions (both read-only and update transactions) are of *long duration* since they involve an interacting, human user. The system should take special care to make sure that they do not affect system performance significantly. For example, in a locking based protocol, if such a transaction is holding locks, it may delay other transactions significantly.

9. Many update transactions have ``useless reads'' in them. For example, a user writing a report might wish to see a report written by a colleague to get a feel for the general contents or structure--- such a read does not affect the actual contents of his own report. Similarly, the user might simply want to surf the database before starting his writing work. Such reads are ``useless'' reads; they violate a common assumption made in traditional concurrency control theory that every write value is an unknown function of every data previously read by the transaction. With internet databases, this assumption is no longer valid. The database system should try to take advantage of this additional information to improve performance. In our algorithm, we allow a transaction to effectively tell the system which of its reads were actually useful reads, and thus the useless reads have no effect on the workings of the concurrency control algorithm.

2. Our Concurrency Control Algorithm

Conceptually, the database goes through a sequence of *versions*, as a result of updates by transitions. The initial version number of the database is 1. With time, the version number changes through the sequence of numbers 1,2,3,.... Each version is guaranteed to be one consistent version of the entire database. How does the database go from a given version I to the next version I+1? Normally this happens when, effectively, some transition T reads version I, updates the database to version I+1, and the system commits to this update. At other times a transaction reads contents of version I, then gets an authorization to update the database, but aborts before completing the update--- in this case contents of version I+1 remain the same as version I. As is commonly assumed in concurrency control algorithms, when a transaction reads from a consistent database and subsequently completes all its updates in isolation from other transactions, the resulting database is also consistent. Thus we see that each version represents a consistent database.

Note that our use of *version numbers* is quite different from that in other multiversion protocols--- in our case, version numbers are assigned to the database as a whole, and not to individual data items. Sometimes we might use a phrase like ``version VN of data item X''--- this only means ``the data item X in version VN of the database''.

We are flexible as to how data is partitioned across the network. A given site might store only some of the data items. Also, a given data item may be replicated at many sites. For the purposes of our concurrency control algorithm, these variations are not important. As is the common convention, these data items can be a relation, or part of a relation, etc.

Suppose site S stores a data item X. Then along with a value of X, it will also store the corresponding version number.

There are two main bodies of information maintained in the system (at the scheduler or the sites):

(a) information as to which site has, in its storage, which versions of which data items, and

(b) information about what versions of the database exist anywhere in the system (without regard to their location), and which data items changed in which version of the database.

These two bodies of information are discussed in Sections 2.1 and 2.2. Note that this is only a conceptual view of what information is being maintained in the system regarding data stored at the sites, and changes in different versions of the database. Other data structures to represent this information are clearly possible; and this issue is not important for the concurrency control algorithm.

2.1. Information about Location of Stored Data

Each site and the scheduler will maintain information that tells what versions of what data items have been actually stored at this site. This information can be updated from time to time via communications among the sites and the scheduler; how that is done is not important to our algorithm. This information is useful when a transaction has decided that it wants to read a data item X in a version number VN, and needs to find out which site can provide this data. This information is also useful when a data item is to be updated.

2.2. Maintenance of Information on Version Changes in the Global Database

The scheduler maintains global information about the changes to various data items as the versions of the database change (but not the actual values of the data items). Specifically, For each data item X, it maintains a linked list of database version numbers which have been committed to by the system, and which resulted from an update to this item (along with updates to possibly some other items as well). In other words, these are the database versions where X has a new, updated value (occasionally, an updated value may turn out to be the same as the previous value, but that is not important. So to simplify the discussion, in our explanations we will assume that each

16-8

updated value is distinct from all previous values. However, note that this is not assumed by the algorithm itself). This list, called the VLIST (Version LIST) is in sorted order by version numbers; the highest version number being at the head of the list. Thus if two consecutive nodes on this list are VN1 and VN2, with VN1>VN2, then that means that the value of X has changed in versions VN1 and VN2; but note that in future, we might insert another node VN3 in this list where VN3 is in-between VN1 and VN2. (The reason for this is that the database version VN3 has been authorized by the scheduler, and X is planned to be updated in that version, but the recovery manager has not finally committed to that update yet; in fact, this update might get aborted due to failures, and therefore may not be inserted in the VLIST. This will become clearer shortly.) If the highest version number on the list for X is VN, then that means that the value of X has changed in version VN, and similar to VN3 above, a higher version than VN might have been authorized but has not been committed by the system yet.

The scheduler has a variable, named CV (Current Version), which is the highest version number such that no other version number VN<CV+1 will ever be inserted in the VLIST of any data item in the future. Therefore all the version change information up to version CV has already been included in VLISTs of all data items and no such node will ever be added in the future. This means that no new versions below CV+1 are currently planned, or will be planned in future, or will be committed in the future. Thus if two consecutive nodes on the VLIST for data item X are VN1 and VN2, with VN1>VN2 and CV+1>VN1, then we are guaranteed that the value of X is same in versions VN2, VN2 +1, ..., VN1 -1 since no new node will be added in this range. Similarly, if the highest version number on the VLIST for X is VN and CV>VN, then that means that the value of X is same in versions VN, VN+1, ..., CV. So if a transaction has read X in version VN, it may assume that version CV will have the same value.

Similar to the above VLISTs and CV *at the scheduler*, each *site* also maintains its own copy of the VLISTS and CV; but they might not be as up to date as the scheduler. This is just a snapshot of the VLISTs and CV of the scheduler, but perhaps a bit older. The purpose of having this information at a site is to provide quick information, without communication with the scheduler or the other sites, to a read-only transaction which is willing to accept a somewhat older version of the database. If it is important to get the latest information, e.g., in case of an update transaction, it will have to communicate with the scheduler. We make the following observations about these variables at a site:

(a) The scheduler maintains VLISTs for all data items in the global database whereas a site may only be interested in some of the items--- typically the data items it stores and possibly a few others that transactions originating at this site are typically interested in. Therefore the site might not have VLISTs of all data items.

(b) The exact values of these variables will not be an exact copy of the values at the scheduler because of delays involved in information exchange.

16-9

(c) CV and VLISTs at a site have information regarding version changes in the global database; and not information regarding which versions of which data items have been actually stored at the particular site (this is a different kind of information, and was discussed earlier in Section 2.1). For example, at site S it is possible to have VLISTs up to version 10, and CV=10, but the actual data values stored at this site are only up to version 5--- the later versions of the data have not yet arrived at this site.

(d) The values of CV and VLISTs, at a given site, must be mutually consistent with respect to their meanings discussed earlier for the scheduler. For example, we should not have a case where, at the scheduler, VLISTs have some nodes with version number 10, and CV=10; and at site S we have CV=10 but the VLISTs do not have any nodes with version numbers 10 because that information has not yet arrived from the scheduler. The information at site S is incorrect because this says that no data items were updated in version 10 (recall that no new nodes will be inserted with version number less than or equal to CV).

(e) The value of CV at any one site is a local value. Different sites might have a different value of CV.

2.3. Behavior of a Transaction and the Transaction Manager

Some of the actions described below are taken by the transaction itself, and some are carried out by the transaction manager on behalf of the transaction. We will not make the distinction between the two.

Before a transaction issues its first read operation, it needs to decide from which version of the database it is going to read various data items. There are several possibilities here. In general, a transaction can find out the CV value at the scheduler or at a particular site by sending a GETVN query to it. However, note that only the scheduler has the latest value of CV. If a transaction needs the most recent data, then it must send the GETVN query to the scheduler. An update transaction would typically need the most recent data. A read-only transaction might prefer an older version, e.g., if that version is actually stored at that site or it had read that version previously.

When a transaction issues a read operation, it specifies which version of the database it wants to read those items from. As long as all the reads have the same version number, the system guarantees that all the values read will correspond to one consistent version of the database. So normally a transaction would use the same version number in all its read operations.

When a transaction issues a write operation, the updates are made in the workspace of the transaction, not the database itself. This is similar to the *validation protocol* discussed in [Korth91].

When a read-only transaction, i.e., one which does not wish to update the database, executes its *commit* operation, it simply terminates, without having to communicate with the scheduler. Unlike the *validation protocol,* it does not need to check if its reads were consistent. If it used the same version number, then automatically they were consistent.

When an update transaction executes its *commit* statement, a validation check needs to be performed at the scheduler to check if the updates by the transaction are to be allowed. To this end, the transaction sends an UP (Update Permission) query to the scheduler with the following parameters:

X1, X2, ... : list of data items that it read and that were used in computing its update values (thus ``useless'' reads need not be mentioned in this list). This is also called the *read set* of the transaction.

VN= the version from which it read the above data items (all of them must have been read in the same version)

Y1, Y2, ... : list of data items that it wants to write to. This is also called the *write set* of the transaction.

The scheduler tests whether this update can be allowed, i.e., it will maintain database consistency; we will discuss this test shortly. If the test fails, the scheduler sends a message DENIED to the transaction, indicating that this update will not be allowed. At this point the transaction will abort itself, and then restart. On the other hand, if the test passes, the scheduler finds a new version number (say VN2) for the database which would potentially result from this update; we will discuss shortly how this number is generated. Then, the scheduler will send a message OK(VN2) to the transaction. On receiving this message, the transaction will update the database. Note that there is no guarantee that the updates will actually be committed on the stable storage--- due to hardware or software failures. In case a failure occurs, the transaction will abort and restart. In case the update gets committed by the system, the transaction will gracefully terminate.

2.4. Behavior of the Scheduler

The UP queries arriving at the scheduler are kept in a FIFO queue and are processed in serial order. To process an UP query, Notice that at any moment, the scheduler may have given update permission to several transactions. It needs to maintain information about them in order to facilitate future updates to its CV and VLISTs. Also, this information is needed to validate future update requests. So the scheduler maintains a list (say, PU--- Pending Updates) of records, each record corresponding to one update permission. Contents of a record are:

      Version number assigned to this update,

      Transaction id,

      The write set

In addition, a variable PV (Pending Version) holds the highest version number assigned to any transaction, whether it has already committed , or aborted, or is still pending.

Below is the test to certify a validation request UP(R, VN, W) where R= the read set and W= the write set:
(a) For each item X in R: the VLIST of X has no node with version number greater than VN. In other words, none of the data items in R has been updated (i.e., with commitment by the system) with a version greater than VN, and

(b) For each record in the PU list: the write set of the record has no common element with R.

Essentially, this ensures that the items read by the transaction have not been updated in more recent versions, and will not be updated by any transaction in the PU list.

If the above test fails, the scheduler sends a DENIED message to the transaction.

If the above test passes, then PV is updated to PV+1, and this new PV value becomes the new version number assigned to this update. A new record corresponding to this update is inserted in PU, and the transaction is sent an OK(PV) message.

Now let us consider what the scheduler does when an update in the PU list is finally committed or aborted by the recovery manager.

In Case of an Abort by the system:

```
Let VN be the version number in the corresponding record in PU list;
delete the above record from the PU list;
if VN=CV+1 then begin
                    CV := CV+1;
                    continue := true;
                    while (CV < PV and continue) do
                    begin
                            if there is a record in the PU list containing version number
                                    CV+1 then continue := false
                            else CV := CV +1
                    end;
        end;
```

16-12

In Case of a Commit by the system:

    Let VN be the version number in the corresponding record in PU list;

    Let W be the write set in the corresponding record in PU list;

    for every data item X in W do

           in the VLIST for data item X, insert a node with value VN;

    delete the above record from the PU list;

    if VN=CV+1 then begin

        CV := CV+1;

        continue := true;

        while (CV < PV and continue) do

        begin

           if there is a record in the PU list containing version number

              CV+1 then continue := false

           else CV := CV +1

        end;

    end;

When the system commits to an update, if the version number VN of this update is not CV+1, then note that CV is not advanced to VN. Doing so would be semantically incorrect in terms of the meaning of the CV, since the status of some earlier versions is still unknown; and it would cause inconsistency retrieval problem, since later an earlier version may get committed. For example, suppose before a particular commitment, CV=5 and the update is for version 10, and the write set has X and Y in it. Suppose some other data item Z, not in this write set, has the highest version number equal to 7 in its VLIST and there is a record in PU with Z in its write set and version number being equal to 8. After the commitment of version 10, if we move CV to 10 then semantically we are declaring that the value of Z does not change in versions 8, 9, and 10. This is semantically incorrect since we don't know if version 8 will get committed or not. Moreover, if we changed CV to 10, then later some transaction T might find that CV is 10, and it may issue a read of X and Z in version 10 (the system will supply the value of Z from version 7, since CV = 10 and therefore Z is supposed to be the same in version 10). In case version 8 gets committed, these values of X and Z do not represent values from a consistent database.

.

2.5. Version Notification Feature

Consider the following situation. An update transaction T has read items X and Y in version 5, and is continuing with more read operations or computing values for write operations. It is quite possible that there is a newer version in the database and X has changed in this version. Therefore the present work of T will go waste since it will not get validated later. It would be useful to inform T of this version change so that it can restart or selectively re-read the items that have changed. Even if X has changed in a pending update (i.e., the update is recorded on the PU list but is not committed or aborted by the system yet), it would be useful for T to know about it.

Therefore, in order to improve performance of the algorithm in such situations, we include the following feature in the algorithm. A transaction T may send a request NOTIFY(VN, RLIST) to the scheduler. VN is a version number and RLIST is a list of data items. If the VLISTs of these items or the PU list indicates that any of these items have changed, or may change if one of the pending updates gets committed, in any version beyond version VN, then the scheduler sends a VCHANGE(VN1) message to T. VN1 here is the highest version number known to the scheduler where one of the items would change. In case there is no such known change yet, the scheduler will keep the NOTIFY message in a list and will process it again every time a new record is inserted in the PU list.

When the transaction T receives the VCHANGE message, T may further interrogate the scheduler to find out exactly what has changed. We skip details of such an interrogation process here. Based on this, the transaction may roll over, or selectively re-read the affected items.

Typically a read only transaction will not be using this feature; this is mainly useful to an update transaction. The NOTIFY message may be sent right after finding the current version number through the GETVN query, or it may be sent after the transaction has made some progress in its computation so that it has a better idea of what the set of useful reads may be. Of course, if it does not know the exact set of useful read, it may simply send a larger set as the parameter RLIST in the NOTIFY message. Indeed, it may send this message several times during its computation after getting replies to the previous ones.

2.6. Correctness of IDP

It should be clear that the protocol is serializable. Irrespective of whether the pending updates in the PU list get committed or aborted, each version of the database is consistent, and is the result of a serial execution of zero or more transactions.

2.7. Some General Observations on IDP

1. A read-only transaction is never rolled back by the scheduler. (It may roll back due to hardware or software failure, but that is unavoidable.)

2. A read-only transaction is never delayed while waiting for an event to take place at other transactions. For example, it does not have to wait for certain locks to become available, etc. The only delays are communication delays and computation delays when a processor (executing this transaction or the scheduler) is executing some other program.

3. To get above benefits, a transaction does not even have to know at its beginning that it is going to be a read-only transaction. It does not have to declare to the scheduler that it is a read-only transaction. For example, it might read some data, and based on data values it may decide that it does not want to do any updates.

3. Simple Variations of the Algorithm

Several variations of the algorithm are possible which may improve user convenience, performance improvements etc. Here we suggest some of the possibilities.

1. Earlier we said that in the UP(read set, VN, write set), all the data items in the read set must have been read in the same version, namely VN. However, this need not be strictly true. For example, it is possible that the transaction T reads X from version 5, then finds that the version number has changed to 7 but X has remained the same, then it reads Y from version 7, and later it submits an UP query with X and Y in its read set and VN being equal to 7. This should be acceptable since the value of X is same in versions 5 and 7.

2. Note that we allow a transaction to read data from different versions. This may be useful to some read-only transactions. For example, such a transaction might want to read the latest available version of certain data items, even if some other items that it read are from a much earlier version. Of course, this does not guarantee that all the data read by the transaction will be mutually consistent. However, the protocol will allow this without causing any rollbacks.

4. Comparison of Our Algorithm with a Multiversion Algorithm

Now we compare our algorithm, i.e., the Internet Database Protocol (IDP) , with the multiversion protocol (say MVP) discussed in [Korth91]. MVP is also based on multiple versions, but the main difference is that the version numbers are assigned to individual data items, not the database as a whole. So there is no simple way to take one consistent snapshot of the database. In our protocol, each version number represents one consistent version of the database. We first briefly review MVP.

4.1. Brief Review of MVP

In MVP, each transaction T is assigned a unique timestamp, denoted by TS(T). Intuitively, the protocol attempts to schedule the operations of the transactions in such a way that the net effect is as if they executed serially, in their timestamp order. One typical way to generate the timestamp is to use the value of the system clock when the transaction begins its execution. In a distributed system additional care is taken to make sure that the transactions running at different machines will still be guaranteed to have different timestamps.

Each data item in the database has several versions. As in IDP, different values of a data item are stored along with their version numbers. In MVP, the version number of any version of a data item X is defined to be the timestamp of the transaction that created that version of that data item, by executing a write operation. This is also called the *write-timestamp* (WTS) of that version of X. With each version of X, we also keep another timestamp called the *read-timestamp* (RTS) of that version of X. This is the highest timestamp of any transaction that read or wrote this version of X (of course, only one transaction could have written this version).

We use the term ``latest'' version of X to mean the version of X with the highest value of WTS.

In MVP, when a transaction wishes to read or write an item X, it issues a read(X) or write(X) operation. Unlike IDP, the operation does not specify which version is to be read or written. Also, unlike IDP, the operation goes to the scheduler which determines which version of the item is to be read or written, and whether the operation is permitted. In case of a write, the new value and its timestamps are written immediately to the database; unlike the case with IDP where the values are written in the local workspace of the transaction and written to the database only at the end of its code.

Behavior of the scheduler when it receives a read(X) request from transaction T: ·
      Let X1 be the version of X whose WTS is the largest among all versions of X that have WTS<TS(T);
      if RTS(X1)<TS(T) then RTS(X1) := TS(T);
      Permit T to read the version X1;

Behavior of the scheduler when it receives a write(X) request from transaction T:
      Let X1 be the version of X whose WTS is the largest among all versions of X that have WTS<TS(T);
      if TS(T) < RTS(X1) then roll back T and any other transactions that read a value written by T or
            by any other transaction being aborted hereby
      else create a new version of X (say X1) with WTS(X1)=RTS(X)=TS(T)

4.2. Performance Considerations

We consider several execution scenarios and compare the relative performance of the two algorithms for those scenarios.

Example 1:

Using MVP, suppose the latest version of X is X1, WTS(X1)=RTS(X1)=100. Suppose TS(T1)=200 and TS(T2)=300. T2 is a read-only transaction. T1 wishes to read X, write X, and then commit. The following might happen:

T2 executes reads(X) and therefore reads X1. Now RTS(X1)=300.
T1 executes read(X) and therefore reads X1. Now RTS(X1)=300.
T1 executes write(X). Since TS(T1) < RTS(X1), T1 rollbacks.

If the same execution took place in IDP, there will be no rollback. This rollback is clearly unnecessary. Even if T2 were an update transaction, IDP will let T1 complete and commit while T2 is still computing. This speeds up the system throughput.

To put it more intuitive terms, MVP assigns a timestamp to the transactions according to when they start, and, informally speaking, hopes that they will execute their operations in the same order. It does not take into account the fact that a transaction that started later may be a short transaction that finishes up earlier. IDP commits to those transactions as they finish, irrespective of when they started.

Example 2:

Consider two transactions T1 and T2 with TS(T1)<TS(T2). T2 simply wishes to read X,Y,Z. T1 wishes to write Y and write X. The following may happen in MVP:

T2 executes read(X).
T1 executes write(Y).
T2 executes read(Y).
T1 executes write(X)--- this results in rollback. That causes T2 also to rollback since T2 has read data written by T1.

16-17

Here a read-only transaction is being rolled back. In IDP, no rollback will take place--- both transactions will finish gracefully. This is because in IDP, T2 will read from a consistent version of the database that existed before these two transactions started.

4.3. Flexibility of the Algorithm

IDP and its simple variations provide a high degree of flexibility to the transactions. For example, a transaction need not decide at its beginning whether it is a read-only transaction or not; it may decide that after reading some data. The protocol allows a transaction to re-read a data item in several different versions without making the algorithm itself more complex or having to rollback other transactions, etc. In MVP, when a transaction issues a read operation on a data item X, it has no control as to which version will be read; the concurrency control algorithm decides that for the transaction. In MVP, a transaction T1 cannot read data values written by transaction T2 if $TS(T1)<TS(T2)$. This is undesirable in a situation where T2 has made substantial progress in its computation and is well ahead of T1, even though it started after T1. The notification feature provides ability of a transaction to keep up with any version during its own computation. Several variations mentioned in Section 3 further illustrate the flexibility of the algorithm. The algorithm itself can be modified in many ways, and new features can be added.

4.4. Simplicity of the Algorithm

IDP does not require generation of timestamps for the transactions. Only one timestamp is needed for individual data items, not two as in MVP (RTS and WTS). Conceptually also, the rules that the scheduler follows in dealing with write operations are much simpler.

5. Concluding Remarks

We have presented a concurrency control protocol that is particularly suited for internet databases. Read-only transactions have been given special treatment and they are never aborted by the algorithm. The algorithm allows transactions to take long time durations in their computations without holding up the rest of the system. We believe the number of rollbacks would be far less in this algorithm than in the multiversion protocol, as illustrated by several examples in Section 4.2. The algorithm is fairly simple and flexible.

Ravindran Krishnamurthy of HP Labs, Palo Alto, California, for a long technical discussion on various parts of this research.

References

[Bassiouni88] M. Bassiouni, ``Single-site and Distributed Optimistic Protocols for Concurrency Control'' IEEE Transactions on Software Engineering, vol. SE-14, no. 8, pp. 1071-1080, August 1988.

[Bayer80] R. Bayer, M. Heller, and A. Reiser, ``Parallelism and Recovery in Database Systems'', ACM TODS, vol. 5, no. 2, June 1980.

[Bell92] D. Bell and J. Grimson, Distributed Database Systems, Addison-Wesley, 1992.

[Bernstein83] P. A. Bernstein, N. Goodman, and M. Y. Lai, ``Analyzing Concurrency Control when User and System Operations Differ'', IEEE Transactions on Software Engineering, vol. SE-9, no. 3, pp. 233-239, May 1983.

[Bernstein87] P. A. Bernstein, V. Hadzilacos, and N. Goodman, Concurrency Control and Recovery in Database Systems, Addison-Wesley, 1987.

[Breitbart92] Y. Breitbart, H. Garcia-Molina, and A. Silberschatz, ``Overvview of Multidatabase Transaction Management'', The VLDB Journal, vol. 1, no. 2, October 1992.

[Buckley83] G. Buckley and A. Silberschatz, ``Obtaining Progressive Protocols for a Simple Multiversion Database Model'', Proceedings of the International Conference on Very Large Data Bases, pp. 74-81, 1983.

[Buckley85] G. Buckley and A. Silberschatz, ``Beyond Two-Phase Locking'', Journal of the ACM, vol. 32, no. 2, pp. 314-326, April 1985.

[Ceri84] S. Ceri and G. Pelagatti, Distributed Databases: Principles and Systems, McGraw-Hill, 1984.

[Date93] C. J. Date, and H. Darwen, A Guide to the SQL Standard, Addison-Wesley, 1993.

[Date95] C. J. Date, An Introduction to Database Systems, Addison-Wesley, 1995.

[Franaszek85] P. Franaszek and John T. Robinson, ``Limitations on Concurrency in Transaction Processing'', ACM TODS, vol. 10, no. 1, March 1985.

[Garcia-Molina87] H. Garcia-Molina and K. Salem, ``Sagas'', ACM SIGMOD International Conference on Management of Data, San Francisco, CA, May 1987.

[Gray93] J. Gray and A. Reuter, Transaction Processing: Concepts and Techniques, Morgan Kaufmann, 1993.

[Korth88] H. F. Korth and G. Speegle, ``Formal Model of Correctness Without Serializability'', Proceedings of the ACM SIGMOD International Conference on Management of Data, pp. 379-386, 1988.

[Korth90] H. F. Korth and G. Speegle, ``Long Duration Transactions in Software Design Projects'', Proceedings of the International Conference on Data Engineering'', pp. 568-575, 1990.

[Korth91] H. F. Korth and A. Silberschatz, Database System Concepts, McGraw-Hill, 1991.

[Mohan92] C. Mohan, D. Haderle, B. Lindsay, H. Pirahesh, and P. Schwartz, ``ARIES: A Transaction Recovery Method Supporting Fine-Granularity Locking and Partial Rollbacks Using Write-Ahead Logging'', ACM TODS, vol. 17, no. 1, March 1992.

[O'Neil86] P. E. O'Neil, ``The Escrow Transactional Method'', ACM TODS, vol. 11, no. 4, December 1986.

[Oszu91] M. T. Oszu and P. Valduriez, Principles of Distributed Database Systems, Prentice-Hall, 1991.

[Papadimitriou86] C. Papadimitriou, The Theory of Database Concurrency Control, Computer Science Press, 1986.

[Reed83] D. Reed, ``Implementing Atomic Actions on Decentralized Data'', ACM Transactions on Computer Systems, vol. 1, no. 1, pp. 3-23, Feb. 1983.

[Sheth90] A. P. Sheth and J. A. Larson, ``Federated Database Systems for Managing Distributed, Heterogeneous, and Autonomous Database Systems, ACM Computing Surveys, vol. 22, no. 3, pp. 183-236, 1990.

[Simon96] E. Simon, Distributed Information Systems, McGraw Hill, 1996.

A PROPOSED EXPERT SYSTEM FOR
ATS CAPABILITY ANALYSIS

Ernest L. McDuffie
Assistant Professor
Department of Computer Science

Florida State University
Tallahassee, Florida 32306-4019

A PROPOSED EXPERT SYSTEM FOR
ATS CAPABILITY ANALYSIS

Ernest L. McDuffie
Assistant Professor
Department of Computer Science
Florida State University

Abstract

Automatic Test Systems (ATS) are used by the Air Force and throughout the Department of Defense (DoD) to assist in the integrated diagnosis of weapon systems. The large amount of complex information available on each of the 617 ATS's in the Air Force inventory makes it necessary to have some automated means of accessing and evaluating the relevant information in order to accurately assess the overall Air Force ATS capability. The proposed system named ATS Comparative Evaluation Expert System (ACEES) will give the Air Force a new automated tool to meet the ever growing demand for efficient information management. This final report will outline the motivation and high-level design of ACEES as well as plans for the establishment of an ongoing research program designed to support and extend ACEES initial capabilities.

A PROPOSED EXPERT SYSTEM FOR
ATS CAPABILITY ANALYSIS

Ernest L. McDuffie

Introduction

As a relatively new assistant professor it is imperative to my career advancement that I develop a strong research program. Such a program is characterized by three primary components. First, to build the program there must be some theoretical foundation. In some disciplines theoretical work by itself can be enough to sustain a career. Typically, this is not the case in computer science. My theoretical background has been in the area of Artificial Intelligence (AI), with specific work dealing with expert systems, automatic scheduling, and temporal reasoning. The second component of a research program is formed by the selection and exploration of an appropriate real-world problem domain. This was my main motivation for being in the Air Force Office of Scientific Research Summer Faculty Research Program. Finally, the hope is that by combining the first two components I will be able to produce results that are of sufficient value to attract funding for continued long term research into the problem domain. Securing such external research funding is the third component of a strong research program. To this end a Summer Research Extension Program (SREP) proposal will be submitted. If granted, this proposal will allow for the continuation of research efforts, in the area described below, through the end of 1998 and hopefully beyond.

What is the nature of the problem domain in this case? The larger domain is one of weapon system maintenance. All operational systems will break down and fail at different times during their life cycle. By using ATS's to assist in the maintenance process systems can be quickly diagnosed, repaired, and returned to operational status. The general diagnosis problem for complex systems is a very difficult one. A sub-problem in this domain, optimizing a diagnostic sequence of test, is known to be NP-complete [1]. The problem of diagnostic test sequencing will be revisited in the discussion of future plans for the ACEES project. For now we will focus on the ATS's themselves. There are two major components of an ATS: Automatic Test Equipment (ATE), its hardware and operating software, and the Test Program Set (TPS), which is composed of hardware, software, and interface documentation. On-system automatic diagnostics and testing are also considered to be ATS [2].

There has been such a proliferation of these systems over the past few years that DoD has established an ATS Master Plan. This plan, among other things, provides a consolidated strategy the implementation of an investment strategy and acquisition policy for ATS's DoD wide. Key elements of the plan include; policy background and management structure; designation of ATS families; research and development efforts; and a modernization strategy. The actual plan itself is currently being rewritten to comply with new DoD regulations [3].

An ATE includes test hardware and its accompanying software. There is a wide range of hardware configurations possible for an ATE. From small man-portable suitcase size equipment to six or more six-foot high racks of over 2,000 pounds of equipment. At the heart of all this is a computer that controls various test instruments such as waveform analyzers, digital voltmeters, signal generators, and switching assemblies. In addition, a number of housekeeping functions are preformed by the ATE's own operating system. The retrieval of digital technical manuals, test procedure sequencing and storage, self-test, self-calibration, and the tracking of preventative maintenance requirements are among these functions [4].

Interface devices and their associated documentation, together with test software are the components of a Test Program Set (TPS). The software is executed by the computer of a given ATE and is written in standard computer languages such as Ada or ATLAS. This software controls the stimulus and measurement instruments in the ATS and the system response at given points in the ATS. A large amount of information relevant to both the ATS and the Unit Under Test (UUT) is available in these TPS's [5]. The Navy's Program Management Air (PMA-260) has complied a database of some 175 software tools that are currently available to TPS developers. A little more than 7% of these tools may be of direct use to the proposed project.

By combining numerous on-line and/or off-line databases, ACEES will have access to all available computerized information relevant to the ATS environment. Whenever one

plans to design an expert system, issues of knowledge representation, interfaces, inference, and knowledge acquisition must be addressed. Fortunately, these issues and more are already being considered by the IEEE Standards Coordinating Committee 20 (SCC-20 formerly known as the ATLAS Committee). For the past seven years SCC-20 has been working on a new test standard called AI-ESTATE. This standard, as proposed in the recently submitted full use PAR 1232, stands for Artificial Intelligence Exchanges and Service Tie to All Test Environments. AI-ESTATE will deal with all Artificial Intelligence (AI) as it relates to test and diagnosis. It will define the interfaces between a test related reasoning system, its users, target test equipment, test information knowledge bases, and conventional databases. Clearly, following this standard during the design phase of this new expert system will ensure that the resulting system will be compatible with the ATS environment.

The remainder of this report will focus on the development of ACEES and an ongoing research program to support it.

Problem Definition

Currently there are some 617 different ATS's Air Force wide. These systems are used to maintain all the weapon systems of the Air Force which consist of literally thousands of UUT's. The need exists to have the capability to compare and evaluate, quickly and accurately, the capabilities of these systems. The ultimate goal of such an analysis would be the identification of redundant systems in terms of their capabilities. This information

could then be used to significantly reduce the total number of ATS's required which would result in reduced deployment footprints. Once all relevant information has been collected and put in the proper format (this process is currently on going) it will be possible for an appropriately designed expert system to accomplish this task of ATS evaluation. In addition, it is conceivable that such an expert system could easily manifest other important capabilities. An example of such a capability would be new weapons system ATS support evaluations. Given a set of UUT's, representing a new weapon system, ACEES could determine an optimal set of ATS's required to support it. Other as yet unforeseen capabilities could be easily added to ACEES due to it object-oriented design.

The proposed expert system would have access to a number of extensive databases containing information about ATS's, Test Program Sets, Test Requirement Documents, UUT's, and other computerized information as it becomes available. Because of all the information at its disposal it may become possible for the system to assist in other areas of the automatic test environment. For example, the ability to manipulate and reason about this information may prove to be valuable to integrated diagnosis, the automatic development of test program sets, the determination of optimal test sequences for complex diagnostic problems, and other as yet unanticipated functions.

In order to ensure that ACEES will be compatible with current and future databases and information bases, it is planned that this project will adhere to all AI-ESTATE standards.

During a recent meeting with members of the IEEE SCC-20, which is responsible for AI-ESTATE development, it was determined that even though the main focus of this standard is diagnosis, the planned use of this standard for the ACEES project is appropriate and will prove to be beneficial for future development.

Methodology

The design and implementation of a large expert system must be approached systematically. To accomplish this some well defined life cycle model should be used. The Linear Model has been successfully used to construct a number of expert systems [6]. Listed below are the six main phases in this model:

1. Planning

2. Knowledge Definition

3. Knowledge Design

4. Code & Checkout

5. Knowledge Verification

6. System Evaluation

Contained in each of these phases are a number of tasks and subtasks that need to be completed. This document will describe two important components of the planning phase for this project. Both, the preliminary functional layout of ACEES, and its high-level requirements will be outlined and discussed in some detail.

In an attempt to partially assess the feasibility of ACEES a small sub-problem was defined as follows: Find Optimal Set of Automatic Test Systems (ATS's) for a Given Weapons System (WS). To solve this problem a simple algorithm was developed that could be executed (on one WS manually) without the need of a full blown expert system or any additional programming. This algorithm was named the ATS Consolidation Algorithm (ACA).

Optimal is defined as a set of ATS's having minimum cardinality and still being able to support all the UUT's and Tester Replaceable Unit's (TRU's) of a given WS. This definition may require modification to include the cost factors for specific ATS's. Cost information relevant to specific ATS's will be available to ACEES. Specially when deployment is considered, it will become necessary to look at cost factors related to this activity and how they affect the definition of optimal. The algorithm described below is designed to identify an optimal set of ATS's for a given WS and uses databases from the ATS Management Analysis Program (ATSMAP) system as input. Information generated by this algorithm could be used to consolidate the number of ATS's currently required to support a given WS. By applying it to all WS's using iteration and then across WS's, the entire Air Force inventory of ATS's could be affected. In its current form ACA only looks at the high level association of UUT's to WS's to ATS's. It may be necessary to examine the current set of UUT's serviced by specific ATS's. In some cases an ATS may

be capable of servicing more UUT's then currently identified. An increase in the cardinality of such a UUT set could possibly led to further ATS consolidation.

ACA Logical Description:

Step # Actions to be Performed

1. Using UUT.dbf (database file used by ATSMAP) construct the (U_{ws}) set. This set will contain all UUT's associated with a given WS.

2. Using PMP.dbf (another database used by ATSMAP) construct the (ATS_{ws}) set. This set will contain all ATS's associated with a given WS.

3. Using (ATS_{ws}) construct the $[OATS_{ws}]$ set. This is an ordered set with its first member being the ATS which can test the largest number of UUT's associated with the given WS. Each following member will have an equal or lesser number.

4. Using $[OATS_{ws}]$ remove the first member from the set and place it into the $(RATS_{ws})$ set. At termination this set will contain only the Required ATS's for the given WS.

5. Using the first member of $[OATS_{ws}]$, identified in step 4, place all its associated UUT's and TRU's into the (U_c) set. This set will be compared to (U_{ws}) as part of the algorithm's termination condition.

6. IF $\{(U_{ws})$ not equal $(U_c)\}$ THEN return to step 4 ELSE IF $\{(RATS_{ws})$ not equal $(ATS_{ws})\}$ THEN go to step 7 ELSE go to step 8.

7. Print the members of (RATS$_{ws}$) under the heading "Only Required ATS's for [given WS]" and terminate algorithm.

8. Print the following message, "No Consolidation Indicated for the [given WS] Weapon System." and terminate algorithm.

Results from the manual execution of this algorithm will be reported in the next section. A more flexible and robust version of ACA will become part of the function capabilities of the finished ACEES.

In order to facilitate the meeting of AI-ESTATE IEEE standards during the development phase of ACEES, an object-flavored information model specification language named EXPRESS will be used [7]. This language will be used during the information modeling phase, which is part of the knowledge definition phase of the linear life cycle model. Information modeling allows you to deal with things, properties they may possess, their behavior, and how the interact with other things. A main idea of the EXPRESS language is that there should be a definite separation between an information model and the information system environment. This makes it much easier to share information from different sources. This ability to share information will allow for the implementation of AI heuristic techniques in the management and manipulation of that information within ACEES.

Results

After reviewing ATSMAP's functional capabilities, it was determined that they were insufficient to completely implement this algorithm. Since the database files used by ATSMAP are FoxPro files, it should be possible to implement ACA in FoxPro or any compatible database environment. In this instance the files were converted to MS Access format and manipulated from that environment. Using the C-141 weapons system as a test case and ATSMAP's functions, 31 ATS's were identified with 276 UUT's and 290 TRU's associated to them. Using ATSMAP's functions it was possible to identify all ATS's associated with the chosen WS and then all UUT's and TRU's associated with those ATS's. However, it is impossible to do the reverse and identify all the UUT's and TRU's associated with a chosen WS. Also, once sets of items are identify it is not possible to do operations on these sets from with in ATSMAP.

At this point it became necessary to convert all the ATSMAP files into MS Access format so that normal database operations could be applied. Once this conversion was accomplished it was possible to complete the manual execution of the ACA for the C-141 weapon system. A few important facts were learned during this process. The presence of inconsistencies in the database was confirmed. The number of ATS's associated with the WS was different depending on the view. This should not occur and will be corrected in future databases. It was also shown that, at least for this single WS, it will be necessary to expand the number of UUT's and TRU's associated with a given ATS if possible. This

can be accomplished by using other more detailed parametric information for other databases. This expansion process will be within the operational capabilities of ACEES.

In recent years a number of other AI techniques and processes have been applied to the ATS environment with varying degrees of success. A report [8] prepared for the Air Force summarized a project named Neural Engineering Utility With Adaptive Algorithms (NEUWAA) along with some twenty-one other recent projects that employed various AI techniques. Eleven of the projects involved expert systems, eight used neural networks, four were based on fuzzy systems, and one utilized genetic algorithms. All of the projects focused on very narrow sub-domains of the ATS environment and produced limited results. It is not clear if any of these projects remain operational at this time. Of all the projects the ones dealing with expert systems seem to be the most promising and none have attempted to implement the goals of ACEES.

In the following two sections a first pass is made at outlining both the high-level design and requirements for ACEES. This report will end with conclusions and future plans for an ongoing research program to support and extend the proposed systems capabilities.

Preliminary High-Level Design

In this section a high-level description of the functionality of ACEES will be presented. The main function of ACEES will be the comparative evaluation of ATS's. It will be

17-13

possible to perform this evaluation in a number of different modes. First, a single ATS can be evaluated. During this evaluation mode, and all subsequent modes, all available information relevant to the ATS will be used. The current plan is to use all the information in the ATSMAP database in addition to parametric information that has already been collected. In the single ATS mode this information is compared to all the other ATS's in the knowledge bases (KB's) of the ACEES system to determine if, and to what extent, the ATS under consideration duplicates capabilities and functionality's of other ATS's represented in ACEES. Two reports will be generated as a result of the execution of this mode on an ATS. One that will summarize the overall findings and another that will give details on every aspect of the evaluation. These details would take the form of a listing of all the relevant points of information used during the evaluation.

There are a number of other functional modes that will be available to ACEES. The weapon systems mode will allow for the evaluation of an ATS in terms of a specific weapons system or family of systems. Because of the similarities between this mode and the single ATS mode it will be possible for ACEES to detect redundancies and inconsistencies that may be present in the KB's of the system. So, by combining certain functionality's of ACEES, a verification mode will be created and provided to the user. A Unit Under Test (UUT) and Tester Replaceable Unit (TRU) mode will take as input a specific ATS, or weapon system, or family of weapon systems, and return all the UUT's and TRU's associated with them. An expanded ATS capability's mode will also be provided. This mode will look at the capabilities of a specific ATS and determine if there

are UUT's or TRU's that are not currently associated with the ATS but are with in the ability of the ATS to test.

Five functional modes for ACEES have been described. By using these modes it will be possible to identify redundancies in the Air Force ATS inventory, evaluate consolidations, additions, as well as deletions to the inventory, therefore reducing deployment footprints. Another powerful feature of ACEES will be the ability to create new functional modes when a need is identified. An option, Create Mode, will be provided. This mode, assuming a new function has been identified, will guide the user through an intuitive step-by-step interactive process that will result in the creation of a new functional mode which ACEES will then be able to save for future use. ACEES capabilities will be able to continually evolve by using this option.

High-Level Requirements

As stated in the previous section, all the modes will have access to all the KB's of the system. There will be a number of KB's in ACEES and the capability to add new KB's as needed and when available. The initial prototype ACEES will be a stand-alone system with all its KB's and expert system components physically located on a single machine. The final operational version will be a distributed system which will combine both local expert system components and knowledge base files with other data, information, and knowledge bases that will be at different locations and accessed via the world wide web.

Databases that will be included as part of ACEES are the ATS Management Analysis Program (ATSMAP) files, some of the earlier files containing parametric data on ATS's, Test Program Sets data generated by the Test Data Analysis System (TDAS), Test Requirement Documents (TRD's), Product Master Plan, Product Line Master Plan, and Integrated Support Equipment Master Plan data. These last five groups of information are in the process of being developed and the nature and format of the final form of the information to be used by ACEES is yet to be determined. The same is true for the other databases in the sense that the final interfaces between the KB's and ACEES are to be determined.

The expert system inference engine component of ACEES, in addition to the standard logical operations capabilities, will be able to incorporate a number of fuzzy logic operations that have been successfully used recently in the medical domain for relating patients, to sign and symptoms, to diseases, another type of diagnosis process. This capability may or may not be of use to the initial functional modes of ACEES but certainly will be of great benefit to the system as it matures and moves more into the diagnosis domain. It is envisioned that a mature ACEES will be able to use it vast knowledge of the ATS environment to assist in the determination of diagnosis test sequencing and the general development of TPS's.

One last important requirement of ACEES is that it have to ability to access the Navy's System Synthesis Models (SSM+). DoD has established an ATS Executive Agent to

oversee ATS programs DoD wide. Both the Navy's Consolidated Automated Support System (CASS) and the Army's Integrated Family of Test Equipment (IFTE) have all their ATS information accessible by SSM+ for analysis. Since DoD has appointed the Assistant Secretary of the Navy for Research, Development, and Acquisition to serve as the DoD ATS Executive Agent the SSM+ system has become a standard DoD wide analysis tool. An ORACLE database, a mapping model, and a workload model are the components of SSM+. These components give the SSM+ some of the same basic capabilities as ACEES. Currently the 617 Air Force ATS's are not represented in the SSM+ database. ACEES will have capabilities not available to SSM+ and access to its database in its final version.

Conclusions And Future Plans

This document represents a first pass at defining the functional capabilities and requirements for the proposed ACEES project. Clearly, much greater detail will be required in this planning phase before progress can be made towards the implementation of an operational prototype. At least four more planning tasks also remain to be completed. These remaining tasks include a feasibility assessment, the creation of a resource management plan, as well as task phasing and scheduling for the entire project. The details of these tasks are beyond the scope of this paper and will be addressed in the SREP proposal. It should be stated that the feasibility of this project, at least at the basic level, is unquestionable because of the existence of SSM+. The potential benefits to the

Air Force are many. Not only will the proposed ACEES project adhere to, support, and enhance all current DoD policies regarding ATS's, but it will also be written to the currently developing IEEE AI-ESTATE standards which will result in an ability for the system to grow into a significant ATS maintenance and diagnostic tool for the 21st century.

For ACEES to realize its full potential, its design, implementation, and continuing development must be accompanied by an ongoing research program. Such a program was missing for all the previously attempted AI type projects. This program would involve a partnership between the Air Force's Advanced Diagnostic and Technology Insertion Center (ADTIC) and the Department of Computer Science at Florida State University. The first steps in the establishment of this partnership will be outline in a soon to be submitted SREP proposal. This type of relationship is extremely cost effective from the Air Force viewpoint. Not only would the Air Force receive the benefits of the best and brightest young minds in the form of graduate students, but the University cost sharing ability would deliver these resources and others common to a research one institution (institutions that have research as a primary mission, of which there are less then twenty in the United States) for much less then would otherwise be possible.

In addition to the development and support of ACEES, this partnership would allow for the establishment of a resource center for the continuing evaluation of new AI techniques as they apply to the ATS environment. The need for such a center has been long

advocated by members of ADTIC [9]. To realize the long term goal of ACEES, to be able to deal with the integrated diagnostic problem directly, such a partnership is essential. The Department of Computer Science at Florida State University is a relatively young and growing department. A recently announced initiative in Trustworthy Systems will establish the department as a national leader in this new and critical area of computer science. Trustworthy Systems is an area of computer science that encompasses correctness, reliability, safety, and security of computer systems and software, including applications. All of the techniques of Trustworthy Systems will be used in the development of ACEES.

The domain of integrated diagnostics (ID) is very complex and far reaching [10]. If the SREP proposal is approved, then the ACEES project will be taking a first step towards mastering ID for not only the Air Force but the Department of Defense in general. It is conceivable that the lessons learned during this project will enhance and extend not only the domain of systems test and diagnosis but also AI. Real world experience in the field of expert systems design and implementation for large complex multi-source information environments will indubitably have unforeseen benefits for both computer science and the Automatic Test Systems environment.

References

[1] M. R. Garey and D. S. Johnson, 'Computers and Intractability: A Guide to the Theory of NP-Completeness,' Bell Laboratories, Murray Hill, New Jersey; W. H. Freeman and Company, New York, (1979).

[2] DoD web page URL: http://dodats.navy.mil/dodats/ats/ats.html.

[3] DoD web page URL: http://dodats.navy.mil/dodats/ats/mstrpln.html.

[4] DoD web page URL: http://dodats.navy.mil/dodats/ats/ate.html.

[5] DoD web page URL: http://dodats.navy.mil/dodats/ats/tps.html.

[6] J. C. Giarratano and G. Riley, 'Expert Systems: Principles and Program,' 2nd edition, PWS-Kent Publishing Company, Boston, (1994).

[7] D. Schenck and P. Wilson, 'Information Modeling the EXPRESS Way,' Oxford University Press, New York, (1994).

[8] Conceptual MindWorks, Inc., San Antonio, Texas, 'NEUWAA Software Support/Evaluation & Artificial Intelligence Utilization in Automatic Test Systems: Final Report,' prepared under contract F41608-D-94-1371, task order 003 for: SA-ALC/LDAE ADTIC, (1995).

[9] J. S. Dean, 'Artificial Intelligence in Test,' AUTOTESTCON '96 Proceedings, IEEE, Dayton, Ohio, September 1996, pp. 309 - 313.

[10] W. R. Simpson and J. W. Sheppard, 'System Test and Diagnosis,' Kluwer Academic Publishers, Boston, (1994).

HOW TO PROVIDE AND EVALUATE COMPUTER NETWORK SECURITY

Prabhaker Mateti.
Associate Professor
Department of Computer Science and Engineering

Wright State University
3640 Colonel Glenn
Dayton, Ohio 45435-0001

HOW TO PROVIDE AND EVALUATE NETWORK SECURITY

Prabhaker Mateti
Associate Professor
Department of Computer Science and Engineering
Wright State University

Abstract

Because of universal connectivity to Internet, computer network security has become urgently important. This is a study of what is desired, and how security can be provided. Big corporations have been installing so-called network firewalls. Smaller organizations are yet to realize the implications. We propose an evaluation methodology that gauges the effectiveness of a security solution package. We also suggest security fortifications for individuals running Windows95 or Linux on their PCs.

HOW TO PROVIDE AND EVALUATE NETWORK SECURITY

Prabhaker Mateti

1 Introduction

Computer networking has become so ubiquitous that a computer system of any importance is networked to others. Often, this networking reaches the public Internet, outside the boundaries of an institution. The Internet Domain Survey (see www.nw.com) reports that there are, as of July 1997, 19,540,000 hosts, and 1,301,000 domains connected to the Internet versus 1,776,000 hosts 26,000 domains in Jul 93. The older security problems of insider breaches of security are now compounded by attacks carried out remotely through the network.

A recent article (Information Week, Sep 1, 1997) claims that "A 1996 survey by The American Society of Industrial Security reported that more than 75 percent of computer security breaches are due to the actions of insiders. FBI statistics show that more than 60 percent of computer crimes originate inside the enterprise. And a recent Yankee Group survey found 44 percent of respondents actually reported security breaches by insiders." But the general impression in the community is that network-based attacks outnumber the attacks based on physical access to the computer systems by a factor of 100 or so. Whether it is justified or not, this paranoia is fueling the growth of what are known as "network firewalls". According to www.TechWire.com, July 10, 1997, "Firewall revenue is expected to reach \$286 million by the end of the year and is expected to grow to \$729 million in 2001. The United States accounted for 68 percent of the firewall market in 1996, but is expected to decline to 38 percent in 2001."

This is a study of what is desired in network security, and how this security can be provided. Big corporations have been installing so-called network firewalls. Smaller organizations are yet to realize the implications. A goal of this report is to present all the security issues, a skeleton of solutions, and further pointers.

In section 1, we summarize the attack space. In section 2, we delineate what ought to be considered network security and distinguish it from security issues that are not network-centric. In section 3, we discuss the so-called firewalls. In section 4, we propose an evaluation methodology that gauges the effectiveness of a security solution package. Section 5 presents a brief survey of products available, and developments that are on the horizon. In sections 6 and 7, we suggest security fortifications for individuals running Windows95 or Linux on

their PCs. We intend to include a section on Windows NT in future revision of this report. Section 8 concludes this report.

In an attempt to suggest the relative severity of an issue, specific numbers are used a few times in this report. The reader is cautioned that these were produced qualitatively without any support from systematically gathered data, after a reasonably thorough WWW-search did not discover any such data.

2 The Attack Space

"The network is the computer." On the other hand, we must not view all computer security issues as network security issues. This is not only a technical mistake, but will produce a mixed-bag of "solutions". This issue has been muddled greatly in the popular/trade literature.

To provide a perspective, we reproduce a table from [FedCIRC 97]. The "Other successful attacks" category includes: break-ins, password cracking, sendmail attacks, exploitation of vulnerabilities, and misconfigured systems, Web abuse/Anon FTP abuse/software piracy, scans (ISS, NFS, NIS, etc.), social engineering, mail spamming/ spoofing, and prank/fraud issues.

Comparison of 1996 to Recent Activity
All numbers are percentages.

	1996	Jan-Mar97
Other successful attacks	51	24
Root compromises	20	23
Unsuccessful attacks	17	32
IP spoofing/denial of service attacks	6	3
Sniffer attacks	5	4

2.1 Bug Exploitation

Networking requires large programs. The technology of software development today is such that all large programs, without exception, contain bugs. It is obvious that buggy software presents security holes. Further exacerbating this unfortunate situation is the fact that Much of the networking software is based on widely available source code that was developed rather casually without too much concern for correctness. Attackers have studied this source code, and experimented with this software. Perhaps 80% of all alerts issued by CERTs

(Computer Emergency Response Team) relate to network-based security breaches through clever exploitations of bugs.

2.2 Trojan Horses, Viruses and Worms

A Trojan horse program hides a task that the user is not informed of. Among the classic Trojan horses is one that replaces the password change program. When the user changes a password, it changes the password, as would a legitimate password changing utility would, but the Trojan horse would also send the information back to the cracker. Trojan horses are not detectable by anti-virus programs. One can, after the fact/damage, determine that a program was a Trojan horse. It is nearly impossible to detect these unless human-readable source code of such programs is available. In the past, such malicious programs arrived via tapes and disks.

A virus is a program that copies itself. It is often quite a short program – a few hundred to a couple of thousand bytes long. A virus often performs malicious acts such as deleting files. Whereas a Trojan horse is delivered pre-built, a virus infects.

Today, Trojan horses, and viruses are network deliverable as Java applets, ActiveX controls, JavaScripted pages, CGI-BIN scripts, or as self-extracting packages.

2.3 Password Cracking

Secret passwords are the basis for most elementary checking *at the gate*, so to speak. Obviously, even the trusted systems can be entered if the right password is given.

> 1997.01.02, PA News: A recent survey by Compaq in the financial district of London showed that poor choices are the norm for computer passwords there. A staggering 82% of the respondents said they used, in order of preference, "a sexual position or abusive name for the boss" (30%), their partner's name or nickname (16%), the name of their favorite holiday destination (15%), sports team or player (13%), and whatever they saw first on their desk (8%).

Guessing the password of a well-placed and legitimate user based on his/her background etc. has become systematic. In addition, attackers have often grabbed encrypted password files, and using computers and large dictionaries have developed effective password *cracking* techniques.

2.4 Sniffing

A machine that is accepting all packets, no matter what the destination address the packet header contains, is said to be in promiscuous mode. In a normal networking environment, account and password information is passed along in clear-text. An attacker can obtain, by other means root access on one machine, and put that machine in promiscuous mode and sniff all the rest.

2.5 Spoofing

Where as passwords are for people and have been in use ever since multi-user systems came to exist, node authentication is nearly absent in most networks. An attacker can make his system look like one of the trusted hosts by spoofing the IP address. DNS spoofing fools your DNS server into thinking that a trusted host name belongs to an alien IP address. Routing path alterations cause your network packets to of course reach their destinations but pass via the attackers machine, where they can be captured and even altered. Web browsers and servers have also become tools in other kinds of spoofing.

2.6 Info-Leaking functions

Most network operating systems are installed with numerous services that are informative, and considered harmless a few years ago. An attacker can use such standard services as `finger`, `sendmail`, `NFS`, `NIS` available not only on Unix systems but on most other OSs that give away considerable information about the state of the system. Even a simple service like `traceroute` can provide an attacker with an understanding of the network structure.

2.7 Denial-of-Service Attacks

In a denial of service attack the attacked system is made unusable for legitimate users by loading it with intensive (but perhaps useless) computation, or by making it crash. An example is the famous SYN flood attack which sends a high volume of SYN packets to a target host and never responds with the rest of the handshake information, thus filling the targeted host's data structures.

2.8 Confidentiality and Integrity

Network packets, as in IPv4, are unencrypted. Any attacker can copy the packets with a typical PC. It takes a little more sophistication to remove the packets from the traffic, alter them, and inject the altered packets as coming from the original sender.

3 What is Network Security?

A computer system consists of many subsystems. These include file system which contains persistent data, memory system which keeps the programs in execution and some of the data, and the input-output system that this computer system is connected to. The IO subsystem includes the keyboard, the screen, the mouse and the "network". Modern operating systems provide abstractions through which a file system may be made to reside entirely in RAM, and what is perceived as keyboard input strokes are actually arriving on the network from a source thousands of miles away. From a security point of view, these can be used by the "bad guys" as parts of spoofing techniques and by the "good guys" in their defense. Issues arising out of these abstractions are hard to classify as belonging to network security.

Internet is a compendium of technologies [Comer 96]. The TCP/IP system was designed at a time when security threats were relatively unknown, and all data including various fields in the protocol headers are sent in the clear. Various protocols designed to make the network self-repairing have lent themselves to devious use.

3.1 User Authentication

Is the user who he claims he is? Ideally, this authentication ought to be made at the point of initial entry into the network, and at entry into each node on the network. Passwords have been the primary method used.

Only a few years ago, the typical Unix system would let the passwords file (/etc/passwd) be publicly readable because (a) the passwords were listed encrypted and thought to be uncrackable, and (b) a number of commonly used utilities read the contents of this file to extract such things as a user's full name, or his office room location.

Once it became known that attackers have often grabbed the file of passwords, saved it on their own machines and then, at their own leisure, cracked the passwords using simple PCs to powerful mainframes using full-sized dictionaries of English, other schemes of password protection have emerged.

We now recognize a need to educate users about properly choosing a password, and protecting it. So-called one time passwords are expected to become universal in organizations that must protect their information resources. Today, the actual password in the clear-text is not (expected to be) recorded in any file, nor is it made public-readable in the encrypted form. Before permitting entry a well-protected system asks not only for a semi-permanently assigned password, but also for a response token generated by credit-card-sized electronic devices, known as authenticators, to a challenge issued on-the-fly by the remote system being entered. In the future, biological characteristics of the user are expected to be machine-readable and become part of user authentication.

User authentication is an extremely important security item, but it is not clear if it should be considered a *network* security item.

3.2 Node Authentication

Is the node who it claims it is? Ideally, this authentication ought to be made at the point of initial entry into the network, and before every "major transaction" that it participates in.

Node authentication is nearly absent in most LANs. A machine merely declares what its IP address is and its neighbors simply believe it. Simple checks that relate the hardware address (such as Ethernet address) with IP address and with symbolic host names have always been available but are only now beginning to see widespread use. But these are easy to defeat.

Node authentication is an extremely important *network* security item.

3.3 Service Authentication

Is the user or node authorized to do a "certain thing"? The "certain things" that we wish to control have been difficult to study in a uniform setting. The most well-known example is the read-write-execute permissions on files.

Service authentication is a network security item, mainly because modern operating systems are internally organized as a networked collection of servers. Its importance depends on the particular service.

3.4 Trusted, Secure and Hardened Systems

Having taken certain measures involving careful screening, extensive testing, and verification by mathematical proof, we label certain systems as trusted. There are so-called secure OS. There are also so-called "hardened" operating systems. The National Computer Security Center, a U.S. government agency, evaluates computer systems and applies security ratings based on strict criteria.

Network firewalls are expected to be based on hardened systems. Care should be taken that these are trusted.

3.5 Data Encryption

The data portion of a network is uninterpreted by network software. Data encryption makes use of this fact, and encrypts the data before inserting the data into network packets. The encryption comes in two strengths: 40 bits and 128 bits. In the *public key* or *asymmetric* encryption systems, there is one key (the public key) used for encoding and another key (a closely held secret) for decoding. The sender encodes the data using the public key of the receiver. Such a message can only be decrypted by the owner of the secret private key, making it safe from interception. This system is also being used to create digital signatures for users.

In view of the fact that many attacks are carried out by insiders, it is now generally recommended that all network communication use data encryption.

4 Firewalls and Proxy Servers

"You have to bear in mind that there are firewalls, and there are firewalls." – Anonymous

A firewall aims to regulate traffic between two LANs, one considered the "protected" LAN and the other viewed as having malicious-intent. At one time (c1994), a firewall was a gateway/router. That is, at a minimum it was a computer system with two network interfaces, and has the ability to suppress, based on a given criteria, packets that arrive on one card from reaching the other. The two interfaces are generally located on two distinct network interface cards (NICs). All traffic from inside to outside, as well as outside to inside was forced to pass through the firewall. Yet the data contents of the packets were not examined.

18-9

Today (1997) there are commercial products that label themselves as "firewalls" that claim to provide security for PCs running Windows95 that have modem(s) but no network interface cards (NICs). They also claim that potential damage, such as eavesdropping, a worm program, file corruption, is prevented from spreading from the outside into the protected LAN.

The non-specialist computer community uses the term "firewall" as being a network security system, whereas most firewall products are packet filters and proxy servers now nicely wrapped in GUI and frequently bundled with network hardware. Such a firewall cannot guarantee protection.

Two excellent books on network security are [Chapman and Zwicky 95] and [Cheswick and Bellovin 98].

4.1 Packet Filtering

Packet filters work at the IP level. They examine the source and destination IP addresses and ports of every packet and decide which packets pass through or get blocked. No information is kept based on content of the packets, from one packet to the next one from the same client or server application. IP filters cannot examine higher protocol layers. So, this provides no way to associate a request with its corresponding reply.

Circuit (or connection) gateways are packet filtering routers, but also maintain limited state. An *address translating firewall* hides the internal IP addresses by translating all addresses as they cross the firewall. *Stateful inspection* in a packet filter refers to the ability of some filters that remember which port numbers are used by which connection. Such filters shut down access to the port when the connection is terminated. Some applications, such as FTP, open multiple ports and may fail to close them. A well-known attack technique is to look for a high port (i.e., a port higher than 1024) in the hope of finding an open one left by a long gone FTP user.

4.2 Bastion Hosts

A security system is run on a machine that is capable of being a general purpose computer system. However, to minimize exposure, general purpose user accounts are not made available. If possible, one should generate a stripped down version of the OS kernel containing only the pieces needed by the firewall. Such a system is often called a *bastion host*. Obviously, we should ensure that all traffic from inside to outside, as well as outside to inside

passes through the firewall. The bastion host runs software which implements the policy "that which is not expressly permitted is prohibited". This policy is implemented at both the IP filtering level and application level.

4.3 Proxies

Proxy servers generate a much finer level of control than a packet filter. These are also called *application gateways*. Proxies are typically run on a bastion host.

An internal client connects to the proxy and then the proxy connects to the external server. To the internal client, the proxy looks like a server, while it looks like a client to the external server. A separate proxy is required for each type of application; e.g., each of FTP, HTTP and DNS will require a different proxy. Using proxies, we can block Java applets, ActiveX controls, and cookies.

5 Desired Features

In this section, we list desired features of a security system. Security is a system *quality* that begins with the deployment of a good security system but must continually be supported active and alert systems administrators, and wise management decisions regarding user access to the Internet and other computer resources. We also rate the relative importance of the feature on a scale of 1 to 10. Unfortunately, we cannot offer an objective basis, at this time, for these ratings. It would be a worthwhile research effort to explore this domain both to make this list a comprehensive list, and to elevate the ratings to objective levels.

There is a draft of the Federal Government Firewall Protection Profile now published [NIST 97]. A security system should come with support for this.

5.1 Transparency of Use

Psychologically speaking a truly transparent security system is undesirable. On the other hand, if gateways are a nuisance to use, users will find ways of avoiding them. E.g., are users required to have accounts on the gateway? Rating: 5.

5.2 High Performance

Packet filtering, proxy services, and data encryption all rob performance of the network. Among the desired features listed here this is the only one that can be measured objectively. One can measure network throughput and speed with and without security measures in place. Rating: 8.

5.3 Authentication

The security system must use advanced authentication measures. It should support most popular access control mechanisms on a per user basis: Time of Day, Day of Week, Date, Source Address, Source Port, Destination Address, and Destination Service. It should be able to permit or deny services to specified host systems.

It should authenticate host nodes. It should prevent hijacking of the connection.

5.4 Data Encryption

Data encryption should be used by default even though it is bound to decrease the throughput. Rating: 6.

5.5 Filtering

IP filtering, and address translation must be provided. In general, the security system should make it difficult to synthesize the infrastructure of the protected network. This may include e-mail address filtering also. Rating: 9.

5.6 Proxies

A security system should be flexible enough to permit proxies for every network service. More specifically, it should be pre-configured with proxies for FTP, telnet, HTTP. It should have integrated virus scanning, and be able to forbid the download of active (ActiveX, Java etc.) content. Rating: 9.

5.7 Multiple Bastions

It is better to have filtering on one machine and a proxy or masquerading based firewall on another instead of all in one. Also, with high traffic organizations one would like to distribute the work of the firewall.

5.8 Ease of Administration

Security system deployment is no easy task. Managing a security system is a critical task and calls for intimate knowledge of various topics including system management, network protocols, and operating system internals. In addition, a complete understanding of the infrastructure of the deployment site is a prerequisite. Security systems, much like the corporate network, are living and growing entities that must be constantly maintained.

A bastion host can be detected by `traceroute` to find out where the connects are stopping. This host becomes an attacker's target and therefore we must take great care in administering it and even make it physically secure.

Periodic security audits must be performed frequently by independents as well as the administrator. Several audit tools are well-known, both to system administrators and, not surprisingly, also to attackers. A security system should permit automation of the audits. It is simple-mindedness to view these tools as the attackers' tools and hence "bad." All the audit tools we mention in this report are written by responsible and well-intentioned people.

Subscription to security alerts issued by the various authorities (see the list of Web sites in the References) should be a built-in feature to keep abreast of the latest attacks, and patches to existing programs. Rating: 4.

5.9 Detection and Documentation of Intrusions

The security system should have an always-on logging facility that logs all attempts to connect to the protected LAN, attempts to connect to the Internet, and problems with firewall software. The size of the audit records produced in a day of normal use can be large. Manual review of this much data by even a skilled system administrator would take too long and become tedious enough to miss crucial aberrations. There should be tools to facilitate the examination of these logs. Rating: 9.

5.10 Attack Recognition in Real-Time

An attack should be recognized as soon as possible. It should generate asynchronous alerts via email, alpha-pagers, or voice mail so that a system administrator can begin active watching. Rating: 5.

5.11 Facilitate Security Policy Formulation

The default policy should be "deny all services except those specifically permitted". Invariably, security policies change, and new services need to be added.

Packet filtering rules that we wish to use because of practicalities are often complex. IP filtering language should be flexible, user-friendly to program, and able to filter on as many attributes as possible, including source and destination IP address, protocol type, source and destination TCP/UDP port, and inbound and outbound interface. For example, suppose we wish to prohibit export of data via FTP through the firewall. FTP can be performed via an autonomous FTP relay, and/or by a HTTP relay. These relays may or may not have a common configuration file or a unified means of authenticating users. There might even be an HTTP cache behind the firewall requiring multi-machine configuration with subtle inter-dependencies. Several typical combinations such as these should be automated.

Pretty GUI are now becoming common. However, it should be remembered that the general experience in the software world is that textual languages are far more expressive than GUIs, even if difficult to use and conceptualize. Rating: 4.

6 Survey of Developments and Products

The TCP/IP system was designed at a time when security threats were relatively unknown. Many insecurities of IP and TCP are discussed in the literature [Bellovin 89, Guha and Mukharjee 97]. We expect to see a strengthening of the protocols and built-in data encryption.

6.1 IPv4 to IPv6

IP version 6 is the successor to IP version 4. There is no IPv5. IPv6 increases the IP address size from 32 bits to 128 bits, to support more levels of addressing hierarchy, greater number of nodes, and simpler auto-configuration of addresses. IPv6 has simplified header format. It

has improved support for extensions and options to allow for more efficient forwarding. It can label packets as belonging to particular traffic flows that can request special handling, such as 'real-time' quality of service. More importantly for us, it has extensions to support authentication, data integrity, and data confidentiality.

6.2 SSL and SHTTP

These are proposed encryption and user authentication standards for the Web.

SSL (Secure Socket Layer) is the scheme proposed by Netscape Communications Corporation (http://home.netscape.com/newsref/std/SSL.html). It is a low level encryption scheme used to encrypt transactions in higher-level protocols such as HTTP, NNTP and FTP. The SSL protocol includes provisions for server authentication, encryption of data in transit, and optional client authentication.

Secure HyperText Transfer Protocol (SHTTP) adds to the HTTP that Web uses. SHTTP (www.eit.com/creations/s-http/ is proposed by CommerceNet, a coalition of businesses interested in developing the Internet for commercial uses. It is a higher level protocol that only works with the HTTP protocol, but is potentially more extensible than SSL. SSL and SHTTP can be used together.

6.3 SOCKS

SOCKS is a networking proxy protocol. SOCKSv5 adds authentication and authentication method negotiation, message integrity and privacy, and UDP proxy to SOCKS v4 functionality. The protocol is conceptually a "shim-layer" between the application layer and the transport layer, and as such does not provide network-layer gateway services, such as forwarding of ICMP messages. The implementation of the SOCKS protocol typically involves the recompilation or relinking of network applications to use the appropriate encapsulation routines in the SOCKS library.

6.4 Commercial Products

NCSA (www.ncsa.com) publishes a list of products that have been tested by them and received certification. As of Sep 1997, NCSA has certified the following firewall products (see www.ncsa.com/fpfs/fwcert.html).

- www.3Com.Com NETBuilder Router v9.1

- www.ans.net/InterLock ANS Interlock

- www.ascend.com Pipeline Router Ver. 4.6C

- www.bull.com Netwall 3.0

- www.checkpoint.com CheckPoint Firewall-1

- www.cisco.com/pix PIX Firewall and Centri Firewall, Ver. 3.1

- www.cyberguardcorp.com CyberGuard FirewallV2.2.3

- www.altavista.software.digital.com AltaVista (UNIX and NT)

- www.gta.com/firewall.html GFX Internet FirewallSystem V2.5 and GNAT Box 2.0

- www.ibm.com/security Firewall, Version 3.1 and IBM Firewall for AS/400, Version 4, Release 1

- www.internetdevices.com AFS 2000 v2.02

- www.lsli.com Portus, version 2

- www.milkyway.com Black Hole

- NTFirewall.com Guardian 2.0

- www.network-1.com Firewall/Plus 1.1-4 and Firewall/Plus for NT

- www.on.com ON Guard

- www.openroute.com GT Secure 60 and GT Secure 70

- www.radguard.com PyroWall Ver. 1.1, and CryptoWall V1.0

- www.raptor.com Eagle 4.0 (Solaris, NT and HP UX)

- www.sctc.com BorderWare V4.0, and Sidewinder 3.0

- www.incog.com SunScreen SPF-100 1.0

- www.tlogic.com Interceptor

- www.tis.com Gauntlet Internet Firewall System

- www.ukiahsoft.com NetRoad FireWALL

- www.watchguard.com Watchguard Security Management

The firewall products currently (1997) range in price from $600 to $20,000. As of 1996, Check Point Software Technologies was the leader in firewall technology with 35 percent market share, followed by Cisco and Trusted Information Systems with 8. The Gartner Group predicts that by the year 2000, there will be roughly five firewall suppliers from which to choose: Check Point Software, Cisco, CyberGuard, Raptor Systems, and TIS.

7 Security Fortification for Personal Machines

Perhaps as many as 90% of the 20 million nodes connected to the Internet are personal machines, the rest being various servers. Perhaps 80% of these personal machines, and nearly 100% of those behind firewalls in private LANs, are running Windows (95, NT or 3.x) and Linux with little supervision from system administrators, the remaining systems being other variants of Unix and Macintoshes. Note that the 20 million count does not include home-based machines that connect via PPP, even though once PPP-connected they are no less vulnerable than the nodes permanently connected to a LAN. Very little security fortification above what was there right out-of-the-box is done for personal machines. In this section, we advice on additional protection.

For comic relief and to experience serious nuisance, we highly recommend a visit to

www.digicrime.com, "A full service criminal computer hacking organization."

7.1 General Precautions

Most services that are defined in the services are unneeded by a typical personal machine. Services such as systat, netstat, tftp, finger, sunrpc, and nfs have been exploited. Apply a filter that makes sure that packets coming in from the outside network do not have source IP address that match the inside network. Read [Cobb 96].

Choose passwords carefully[Microsoft].

Use PGP (www.pgp.net/pgpnet/pgp-faq).

7.2 Web Browsers and Servers

Web browsers are common place now, and even though most organizations are forbidding personal Web-servers we predict that a decentralization of Web-servers is around the corner. While most users have become good surfers, many are unaware of the loss of privacy and security that can happen. We recommend reading [Felten et al. 96]. Much functionality is added by ActiveX controls and Java applets, and it is hard to recommend that they be summarily blocked. Techniques of selectively blocking such active content at gateways are now being developed[Martin et al. 97]

7.3 Windows 95

For what it is worth, enable user profiles.

Use a winsock add-on which "socksifies" any winsock application. There are several such add-ons available (www.socks.nec.com), including some that are free.

7.4 Windows NT

Windows NT is now considered the primary target for attack. Audit the security of your system periodically using

SPI for NT, the Security Profile Inspector from
ciac.llnl.gov/cstc/spi/spiwnt/spiwnt.html.

Keep abreast of NT security issues (www.ntsecurity.net).

7.5 Linux and other Unix Variants

Unix systems have long been the host operating systems for servers, and perhaps are the most attacked systems. For the same reasons, there are many quality products some free of cost that can help (see www.cs.purdue.edu/coast).

SATAN can search, in a matter of minutes, for several known security holes, such as the ones in NFS, sendmail, TFTP, X11 servers, and FTP. Crack is a program that can systematically guess a password of a user. Tripwire program periodically scans your system and detects if any system files or programs have been modified. Esniff run on your network can quickly establish how effective it is in compromising local machines.

Unix systems can serve as excellent bastion hosts, and Linux (www.linux.org) is very good at this. In a follow-up study, we intend to build a low-cost hight performance firewall based on Linux as an actual demonstration.

8 Conclusion

This report is a study of what is desired in network security products. We outlined an evaluation methodology that gauges the effectiveness of a security solution package. We also suggest security fortifications for individuals running Windows95 or Linux on their PCs.

References

[CERT] CERT Coordination Center advisories, posted to the newsgroup comp.security.announce, and archived at ftp://ftp.cert.org/pub

[Cobb 96] Stephen Cobb, The NCSA Guide to PC and LAN Security, McGraw-Hill, 640pp, May 1996, ISBN: 0079121683

[Chapman and Zwicky 95] D. Brent Chapman and Elizabeth D. Zwicky, Building Internet firewalls, O'Reilly & Associates, c1995, ISBN: 1-56592-124-0

[Cheswick and Bellovin 98] William R. Cheswick and Steven M.Bellovin, Firewalls and Internet Security: Repelling the Wily Hacker, Addison-Wesley, c1994, ISBN: 0-201-63357-4.

[Comer 96] Douglas E. Comer, Computer Networks and Internets, Prentice Hall, ISBN: 0132390701.

[Dean et al 96] Drew Dean, Edward W. Felten, and Dan S. Wallach, Java Security: From HotJava to Netscape and Beyond, Proc. of 1996 IEEE Symposium on Security and Privacy. www.cs.princeton.edu/sip/pub/secure96.html

[Digicrime] Digicrime, Online Security Experiments, Digicrime, Inc. www.digicrime.com

[FedCIRC 97] FedCIRC, Practical Intrusion Detection Seminar, April 23-24, 1997, fedcirc.llnl.gov

[Felten et al. 96] Edward W. Felten, Dirk Balfanz, Drew Dean, and Dan S. Wallach, Web Spoofing: An Internet Con Game, Technical Report 540-96, Dept. of Computer Science, Princeton University, Dec 1996.

[Foresight 97] The Foresight Computer Security Fact Forum, discuss.foresight.org

[Garfinkel and Spafford] Simson Garfinkel and Gene Spafford, Practical Unix and Internet Security, second edition, April 1996, 1004 pages, ISBN: 1565921488.

[Martin et al. 97] D. Martin, S. Rajagopalan and A. Rubin, "Blocking Java Applets at the Firewall", in Proceedings of the InternetSociety Symposium on Network and Distributed System Security, February 10-11, 1997.

[Microsoft] Guidelines for Strong Passwords, www.microsoft.com/ security/ guide.htm

[NCSA] NCSA, The NCSA Firewall Product Developers' Consortium Criteria, www.ncsa.com/fpfs/fwct20.html

[NIST 97] Draft Federal Government Firewall Protection Profile csrc.nist.gov/nistpubs/cc/pp/pplist.htm

[Princeton 97] "Security Tradeoffs: Java vs. ActiveX, An Unofficial View from the Princeton Secure Internet Programming Team", 1997. www.cs.princeton.edu/sip/

[Security Related Organizations] .

   www.cert.org A 24-hour Computer Emergency Response Team.
   ciac.llnl.gov Computer Incident Advisory Capability, U.S. Department of Energy.
   cpsr.org.home Computer Professionals for Social Responsibility.
   www.eff.org Electronic Frontier Foundation.
   www.ncsa.com National Computer Security Association.
   epic.org Electronic Privacy Information Center.
   www.first.org Forum of Incident Reponse and Security Teams
   www.isoc.org Internet Society.
   csrc.nist.gov NIST Computer Security Resource Clearinghouse.

[Stein 97] Lincoln D. Stein, WWW Security Faq, Version 1.4.1, September 2, 1997. www.w3.org/Security/Faq This FAQ will soon be a book *The Web Security Reference Guide*, Addison-Wesley Longman, in December 1997.

OPTIMAL STRUCTURAL DESIGN OF MODULAR COMPOSITE BARE BASE SHELTERS

Mansur Rastani, Ph.D.
Associate Professor
Department of Manufacturing Systems

North Carolina AT State University
1601 East Market Street
Greensboro, NC 27411

Final Report for:
Summer Research Program
Air Logistic Center

OPTIMAL STRUCTURAL DESIGN OF MODULAR
COMPOSITE BARE BASE SHELTERS

Associate Professor
Department of Manufacturing Systems
North Carolina AT State University

Abstract

This reports reflect the result of my research work at McClellan Air Logistics Center during summer 1997. The work was to optimally design a composite structural shelter for better operational characteristics compare to the current metallic bare base shelters and to reduce the airlift requirement. An analytical engineering design approach was implemented , and a composite along with a spreadsheet programming were developed for the analysis. The structural design included the weight of the materials, snow load. personnel load, wind gust of 110 mph, and a seismic zone of 3.

The optimal design concluded a 2"-thick web-stiffened core and a 0.05" thickness for the facing of the roof and wall sandwich panels. The floor panel, however, uses a 0.06" facing and similar core design. The webs are 0.0625" thick and placed in the core at 1.5" spacing. The material for facing is quasi-isotropic (0/90/45/-45) glass/epoxy laminate and polyurethane foam was used for the core. The material (45/-45) glass/epoxy was selected for the web stiffeners for its high shear resistance. The design of the beam and frame supports resulted in a 0.125"-thick rectangular pipe of 2"x2" dimension, using (0/90) graphite/epoxy composite. The connection beams along with all other interfacial components use an I-beam of 0.1" thickness and 2" depth of the same material as in beam supports.

The result of the weight analysis concludes a total weight of 12,947 lb. for the composite GPS shelter and 13,742 lb. for Composite ACH shelter. Composite GPS shelter weighs only 25% per volume of the weight of the current metallic MERWS shelters.

Further study is recommended for the stress analysis of the shelter under thermal, impact, and creep loading. Also, the manufacturing process of the shelter fabrication is required to be analyzed for the selection of the optimal processing method and feedbacking into design process.

OPTIMAL STRUCTURAL DESIGN OF MODULAR COMPOSITE BARE BASE SHELTERS

Mansur Rastani, Ph.D.

Introduction

Developments in lightweight, highly durable composite materials offer opportunities to develop bare base shelters that will require less airlift to deploy. Existing and developing technology offers the potential to significantly improve existing family of portable shelters by reducing shelter weight, packing volume, and installation time/manpower while improving reliability, durability, and maintainability.

The ultimate objective of this effort was to introduce an air-transportable, rigid-wall Air Force Bare Base shelter which uses less airlift requirements and yet has better operational characteristics than do current similar shelters. The scope of this research project spans the engineering optimal structural design of the required components as well as material selection for the next generation of bare base shelters. The necessary design requirements used for this development were established in accordance with the terms stated in the Operational Requirements Document for a New Family of Portable Shelters, ORD-CAF 316-92-I-B, Air Combat Command; and MIL-STD-907B, Military Standards, Engineering and Design Criteria for Shelters.

Background

The identified existing hard-wall shelters which was used as an initial model to this development effort were Harvest Falcon General Purpose Shelters (GPS) and Aircraft Maintenance Hanger (ACH), (ref. 7). The older is used for a variety of bare base operations such as maintenance shop, personnel office, and warehouses. The later is used for aircraft and vehicle maintenance. Both units consist of a series of freestanding arches. These arches are made from individually shipped aluminum beams and honeycomb core with aluminum skin facings. These bare base shelters are too airlift intensive to support any anticipated force structure and power projection operation of the future. They are bulky and difficult to handle, erect, and maintain; provide only limited protection from the environment; are the product of decades-old technology; and will exceed life expectancy by the year 2000.

Methodology

Today the use of composite materials in structures of all kinds is accelerating rapidly, with the major impact already being felt in the aerospace industry where the use of composites has directly enhanced the capability of fuel-efficient aircraft in the commercial arena and new-generation aircraft in the military sphere. Many fiber-reinforced composite materials due to their low specific gravities, their specific strength and specific modulus are markedly superior to those of metallic materials (ref 12). In addition, fatigue strength-to-weight ratios, as well as fatigue damage tolerances, of many composite laminates are excellent.

Fiber-reinforced composite materials consist of fibers of high strength and modulus embedded in or bonded to a matrix with distinct interfaces between them. In this form, both fibers and matrix retain their physical and chemical identities, yet they produce a combination of properties that can not be achieved with either of the constituents acting alone. The principal fibers in commercial use are various types of glass and carbon, as well as kevlar. The most common used matrix are polymeric materials: epoxy, polyester, and vinyl ester.

The most common form in which fiber-reinforced composites are used in structural application is called a laminate. Laminates are obtained by stacking a number of thin layers of fibers and matrix and consolidating them into the desired thickness. Fiber orientation in each layer, as well as the stacking sequence of various layers can be controlled to generate a wide range physical and mechanical properties for the composite laminate. Another form of structural composite is foam-filled sandwich panel. These structures have an intriguing combination of properties. They are light, stiff, strong, provide excellent thermal insulation, and can be made into shapes which are both structurally efficient and esthetically pleasing. A composite sandwich consists of top and bottom facings and a core. The facings are usually thin while the core is thick. The material for facing is made of some composite laminate such as glass or carbon epoxy and for the core of some type of foam such as polystyrene or polyurethane. This form of structural composite has been selected for the undergoing research project.

There is major difference in stress analysis of these type structures compare to conventional ones. That is the need to account for shear deformation of the core which is so slight in normal structures that can be neglected. In foam-filled structures, however, the shear deformation is appreciable. In these structures, flexural stresses are carried by the faces and shear stresses by the core. Since these panels in the undergoing study are bending in every-direction, the faces are required to act as an isotropic materials. For this reason the sandwich faces are required to use multidirectional plies (i.e., 0/90/45/-45) to produce a quasi-isotropic laminate. The core may be web stiffened to increase the shear rigidity, overall flexural stiffness, as well as the buckling resistance of the panels.

The study uses an analytical design approach for the structural design analysis. A composite computer programming (UNILD) is used for the calculation of roof and floor panel structures. Also a spread sheet analysis was developed to determine the weight of the modular composite shelter's components and subassemblies.

Design Criteria

ORD-CAF 316-92-I and MIL-STD-907B, (ref 2, & 4) specify a uniform snow load of 40 psf and a static concentrated force of 660 pounds at the weakest area (i.e., center of the panel) of the roof. The floor loads are specified as 65 psf uniform load and 2000 pounds concentrated force over a 4-ft^2 area at the center of the floor panel. They also specify a 100 mph wind gusts and a solar load of 205 °F w/no permanent deformation. Neither document specifies requirements for seismic load, panel maximum-permissible deflection, and a safety factor for any potential failure mode. The design criteria used in this

study were a seismic zone of 3, an allowable deflection of less than the panel thickness, and a safety factor for each potential failure mode of at least 1.5. The dimension used for the analysis of the composite GPS and ACH shelters are 20'x40'x48' and 25'x50'x48' (ref. 7) respectively.

Modular Composite Structural System

The proposed composite shelter structure incorporates six freestanding arch modules. Each module consists of composite-sandwich roof and floor panels. Each panel has two quasi-isotropic E-glass/epoxy composite faces and a foam core, and is pultruded with an integrated box beam at each side. Upon assembling of a module, these beam boxes establish the structural arch frame for these panels. The box beams are made of graphite/epoxy composite and are connected through an inserts and more than four bolts on each side to provide the rigidity stiffness for the arch. Also, the box beams on floor panels play the role of a tension bar for the arch frame at the bottom end of the arch which promote the arch rigidity support for the module. The arch members are joined through the lateral connection beams. Similar connection beams are used to tie the modules together. Each shelter uses two walls at the ends. Figures 1 shows typical structural modules for composite GPS and ACH shelters.
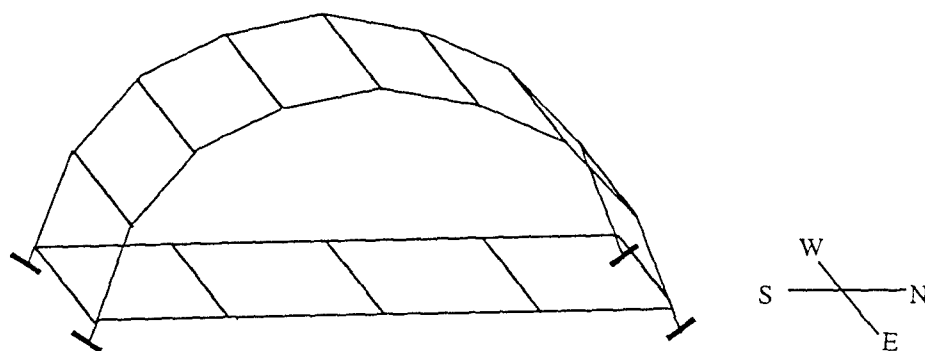
Figure 1. Composite ACH Module

Seismic Load /Wind Analysis

The Design wind pressure shall be determined in accordance with the following formula (ref. 9):

$$p = C_e \cdot C_q \cdot Q_s I$$

An exposure coefficient C, most severe exposure, ($C_e = 1.2$), a wind speed of 110 mph ($Q_s = 31$ psf), and an essential facilities (importance factor of I =1.25) have been used for the seismic analysis. The structure is assumed as a windward wall ($C_q = 0.8$).

Thus,

p = (1.2) (0.8) (31 psf) (1.25) =40 psf

The total wind load (P) in the N-S direction then is:

P = p x Area = 40 psf x (8'x25') = 8,000 (lb.)

The design seismic force in a given direction shall be determined by the following formula (ref. 9):

$$V = (Z \, I \, C \, W) / R_w$$

A seismic zone of 3 (Z = 0.3), a maximum value of 2.75 for C coefficient (dictated by UBC code (ref. 9) regardless of site characteristics and structure period), the same value for I as in wind load analysis, and a numerical value of 10 for R_w (table 23-O, and 23-Q of UBC code) are used. Also a weight of 10 psf are assumed for the structure, which yields a total load of W = 10 psf x (8'x50') = 4,000 (lb.)

Thus,

V = (0.3x1.25x2.75 x 4,000 /10 =415 (lb.)

UBC 2312-2, dictates that when the code-prescribed wind design produce greater effects (i.e., P=8,000lb>V=415lb), the wind design shall govern. Hence the design of the composite shelter module shall be based on wind pressure. Figure 2 shows the shelter module projected area for the wind and the calculated dead load of the structure.
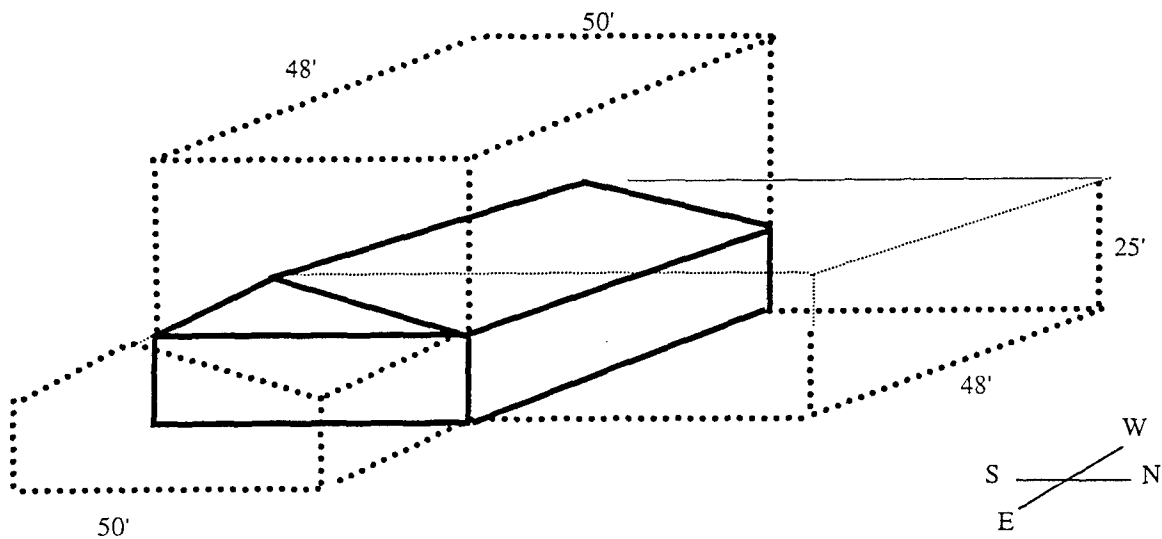
Figure 2. Shelter's Schematic Projected areas for downward, N-S, and E-W wind loads

Design of Modular Composite Shelter

The Shelter consists of six modules. Each shelter module consists of roof, floor, and two arches subassemblies. The arches consist of member elements with rectangular box cross sections attached at their ends through rigid joints. The two arches are joined through lateral connection beams and their bottom ends

are tied together by a box beam bar. The roof and floor subassemblies consists of sandwich panels connected at their ends through rigid inserts. Figure 3 shows a ACH arch frame.
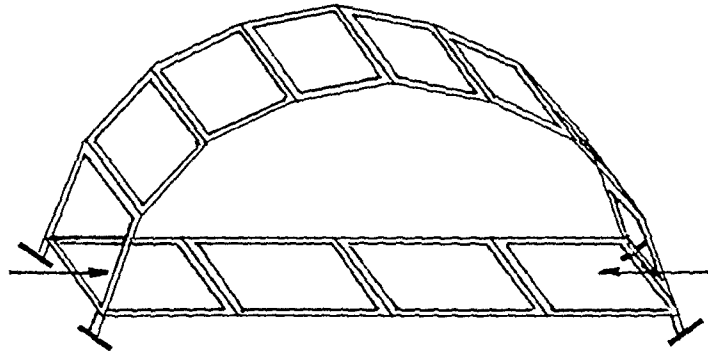
Figure 3. ACH arch frame module

Arch Member

The loaded ACH arch is shown in figure 4. The total downward load and the horizontal wind load are calculated,
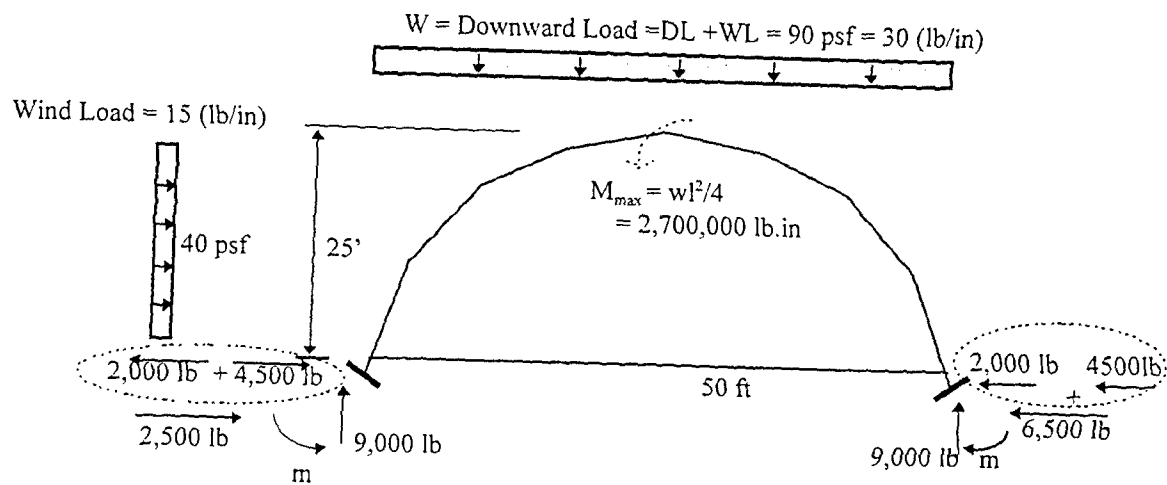
Figure 4. Downward and horizontal wind loads on composite ACH module

The total downward uniform load on the module is calculated as follows:

W = Downward Load =Dead Load (DL) +Wind Load (WL) = $[(40^{psf}+10^{psf})+(40psf)]$ (8'x50')

$$= 36,000 \text{ (lb.) } /2 \text{ arch units } =18,000 \text{ (lb.)}$$

The vertical support reaction then is, $V = 18,000/2 = 9,000$ lb.

The intensity of the distributed load then is,

$w = W/L = 18,000 / 50$ ft $= 360$ (lb./ft) $= 30$ (lb./in)

The total horizontal uniform load is,

WL $= 40$ psf x 8' x 25' $= 8,000$ (lb.) / 2 arch units $= 4,000$ lb.

The horizontal support reaction due to wind load then is, $H_w = 4,000/2 = 2,000$ lb.

The intensity of the wind load then is,

$w = WL / a = 4,000 / 25' = 160$ (lb./ft) $= 15$ (lb./in)

The horizontal support reactions due to arch moment is

$H_m = \frac{1}{2}(M_{max} / h) = 1/2$ x $2,700,000 / 25$x$12 = 4,500$ (lb.)

The design of an arch to resist bending moment could be approximated (ref. 10) by calculating the section of the arch at the support where the force is at its largest magnitude. In such an approximation a nominal factor of safety of five or more is usually assigned to the ultimate compressive strength of the material used. Thus, the support reaction is calculated,

$R = \{(6,500$ lb.$)^2 + (9,000$ lb.$)^2 \}^{1/2} = 11,100$ (lb.)

Using a typical ultimate compressive strength (S_u), for a graphite/epoxy composite (ref. 12), that is about 75,000 psi (refer to appendix) and using a safety factor of N=5, the cross sectional area of the arch member is,

$S_a = S_u / N = 75,000$ psi / $5 = 15,000$ psi

$A = R /(S_a) = 11,100$ lb. / $15,000$ psi $= 0.75$ (in^2)

At later section, the roof panel is designed, and the resulting thickness is calculated to be 2". Hence the arch section is selected to be a hollow rectangular box with a dimension of 2", since it will be the integral part of the pultruded composite panel. The thickness of the box then is calculated to be 0.1". The cross section of the arch member is shown in figure 5.
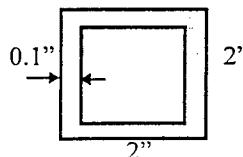
Figure 5. Cross section of the Arch member

Lateral Connection Beam

Lateral connection beams connects the two arches of the shelter module, and provide sidelong stiffness against wind load for the module. The wind load (WL) in the w-e direction is,

WL $=$ (wind pressure) x Area $= (40$ psf$) (50'$x$15' + 50'$x$10'/2 = 1000$ ft$^2) = 40,000$ (lb.), which is supported by six module, since each module has nine connection beams the amount of wind load on a lateral beam

then is equal to 1/54 WL = 750 (lb.). The same material as in box beam is used for the lateral connection beam (i.e., 0/90 graphite/epoxy). Then the sectional area for the beam is, A=750 lb. /15,000 = 0.05 in^2, which is trivial. The sectional area of the connection beam then would be evaluated based on the geometrical requirement of the panels. In the later sections the thickness of the roof panels is calculated to be 2". Based on the panel thickness the minimum sectional area for the connection beam is determined, as shown in figure 6. The length of the beam member is the same as the module width that is 96".
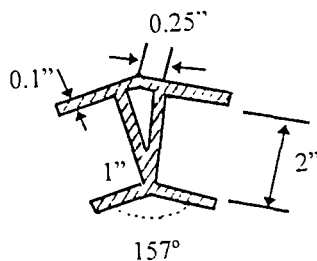
Figure 6. Sectional area for the lateral connection beam

The assembled panel using the connection beam and the insert is shown in figure 7. The insert component is made of (0/90) graphite/epoxy. The box beams are connected through insert using at least four bolts on each side of the connection to provide rigid joint for the arch. For other type joints, it is required to use roof trusses supports for the arch to create enough flexural rigidity for the shelter module. The latter is not covered in this report since it was beyond the scope of this short term research work.
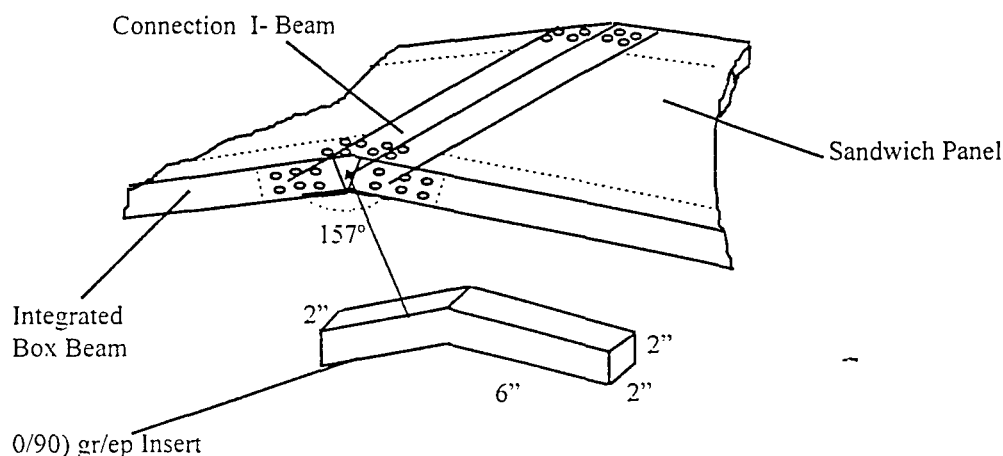
Figure 7. Assembled panels using connection beams and inserts

Composite Roof Panel

The length and width of the panels are determined based on the maximum running length capacity of the current pultrusion machines and the required geometry of the shelter. This criteria determines a dimension of 8' x 10' for the panels. The sandwich panels are simply supported along the edges. To design the panel for the facing and core thickness, the amount of necessary stiffness for handling the loading condition with respect to deflection constraint is required. This requirement could be evaluated by first assuming a steel plate rather than a sandwich panel which has a more complex design.

Assuming a 8'x10' steel plate (a=8', b=10') with a typical allowable bending stress of S_a =30,000 psi, under a total downward load of w= 90 psf =0.626 psi, the required thickness is calculated as follows: (ref. 8 & 13), $t = a [(k.w/S_a]^{1/2} = 0.306$", (k=0.487 for a/b=10/8=1.25, table 5.2.20, ref. 8 & 13)

Steel plates have a typical young modulus of 30e6 psi, and Poisson's ratio of v=0.3. Hence, the flexural stiffness requirement is: $D = E.b.t^3/12(1- v) = 8.367e06$ lb.in^2.

To calculate the optimal thickness for the faces (t_f) and core (d) of the sandwich plate, the type material for the faces and the core are specified. Then, the cost function (i.e., weight of the panel) is minimized for the core depth with respect to the flexural stiffness constraint (ref. 1). Type material for the faces are selected from quasi-isotropic (i.e., 0/90/+45/-45) E-glass/epoxy laminates which are structurally stiff enough for the panels and cheaper compare to the graphite/epoxy laminate. Polyurethane foam is selected for the panel's core. For information on mechanical properties of these materials refer to appendix. Now consider a 8'x10' sandwich panel under the uniform downward load of 90 psf, figure 8.

Total Load (P) = 90 psf x 80 ft^2 =7,200 lb
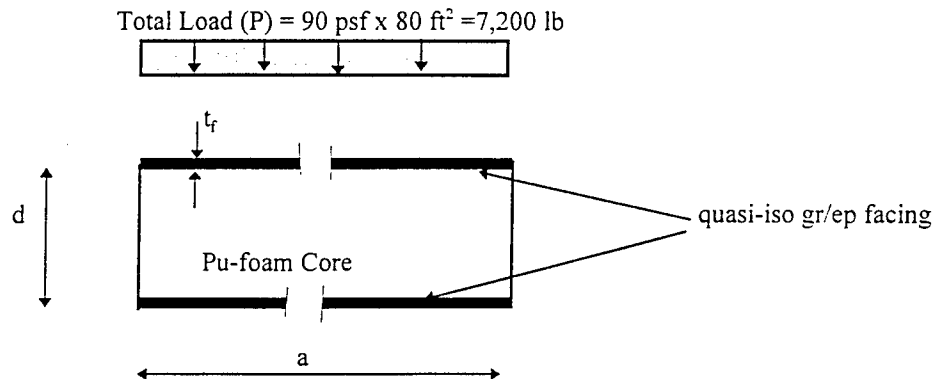
Figure 8. Composite sandwich panel under downward uniform load

The objective function for the panel weight is, $W_p = \mu_c.b.d +2 \mu_f.b.t_f$, where μ_c, and μ_f are the weight density of facing and core respectively. The faces in a sandwich panel provide flexural stiffness and the core provide shear rigidity . The equation for flexural stiffness of the sandwich face is given by, $D_f = E_f.b.t_f.d^2/2(1-v_f^2)$. The facing thickness is derived from this equation and substituted in the objective function. The objective function in terms of core depth could then be written as follows:

$W_p = \mu_c.b.d + 4 \mu_f. (1-v_f^2).D_f / E_f.d^2$

For a minimum value of W_p with respect to d, the condition, $d(W_p) / d(d) = 0$, should be satisfied. That is,

$\mu_c.b - 8 \mu_f. (1-v_f^2).D_f / E_f.d^3 = 0$,

or

$$\boxed{d = [8 \mu_f. (1-v_f^2).D_f / E_f. \mu_c.b]^{1/3}}$$

which is the optimum core depth. Using the required flexural stiffness, $D_f = 8.367e06$ lb.in^2, and the information on mechanical properties of the panel's facing, and core material, (refer to appendix) the results are as follows:

d = 2.05, which a nominal size of $\boxed{d = 2"}$ would be selected.

The thickness of the face from the above formula then is equal to $\boxed{t_f = .012"}$.

<u>Check on Stresses</u>: For the calculated composite sandwich roof panel, the stresses and deflection are examined. The flexural stress in the facing (ref. 3 & 5) is, $S_f = P. a/8 t_f.b.d = 30,000$ psi , which is smaller than the allowable bending stress of the face material (ref. 6), 35,000 psi, (refer to appendix) and it is O.K.. The shear stress in the core is, $S_c = P/2.b.d = 15$ psi , which is smaller than the allowable shear stress of the core material, 57 psi (ref.14), (refer to appendix) and it is ok. The composite sandwich roof panel then is checked for any possible violation in maximum permissible deflection.

<u>Check on Deflection</u>: The deflection of a sandwich panel is a function of the face flexural stiffness (D_f,), and the core shear rigidity ($D_{s.c}$). The deflection formula for the panel under a uniform distributed loading (ref. 3 & 5) is: $y = 5 P.a^3 /384 D_f + P.a /8 D_{s.c}$, the required flexural stiffness of the face was calculated before, $D_f = 8.367e06$ lb.in^2, the shear rigidity of the core is $D_{s.c} = b.d G_c = 54,240$ lb., the deflection would then be,

<u>$y = 11.3" >> d=2"$, which is in violation.</u>

A larger face thickness is selected (**$t_f=0.05"$**), and the face stress and panel deflection are calculated,

$S_f = 7,200$ psi $< S_a = 35,000$ psi, which is O.K. ,

$D_f = 35.58e6$ psi ——— <u>$y = 3.87" > d=2"$, which still is in violation.</u>

A method to solve this problem is to select a web-stiffened core for the sandwich panel, figure 9.
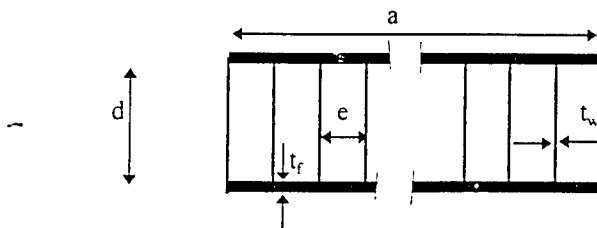
Figure 9. Web- stiffened core sandwich panel

The material for the web stiffeners shall be +-45 ply E-glass/epoxy, which has large shear strength. A thickness of (**$t_w = 0.0625"$**) is selected for the webs which are placed at (**e=1.5"**) spacing. To determine the

deflection of the web-stiffened panel, the new stiffness (D_f^*) and rigidity ($D_{s,c}^*$) should be determined. The flexural stiffness of the core is neglected, because it is too small, then, (for information on material properties refer to appendix),

$D_f^* = D_f + D_w$, and $D_s^* = D_{s,c} + D_{s,w}$

$D_w = (1/12)E_w.t_w.(b/e-1).d^3 = 4.77e6$ lb.in^2,

$D_{s,w} = t_w (b/e-1)d.G_w = 11.56e6$ lb.,

hence,

$D_f^* = 35.58e6 + 4.77e6 = 40.35 e6$ psi,

$D_{s^-} = 54.200 + 11.56e6 = 11.62e6$ psi,

thus, <u>y = 1.95" < d=2", which is within the permissible deflection.</u>

Roof panel details is shown in figure 10.
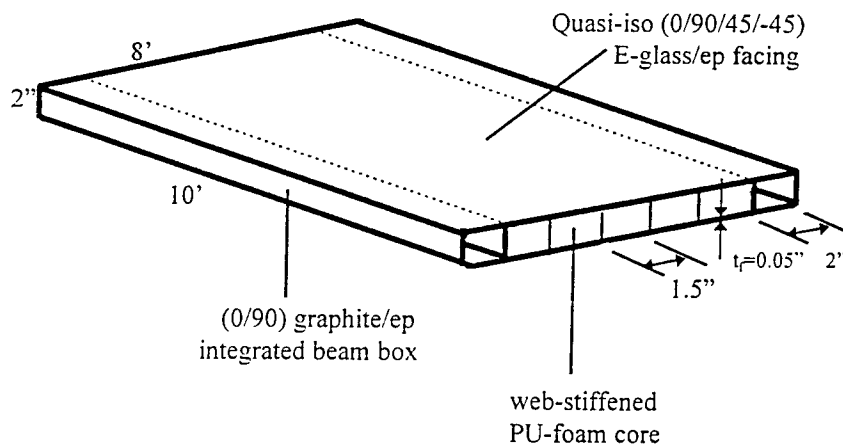
Figure 10. Composite web-stiffened roof sandwich panel

Composite Floor Panel

The floor shall be capable of supporting a uniform load of 65 psf and a point load of 2,000 lb. at the center, figure 11.
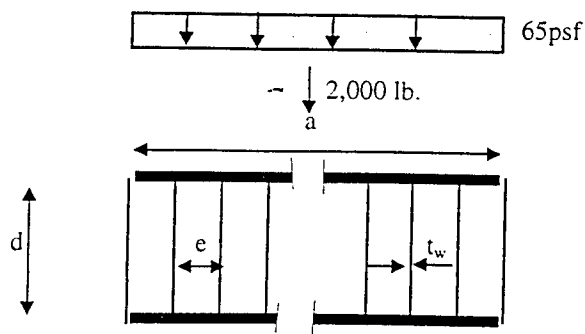
Figure 11. Composite floor panel with uniform and concentrated load

A typical roof sandwich panel of 2" thickness with the same web-stiffened core is used for the floor sandwich panel. However, a thicker facing $(t_f = 0.06")$ than the roof panel shall be used since the floor panel is experiencing a concentrated force of 2,000 lb. at the center which can dramatically increase its deflection. Check on face stress: The formula for the deflection (ref.3& 5) is,

$S_f = 9P.a/[24bdt_f + 4t_w(b/e-1)d^2]$, the total force is, $P = 65$ psf x8'x10' + 2,000 lb. = 7,200 lbs.,

thus, $S_f = 14,650$ psi $< S_{f,a} = 35,000$ psi, and it is O.K.
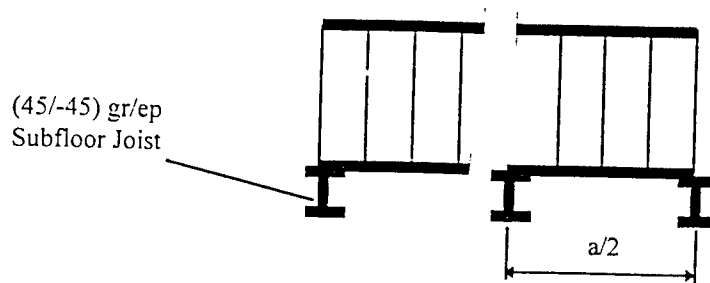
Figure 12. Composite floor sandwich panel with subfloor joists
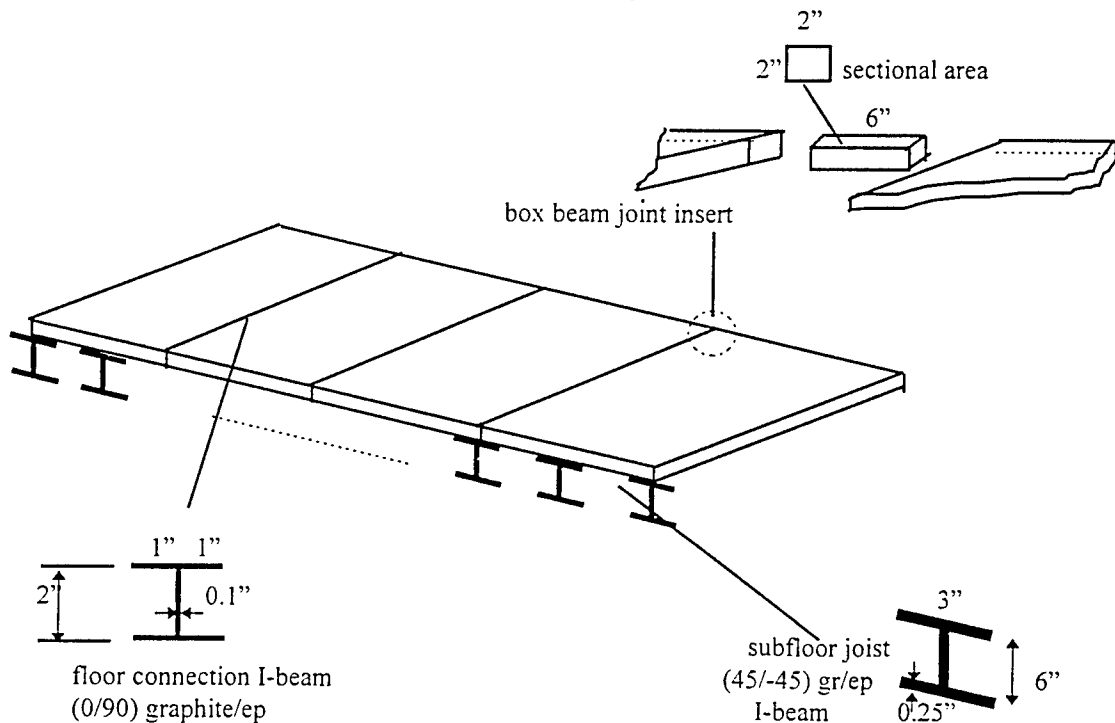
Figure 13. Composite sandwich floor panel assembly

Check on deflection: The same formula that was used in design of roof panels are used here, however the total load and the stiffness have been changed and should be calculated first, D_f^* ($t_f = 0.06$") = 47.48e6 lb.in^2, D_s^* has not been changed, hence, $y(P=7200, D_f^*, D_s^*) = 4.56" > d=2"$, which is in violation.

19-13

To solve the problem, it is recommended that the floor panel be used with subfloor joist at spacing of half length of the panel span, figure 12. The joists are made of (45/-45) graphite/epoxy composites to withstand shear loading. The new loading is calculated, P = 65 psf x4'x10' + 2000 lb. = 4,600 lb., the deflection then is, v = 0.36" < d=2" which is O.K. Four floor sandwich panels are connected through their integrated box beams by insert components and at least four bolts on each side. Figure 13 shows the floor sandwich panel assembly.

Composite Wall Panel

The same geometrical properties of the roof panel are used for the typical wall sandwich panels. However since the wall panels are under in-plane load due to the downward loading on the structure, it is required that they be checked for the critical buckling load and any possible violation. The amount of the in-plane load carried by one wall is calculated.

P = (total downward load on GPS shelter) / [(# of walls)x(# of panels in a raw per walls)

=(90 psf x 40' x 48')/(2x4) = 21,600 lb.
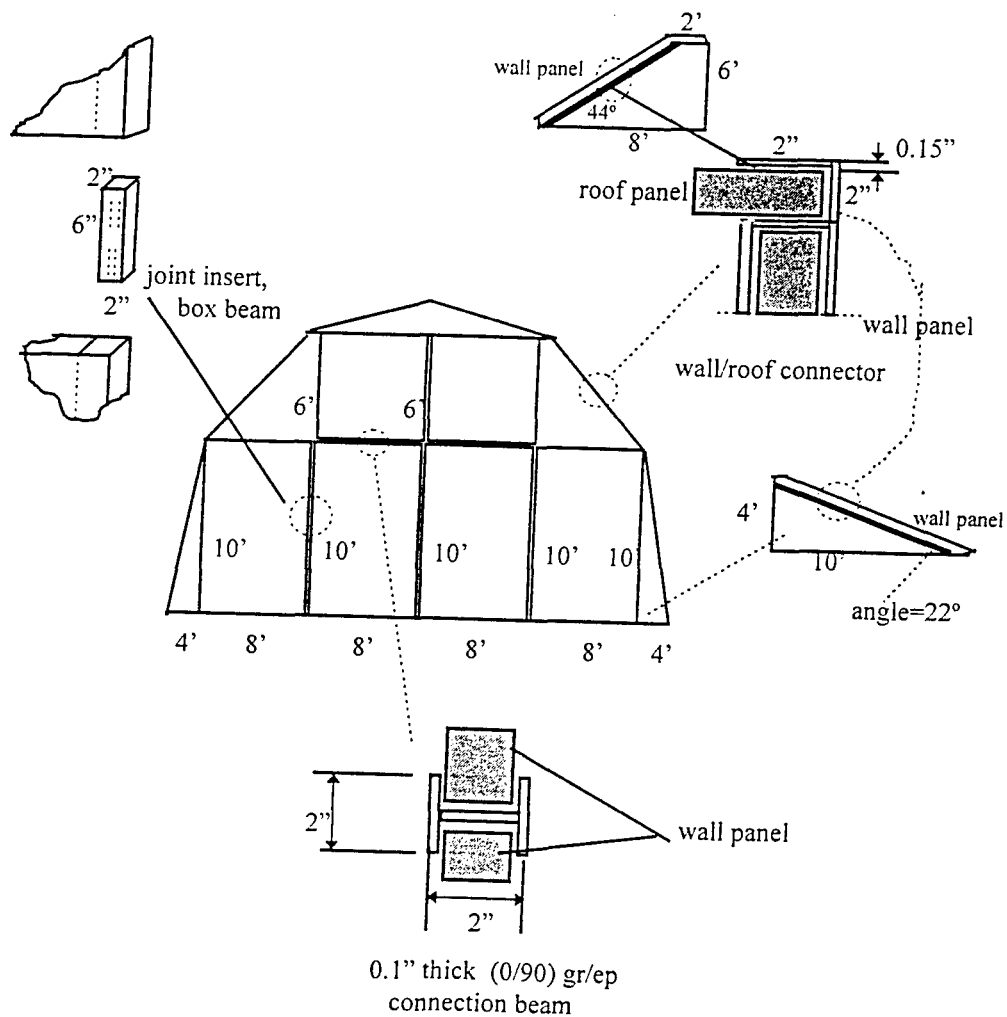
Figure 14. GPS shelter, wall assembly

The critical buckling load for a sandwich panel under in-plane load (ref. 1. & 5) is,

$P_{cr} = (\pi^2.D_f^*) / [b^2(1+\pi^2.D_f^*/b^2.D_s^*)]$. Using the data from before critical load is calculate, $P_{cr} = 27,600$ lb., which larger than applied in-plane load on a wall panel, $P_{cr} = 27,600$ lb.$> P=21,600$ lb., which means the wall is O.K. and will not buckle. The wall assembly is shown in figure 14. There are four types of panels in the wall assembly of GPS shelter, two types of rectangular shape 8'x10', and 8'x6', and the other two, one of trapezoid and one of triangular shape as shown in the figure 14. The information on the facing and core thickness and material are the same for all of them as was calculated before.

The modules are connected through GPS unit connectors, which are (0/90) graphite/ep composite I-beams, of 10' long. The sectional area is shown in figure 15.
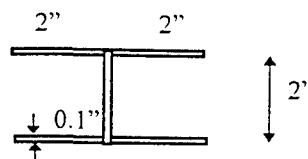
Figure 15. GPS unit connector, composite I-beam

Weight Analysis

A Microsoft excel spread sheet was developed to do the weight analysis for the GPS shelter. The analysis are taken into four steps, the weight of the panels (roof. floor. wall), the weight of the interfacial components, the weight of the subassemblies, and finally the weight of shelter module. The weight of the GPS module is 2,000 lb., and of the whole shelter is 12,947 lb. The weight of ACH module is 2,632 lb and of the composite ACH shelter is 13,742 lb. The results are shown on the following pages. There was no weight data available on the similar metallic shelters such as Harvest Falcon Assets for comparison purposes. However, the data on the weight of MERWS shelters was found on the literature, which was compared with the composite GPS shelter in table 1.

Table 1. Weight Comparison of Composite GPS with the Metallic MERWS Shelters

	Composite GPS	Metallic MERWS
Dimension	20'x40'x48'	19'x 8' x 53'
Volume (cu. ft)	34,800	8,056
Sq ft Area	1,920	1,007
Weight (lb.)	12.947	13,500
Weight/Volume (lb/ft^3)	0.38	1.68
Weight/sq ft Area (lb/ft^2)	6.75	13.41

A shown in table 1, per volume, the composite GPS shelter weighs only one quarter weight of the metallic MERWS shelters. This figure goes up to 50% for per sq ft area which still is a significant weight saving.

Roof Panel with Integrated Box Beam

Components	Number	Description	Width (in)	Cross Sectional Area (in2)	Density (pci)	Length (in)	Volume (in3)	Weight (ib)
Inner Skin	1	0.05"thick Qausi Isotrop. E-Glass/Epoxy	120	6	0.073	92	552	40.296
Core	1	2"thick Web-Stiffened PU Foam	120	240	0.0015	92	22080	33.2304
Outer Skin	1	0.05"thick Quasi-Isotrop. E-Glass/Epoxy	120	6	0.073	92	552	40.296
Integ. Box Beam	2	0.125"thick(0/90)Gr./Epoxy rectang. pipe		1	0.057	120	240	27.36
Total Weight								**141.1824**

Weight of Roof Panel

19%   28%

29%   24%

- ☐ Inner Skin
- ■ Core
- ☐ Outer Skin
- ☐ Integ. Box Beam
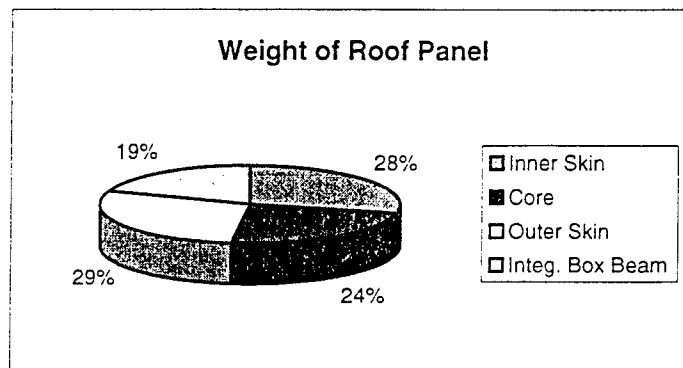
Floor Panel with Integrated Box Beam

Components	Number	Description	Width (in)	Cross Sectional Area (in2)	Density (pci)	Length (in)	Volume (in3)	Weight (lb)
Inner Skin	1	0.06"thick Quasi Isotrop. E-Glass/Epoxy	120	7.2	0.073	92	662.4	48.3552
Core	1	2"thick Web-Stiffened PU Foam	120	240	0.0015	92	22080	33.2304
Outer Skin	1	0.06"thick Quasi Isotrop. E-Glass/Epoxy	120	7.2	0.073	92	662.4	48.3552
Box Beam	2	0.125"thick(0/90)Gr./Epoxy rectang. pipe		1	0.057	120	240	27.36
Total Weight								**157.3008**

Weight of Floor Panel

Box Beam 17%

Inner Skin 31%

Outer Skin 31%

Core 21%

- ☐ Inner Skin
- ■ Core
- ☐ Outer Skin
- ☐ Box Beam
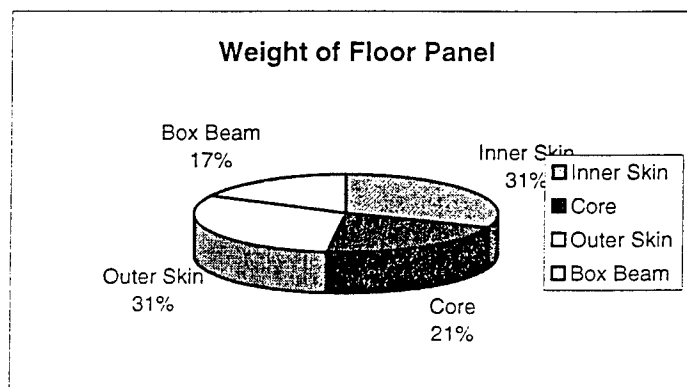
Weight of Composite GPS Wall

Component	Description	Number	Cross Sec Area (in2)	Density (pci)	Length (in)	Volume (in3)	Weight (lb)
Wall Panel Type I							
Panel w/o Stiffener	The Same as Roof Panel						143.9184
Box Stiffner	.1"x2"x2" (0/90) E-Gl/Ep	4	0.8	0.073	120	96	28.032
Total Weight of Panel I							**171.9504**
Wall Panel Type II							
Inner Skin	0.05"x92" Q/ISO E-Gl/EP	1	4.6	0.073	72	331.2	24.1776
Core	2"x92"Q/ISO E-Gl/EP	1	184	0.0015	72	13248	19.872
Outer Skin	0.05"x92" Q/ISO E-Gl/Ep	1	4.6	0.073	72	331.2	24.1776
Integ. Box Beam	0.125"x2"x2" (0/90)Gr/Ep	2	1	0.057	72	72	8.208
Wall Box Stiffener	.1"x2"x2" (0/90) E-Gl/Ep	4	0.8	0.073	72	57.6	16.8192
Total Weight of Panel II							**93.2544**
Wall Panel Type III							
Inner Skin	0.05"Thick Q/ISO E-Gl/EP	1	4320	0.073	0.05	216	15.768
Core	2"Thick Q/ISO E-Gl/EP	1	4320	0.0015	2	8640	12.96
Outer Skin	0.05"Thick Q/ISO E-Gl/EP	1	4320	0.073	0.05	216	15.768
Integ. Channel Section	0.125"x2" (0/90) Gr/Ep	1	0.75	0.057	120	90	5.13
Wall Box Stiffener	.1"x2"x2" (0/90) E-Gl/Ep	4	0.8	0.073	36	28.8	8.4096
Total Weight of Panel III							**58.0356**
Wall Panel Type IV							
Inner Skin	0.05"Thick Q/ISO E-Gl/EP	1	2880	0.073	0.05	144	10.512
Core	2"Thick Q/ISO E-Gl/EP	1	2880	0.0015	2	5760	8.64
Outer Skin	0.05"Thick Q/ISO E-Gl/EP	1	2880	0.073	0.05	144	10.512
Integ. Channel Beam	0.125"x2" (0/90) Gr/Ep	1	0.75	0.057	130	97.5	5.5575
Wall Box Stiffener	.1"x2"x2" (0/90) E-Gl/Ep	4	0.8	0.073	24	19.2	5.6064
Total Weight of Panel IV							**40.8279**
Wall Assem.Component							
Joint Insert, Box Column	(0/90) Gr/Ep Strip	4	4	0.057	6	24	5.472
Wall Connection Beam	0.1"Th.(0/90)Gr/Ep I-Beam		0.75	0.057	1488	1116	63.612
Wall-to-Floor Con. Beam	0.1"Th.(0/90)Gr/Ep I-Beam		1.4	0.057	480	672	38.304
Total Weight of Panel IV							**107.388**

Weight of Compos. GPS Component

Component	Description	Cross Sectional Area (in2)	Density (pci)	Length (in)	Volume(in3)	Weight (lb)
Joint Insert, Roof Box Beam	(0/90)XAS Gr/Ep Angle Strip	4	0.0553	12	48	2.6544
Roof Lateral Connect. Beam	0.1" Thick(0/90)Gr/Ep I-Beam	0.8	0.0553	96	76.8	4.24704
Joint Insert, Floor Box Beam	(0/90)XAS Gr/Ep Angle Strip	4	0.0553	12	48	2.6544
Floor Lateral Connect. Beam	0.1" Thick(0/90)Gr/Ep I-Beam	0.6	0.0553	96	57.6	3.18528
Roof -to-Floor Connect. Beam	0.1" Thick(0/90)Gr/Ep ForkBeam	0.6	0.0553	96	57.6	3.18528
Subfloor Joinst	(+45/-45/O2)Gr/Ep I-Section	3	0.0553	96	288	15.9264
GPS Units Connect. Beam	0.1" Thick(0/90)Gr/Ep I-Beam	1	0.0553	120	120	6.636

Weight of Compos. GPS SubAssem.

Assembly Component	Unit Weight (lb)	Number of Req'd Units	Total Weight (lb)
Roof Assembly			
Roof Panel	141.1824	6	847.0944
Joint Insert, Roof Box Beam	2.6544	14	37.1616
Roof Lateral Connect. Beam	4.24704	5	21.2352
Roof Assembly	**Total Weight:**		**905.4912**
Floor Assembly			
Floor Panel	157.3008	4	629.2032
Joint Insert, Floor Box Beam	2.6544	6	15.9264
Floor Lateral Connect. Beam	3.18528	5	15.9264
Subfloor Joist	15.9264	7	111.4848
Floor Assembly	**Total Weight:**		**772.5408**
Wall Assembly			
Wall Panel Type I	171.9504	4	687.8016
Wall Panel Type II	93.2544	2	186.5088
Wall Panel Type III	58.0356	2	116.0712
Wall Panel Type IV	40.8279	4	163.3116
Wall Link/Fixture Component			107.388
Wall Assembly	**Total Weight:**		**1261.0812**

Weight of Compos.GPS Shelter

SubAssembly	Unit Weight (lb)	Number of SubAssemb. Req'd per GPS Shelter	Total Weight (lb)
Roof Assembly	905.49	6	5432.94
Floor Assembly	772.54	6	4635.24
Floor/Roof Connect. Beam	3.18528	12	38.22336
GPS Units Connection Beam	6.36	50	318
Weight of GPS Module	**1999.21528**		
Wall Assembly	1261.0812	2	2522.1624
Weight of GPS Shelter			**12946.56576**

Weight of Composite GPS Shelter SubAssemblies

19%, 0%, 2%, 43%, 36%

- ☐ Roof Assembly
- ■ Floor Assembly
- ☐ Floor/Roof Connect. Beam
- ☐ GPS Units Connection Beam
- ■ Wall Assembly
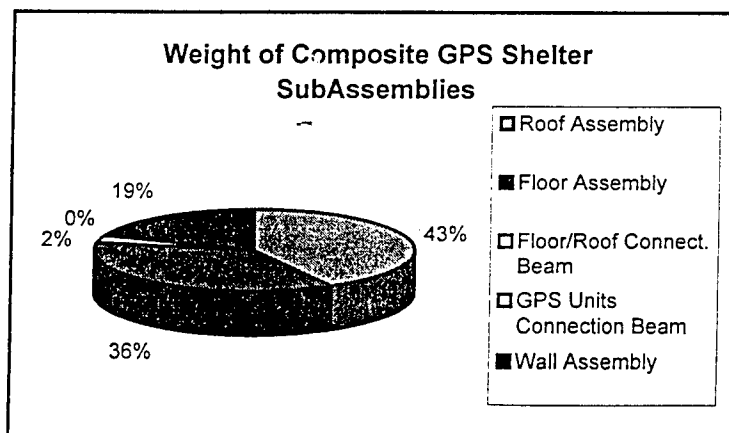
Weight of Compos. ACH SubAssem.

Assembly Component	Unit Weight (lb)	Number of Req'd Units	Total Weight (lb)
Roof Assembly			
Roof Panel	141.1824	8	1129.4592
Joint Insert, Roof Box Beam	2.6544	18	47.7792
Roof Lateral Connect. Beam	4.24704	7	29.72928
Roof Assembly	**Total Weight:**		**1206.96768**
Floor Assembly			
Floor Panel	157.3008	5	786.504
Joint Insert, Floor Box Beam	2.6544	8	21.2352
Floor Lateral Connect. Beam	3.18528	8	25.48224
Subfloor Joist	15.9264	11	175.1904
Floor Assembly	**Total Weight:**		**1008.41184**

Weight of Compos.ACH Shelter

SubAssembly	Unit Weight (lb)	Number of SubAssemb. Req'd per ACH Shelter	Total Weight (lb)
Roof Assembly	1207	6	7242
Floor Assembly	1008	6	6048
Floor/Roof Connect. Beam	3.18528	12	38.22336
GPS Units Connection Beam	6.36	65	413.4
Weight of ACH Module	**2631.58528**		

Weight of ACH Shelter	**13741.62336**

Weight of Composite GPS Shelter SubAssemblies

- □ Roof Assembly
- ■ Floor Assembly
- □ Floor/Roof Connect. Beam
- □ GPS Units Connection Beam
- ■

3% 0% 0%
44%
53%
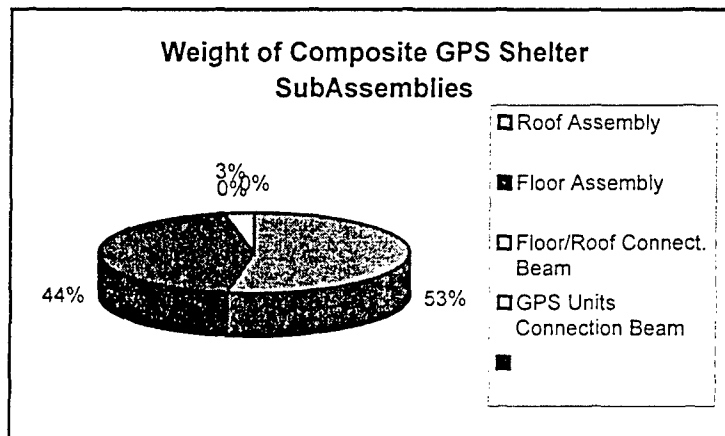
Conclusion

The proposed sandwich panels all have a web-stiffened core made of Polyurethane foam with a thickness of 2". The webs are made of (+45/-45) E-glass/epoxy composite plies with a thickness of 0.0625" and spaced at 1.5" on centers. The designed roof and wall panels have a facing of 0.05" thickness and made of quasi-isotropic (0/90/+45/-45) E-glass/epoxy composite laminate. The calculated floor panels have a different facing thickness of 0.06". The integrated box beams, all the interfacial components such as different type connection beams, and joint inserts are made of (0/90) graphite/epoxy composites. For detail information on their geometry refer to the appropriate section of the report. The subfloor joists are made of (+45/-45) graphite/epoxy composite. The weight analysis showed that the proposed composite shelter has much lower relative weight compare to metallic MERWS shelters.

Further study on the numerical stress analysis of the composite shelter for even more optimum solution is a necessity. Thermal shock as well as creep analysis for the composite shelters are required to be implemented. The foundation as well as the wall analysis for the ACH shelters are required to be considered. The results of the weight analysis is required to be compared with other group metallic shelters such as Harvest Falcon Assets to estimate the amount of saving. The optimal manufacturing process for the fabrication of composite GPS and ACH shelters should be investigated and the candidate methods be identified. Finally cost study for these candidate methods should be made for an economical evaluation. These further studies are also recommended by my focal point through granting an extension program proposal.

Reference

1. Howard, G. Allen, Analysis & Design of Structural Sandwich Panels, Perg. Press, 1969.
2. CAF 316-92-I-B, Operational Requirement Document (ORD) for a new Family of Portable Shelters, Headquarters Air Combat Command, 21 June 1994.
3. Walter E. BecKer, U.S. Sandwich Panel, Manufacture Marketing Guide. Mobay Chemical Co., 1969.
4. MIL-ST-907B, Military Standard Engineering & Design Criteria for Shelters, 9 Sept. 1985.
5. John Hartsock, Design of Foam-Filled Structures, Technomic Publication Inc., 1968.
6. Stephen W. TSAI, Composite Design, 4th edition, Think Composite, 1994.
7. Harvest Falcon Assets, training course textbook, Air Force.
8. Rudolph S. Zilard, Theory & Analysis of plates, Prentice Hall, 1974.

RE-ENGINEER AND RE-MANUFACTURE AIRCRAFT STRUCTURAL COMPONENTS USING LASER SCANNING

Dr. Joe G. Chow
Associate Professor
Department of Industrial Engineering

Florida International University
Miami, FL 33199

Final Report for:
Summer Research Program
Air Logistics Center

August 1997

RE-ENGINEER AND RE-MANUFACTURE AIRCRAFT STRUCTURAL COMPONENTS USING LASER SCANNING

Dr. Joe G. Chow
Associate Professor
Department of Industrial Engineering
Florida International University

Abstract

In the summer of 1995, a feasibility study was initiated to study the applicability of laser scanning to aircraft structural component manufacturing. A sample part, F-15's leading edge rib, was selected and scanned at Laser Design Inc. (LDI) and Sharnoa Corp, respectively.. Following scanning, a CAD model was created at LDI and an aluminum part was machined at Sharnoa. The results from this study indicated that laser scanning had matured to a stage that they could possibly capture and reproduce intricate surface details typically present in aircraft structural components. However, the data produced did not show enough accuracy for the Warner Robins Air Logistics Center (WR-ALC) to implement this new technology on their production floor.

A follow-up study was conducted at WR-ALC in the summer of 1996. To broaden the scope of the study, two more sample parts, an F-15's canopy fitting and a C-141's forward latch fitting, were also included. A research plan that could demonstrate the accuracy and efficiency of laser scanning in duplicating existing aircraft structural components was devised. Because of time constraints, only a few tasks in the research plan were completed in the 1996 summer. The principal investigator, committed to complete this research work, has been using the manufacturing facilities at his university to continue this project since the completion of the 1996 summer program.

During the summer of 1997, entire effort was again devoted to the laser scanning project. In this report, the work performed during the 1997 summer research program and the plan to complete the entire project by the end of this year are presented.

RE-ENGINEER AND RE-MANUFACTURE AIRCRAFT STRUCTURAL COMPONENTS USING LASER SCANNING

Joe G. Chow

1. Introduction

WR-ALC is one of the five Air Logistics Centers of the United States Air Force. One of its primary responsibilities is to manufacture structural parts for military aircrafts. Machining structural components at WR-ALC typically starts with a 3D CAD surface model for the part. Though all the parts that need to be machined are accompanied by 2D part drawings and, in most cases, by a copy of the actual part, a 3D CAD model of the part is rarely provided. The CAD model, therefore, needs to be created manually from the 2D drawings or by direct measurement of the part. This modeling process is very labor intensive and time consuming, and it can only be performed by a highly skilled modeler.

Once a 3D CAD model is created, subsequent activities can benefit greatly. The model can be used during process planning to determine the machining operations required and to select the machine tools, cutting tools and machining parameters for each operation. The NC toolpaths for each machining operation can be generated using the CAD model and commercial CAM software. The fixtures for holding the part during each machining operation are then designed and manufactured. Fixturing elements can be displayed along with the CAD model to verify fixtures designed and toolpaths generated. The CAD model also facilitates part inspection after it is built.

Currently, at WR-ALC, all the activities in the manufacturing process are performed manually. Due to the extent of human involvement, it requires on average six weeks to manufacture parts of medium complexity. In the case of high complexity parts, the turnaround time may be several months long. To increase combat readiness and sustainability, it is essential for WR-ALC to reduce the part turnaround time and costs. This requires automation and elimination of some of the steps in the manufacturing process.

In the summer of 1995, a feasibility study was initiated to study the applicability of laser scanning to aircraft structural component manufacturing [1]. Two different approaches, re-engineering and re-manufacturing, have been studied. For re-engineering approach, an F-15's leading edge rib, was digitized using a laser scanner at Laser Design Inc. (LDI), Minneapolis MN and the point cloud data obtained was used to create a 3D CAD model. A comparison between the reconstructed model and the part's blue prints indicated that the errors of the reconstructed model were on the order of 0.005". Due to time and equipment constraints, toolpathing and machining were not carried out for this approach.

For the re-manufacturing approach, the model creation step was skipped and the NC toolpaths were generated directly from the point cloud data. Eliminating model creation from the manufacturing process allowed a part to be duplicated quickly. To verify the accuracy and efficiency of this approach, a leading edge rib was scanned and machined at Sharnoa, Plymouth MI. The duplicated part was then compared to the original part using a dial caliper. The comparisons indicated that, except for a few locations, the dimensional discrepancies between the two were within the desired tolerances (0.010"). This study also demonstrated that laser scanning can reduce not only the part turnaround time, but also the skill levels required to reproduce the part.

While both approaches showed great promise, additional studies would be required before it can be considered for implementation on the WR-ALC's production floor. During the summer of 1996, the feasibility study was continued with an aim to obtain more concrete and conclusive results [2]. To prove this technology is applicable to WR-ALC, it is highly desirable to replicate their parts using the re-engineering and/or the re-manufacturing approach and show that the replicated parts are within tolerances. A detailed research plan to complete this project to WR-ALC's satisfaction is summarized as follows.

2. Objective and Research Plan

The objective of this research is to characterize the accuracy and efficiency of laser scanning systems for aircraft structural component manufacturing. To achieve this objective, three sample parts, a leading edge rib, a forward latch fitting, and a canopy fitting, that capture most of geometric features in the aircraft structural components will be scanned and reproduced. Both

re-engineering and re-manufacturing approaches will be studied: first re-engineering and then re-manufacturing.

For the re-engineering approach, all the sample parts will be digitized by a laser scanner and the scan data for each part will be converted into a CAD surface model by commercial reverse-engineering software. A part will be machined using toolpaths created from each reconstructed surface model. A comparison between the duplicated and the original part will reveal the accuracy of the entire approach.

Upon the completion of the re-engineering study, scan data for the sample parts will be used to generate toolpaths without creating a CAD model first. Machining will be performed and the duplicated parts will be compared to the original parts. Also, by comparing the parts produced by the re-manufacturing approach to the ones made by the re-engineering approach, one can determine the effect of model creation on part accuracy and quality.

The results will be summarized and stored in a table, which displays the accuracy and consistency of the laser scanning systems for various geometric features. Results will also include records of time utilization for each phase, such as setup, scanning, editing, model reconstruction, and toolpathing from the scan data. Estimates for the savings achieved through new improved processes over the traditional method will also be made.

3. Work Performed Before and During 1997 Summer Research Program

3.1 Work Done Before 1997 Summer

Since the completion of 1996 summer research program, the principal investigator and his students have continued to work on the laser scanning project at his home institution, FIU. The following tasks have been completed prior to his 1997 summer research program.

- **Concluded that Hymarc scanner is the best laser scanner for the WR-ALC's parts.**

After more than one year's on-site scanning and data evaluation at FIU, the principal investigator reached a conclusion that Hymarc scanner is the most accurate laser scanner for the WR-ALC's parts. In addition to high accuracy, it is also the fastest scanner, capable of scanning 10,000 points per second.

- **Purchased a copy of Imageware's Surfacer to transform the scan data into a surface model**

For the re-engineering approach, the scan data need to be converted into a surface model. In the past, we were using the Strim software to convert scan data into surface models. Since Strim was not designed specifically for the reverse engineering applications, we found it was very difficult to use. To improve the efficiency of surfacing task, we purchased a copy of Surfacer (the most popular reverse engineering software on the market) and started to use it to convert some of our scan data into surface models

- **Machined a leading edge rib out of machinable wax**

For the last two summers' studies, all tasks for this project were performed at scanner vendors' sites because no proper equipment and tools were available on the Base. The laser scanner vendors were usually willing to scan the WR-ALC's sample parts free of charge, but they usually did not have resources and expertise in surfacing and machining aircraft structural components. As a result, the laser scanning project initiated in 1995 has not been performed to WR-ALC's satisfaction because of difficulties in producing replicated parts.

To overcome this problem, it was decided that machining facilities at the principal investigator's university should be used for this project. Earlier this year, we machined a leading edge rib using a Sharpe milling machine at FIU. The toolpaths used were generated from the surface model created by the LDI during the 1995 summer study. The duplicated part has a strong resemblance to the original, but was not very accurate. In spite of this deficiency, this machining exercise demonstrates that aircraft structural components that require only a 3-axis machine tool can be machined at FIU.

3.2 Work performed during 1997 summer tour

The principal investigator accomplished the following tasks during his 1997 summer research program at Robins Air Force Base:

- **Created a CAD model and generated toolpaths for the forward latching fitting**

The forward latch fitting' scan data produced by the Kreon laser scanner were converted into a surface model using the Surfacer software and the toolpaths were created using the Smartcam software.

- **Modified toolpaths for the leading edge rib.**

As an attempt to improve part accuracy, the toolpaths and the process plan created for the leading edge rib earlier were modified.

- **Duplicated a leading edge rib and a forward latch fitting using the re-engineering approach.**

Using the toolpaths the principal investigator generated, his graduate students machined a leading edge rib and a forward latch fitting at FIU. This time, they paid more attention to the part setup, fixturing, and machining order, which were considered as the primary sources of error during last machining session. The machined wax parts are shown in Figures 1 and 2, respectively.

- **Compared the duplicated part with the original part, and/or with the CAD model.**

To determine the accuracy of the duplicated parts, the replicated leading edge rib was compared with the original part using a dial caliper and also with its CAD model. The comparisons are displayed in Table 1.
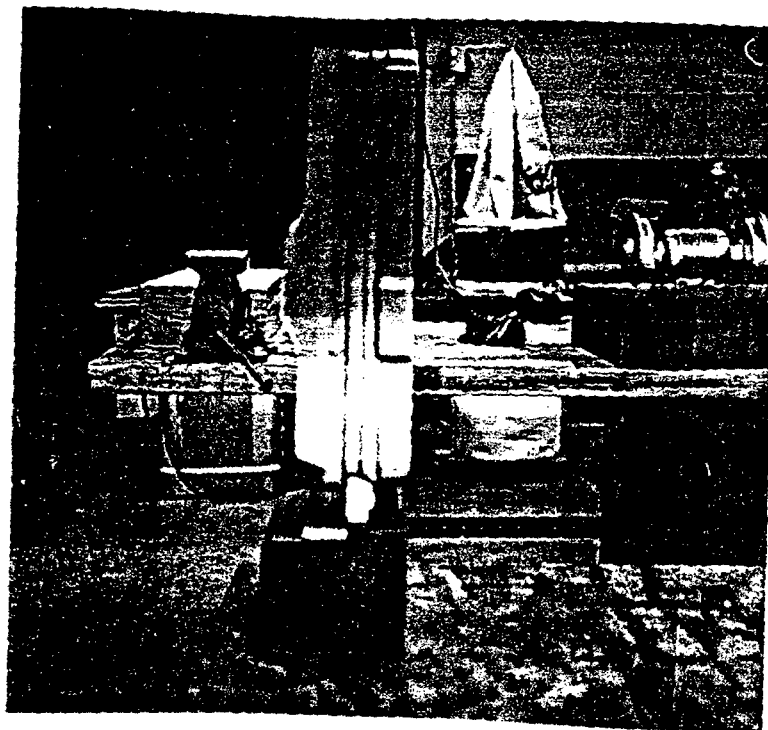
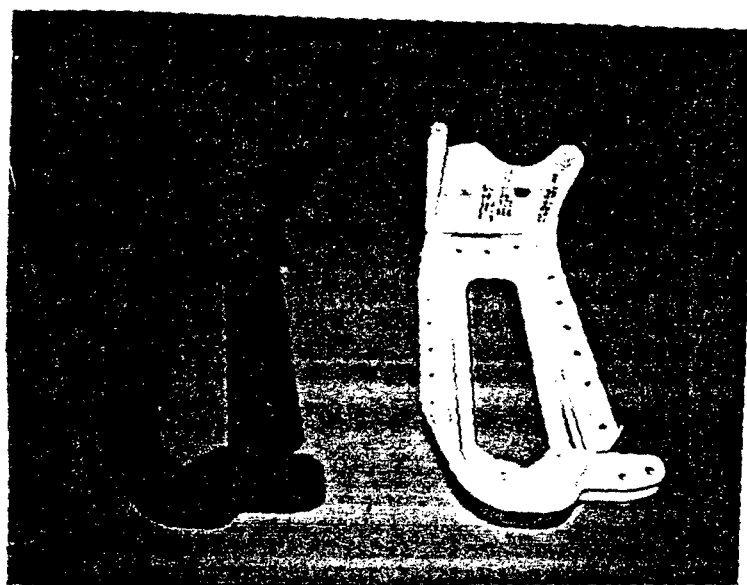Figure 1. A duplicated leading edge rib

Figure 2. A duplicated forward latch fitting (the one on left)

Table 1. A comparison between duplicated part, original part and its CAD model for the leading edge rib.

LEADING EDGE RIB (Units= inch)

Data No.	Duplicated Part (DP)	Original Sample (OS)	CAD Model (CM)	DP-OS	DP-CM	OS-CM
1	0.363	0.365	0.364	-0.002	-0.001	0.001
2	0.101	0.148	0.148	-0.047	-0.047	0
3	0.16	0.133	0.148	0.027	0.012	-0.015
4	0.9	0.875	0.909	0.025	-0.009	-0.034
5	3.23	3.225	3.289	0.005	-0.059	-0.064
6	0.057	0.088	0.064	-0.031	-0.007	0.024
7	0.231	0.24	0.22	-0.009	0.011	0.02
8	1.59	1.615	1.624	-0.025	-0.034	-0.009
9	0.465	0.481	0.49	-0.016	-0.025	-0.009
10	0.128	0.134	0.119	-0.006	0.009	0.015
11	2.78	2.79	2.785	-0.01	-0.005	0.005
12	0.88	0.87	0.85	0.01	0.03	0.02
13	0.125	0.136	0.134	-0.011	-0.009	0.002
14	0.125	0.137	0.129	-0.012	-0.004	0.008
15	0.088	0.081	0.083	0.007	0.005	-0.002
16	1.005	0.983	0.993	0.022	0.012	-0.01
17	0.089	0.084	0.078	0.005	0.011	0.006
18	0.265	0.27	0.272	-0.005	-0.007	-0.002
19	0.985	0.985	0.987	0	-0.002	-0.002
20	0.462	0.48	0.492	-0.018	-0.03	-0.012
21	0.387	0.385	0.391	0.002	-0.004	-0.006
22	1.806	1.814	1.786	-0.008	0.02	0.028
23	0.051	0.088	0.071	-0.037	-0.02	0.017

The locations where these dimensions were taken are shown in Figures 3-4.

To facilitate data interpretation, the discrepancies between the duplicated part, the original and the CAD model are also displayed in graphs, as shown in Figures 5-7.
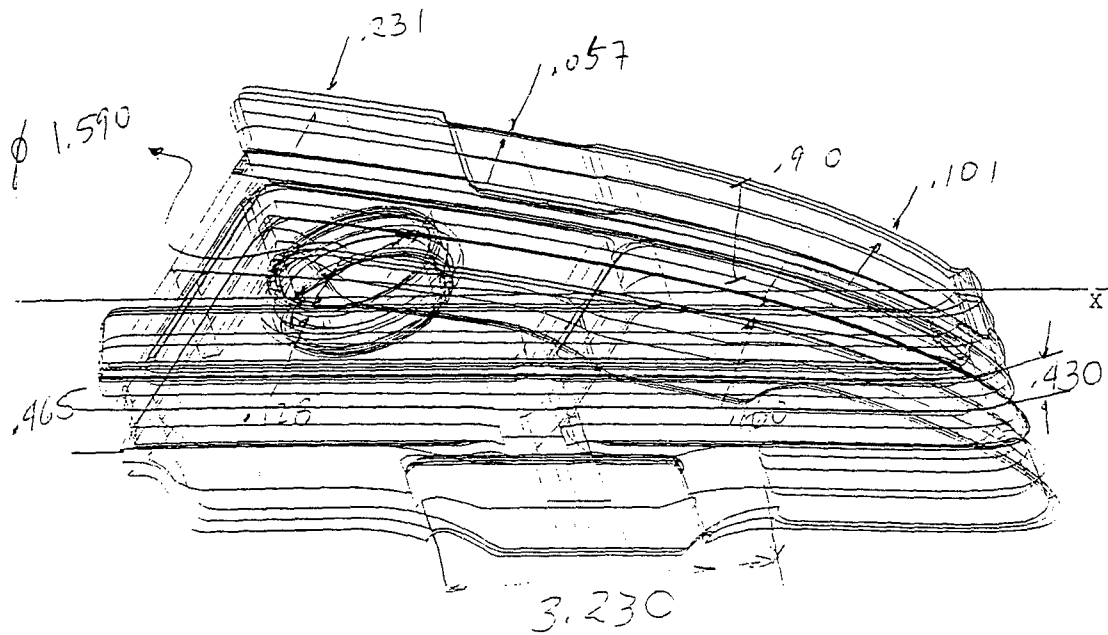
(a)
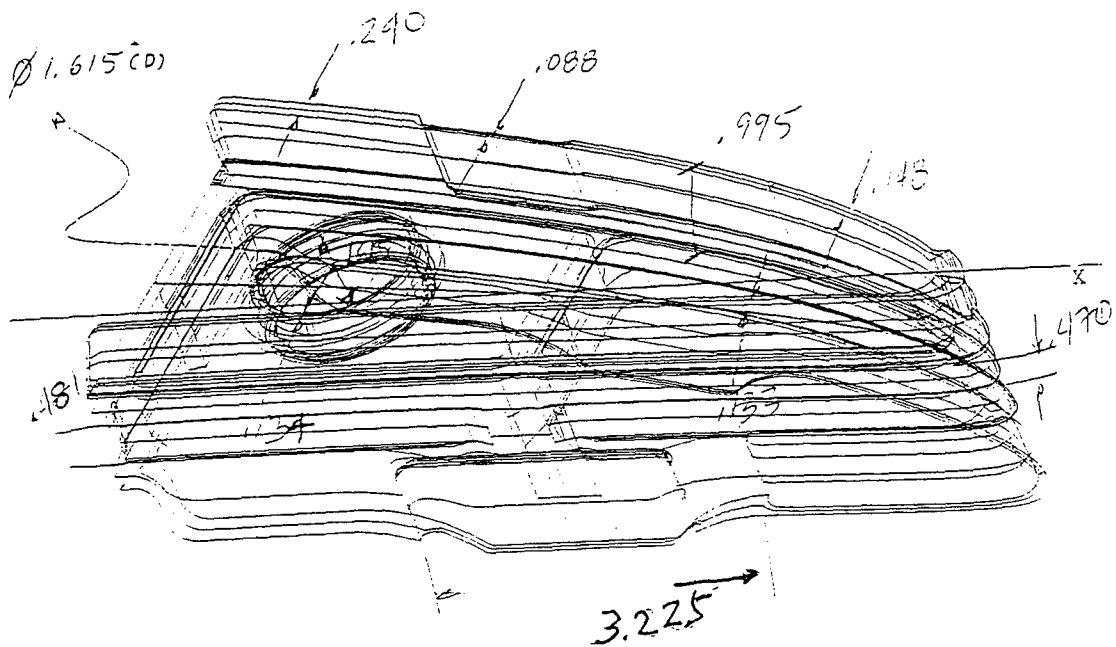
(b)

Figure 3. Locations where the measurements, Data 1-10 in Table 1, were taken (a) wax part and (b) original sample

(a)

(b)

Figure 4. Locations where the measurements, Data 11-23 in Table 1, were taken (a) wax part and (b) original sample
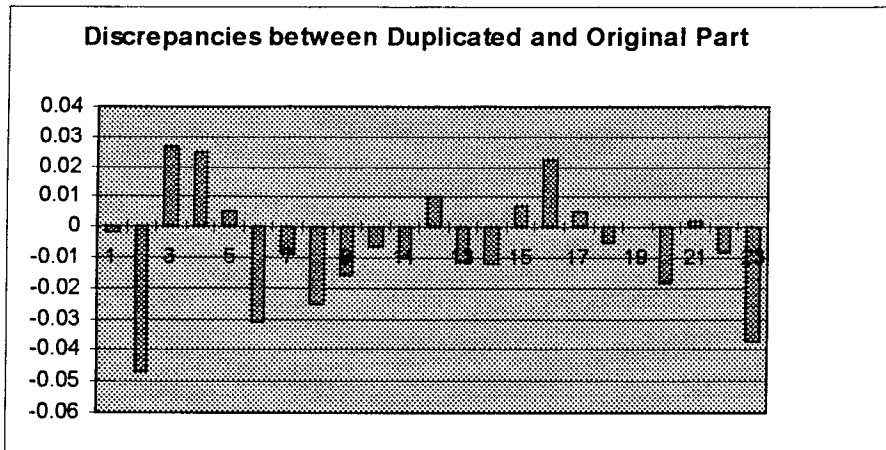
Figure 5. A comparison between the duplicated leading edge rib and the original leading edge rib
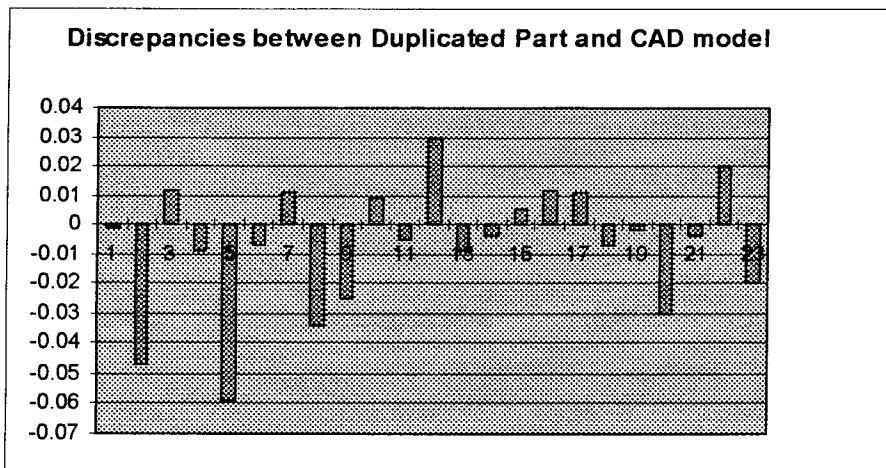

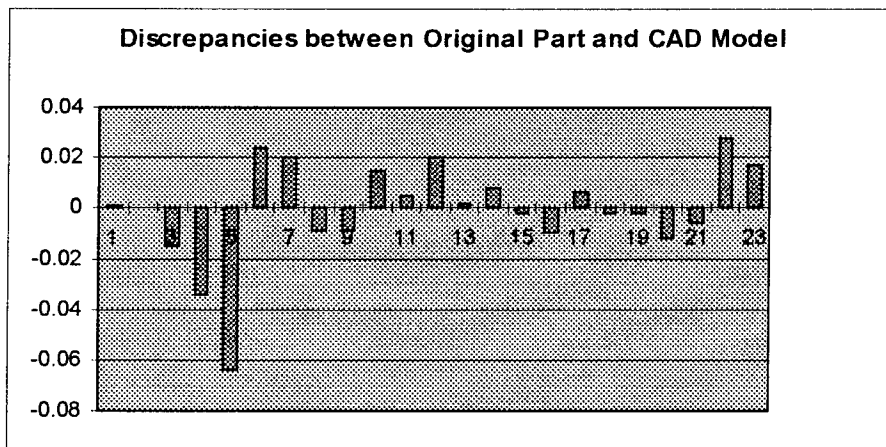Figure 6. A comparison between the duplicated leading edge rib and the CAD model

Figure 7. A comparison between the duplicated leading edge rib and the CAD model.

For the forward latch fitting, only the duplicated part was compared with the original. The results are shown in Table 2.

FORWARD LATCH FITTING (Units=Inch)

Data No.	Duplicated Part (DP)	Original Sample (OS)	DP-OS
1	1.839	1.743	0.096
2	0.095	0.105	-0.01
3	1.483	1.5	-0.017
4	0.562	0.5	0.062
5	0.151	0.215	-0.064
6	0.096	0.114	-0.018
7	0.553	0.448	0.105
8	0.12	0.172	-0.052
9	0.245	0.162	0.083
10	0.562	0.5	0.062
11	0.347	0.26	0.087
12	0.149	0.151	-0.002
13	0.399	0.5	-0.101
14	1.187	1.187	0
15	1.732	1.733	-0.001
16	1.43	1.434	-0.004
17	0.297	0.299	-0.002

Table 2. A comparison between the duplicated and original forward latch fitting

The difference between theses two sets of dimensions are plotted in Figure 8.
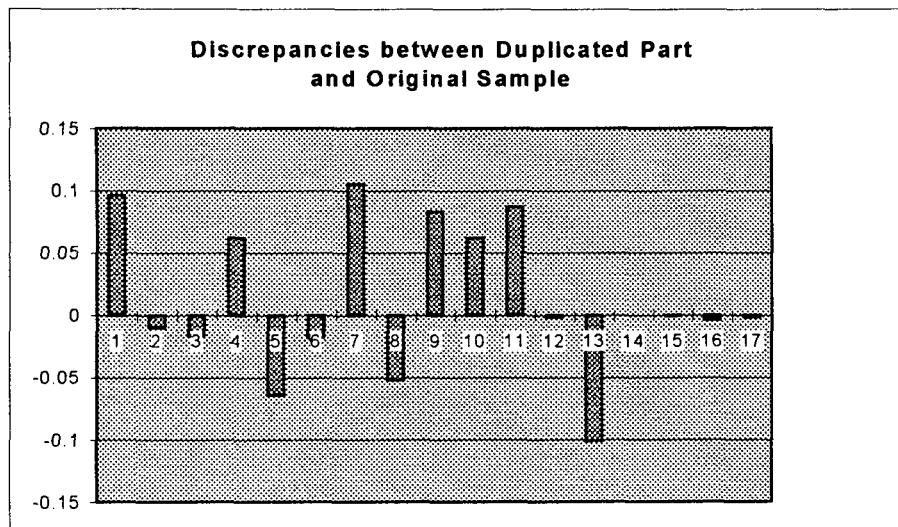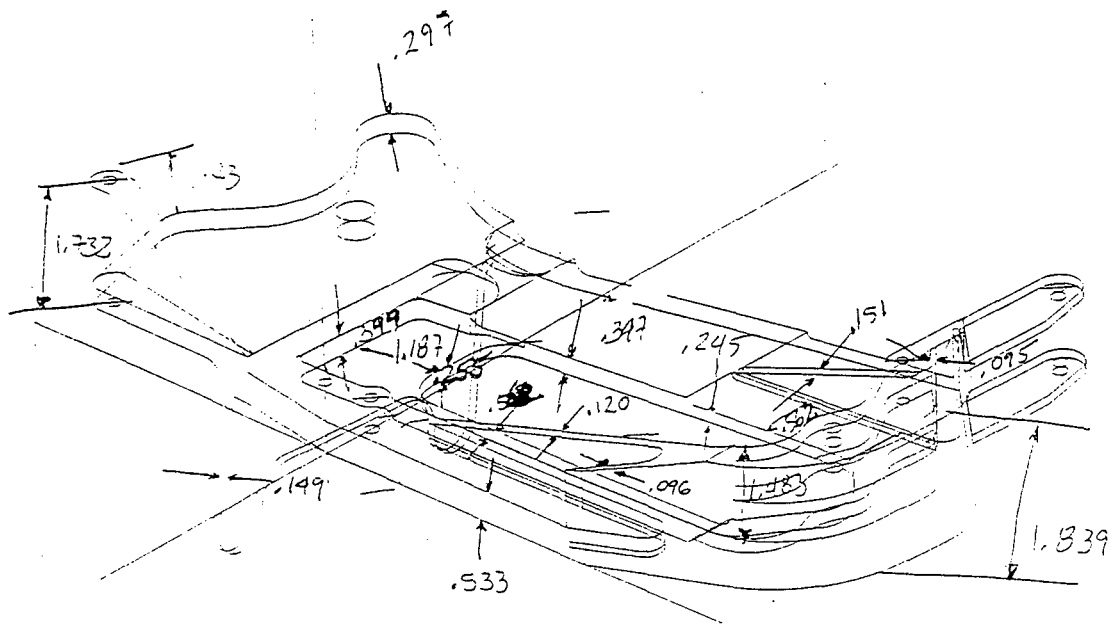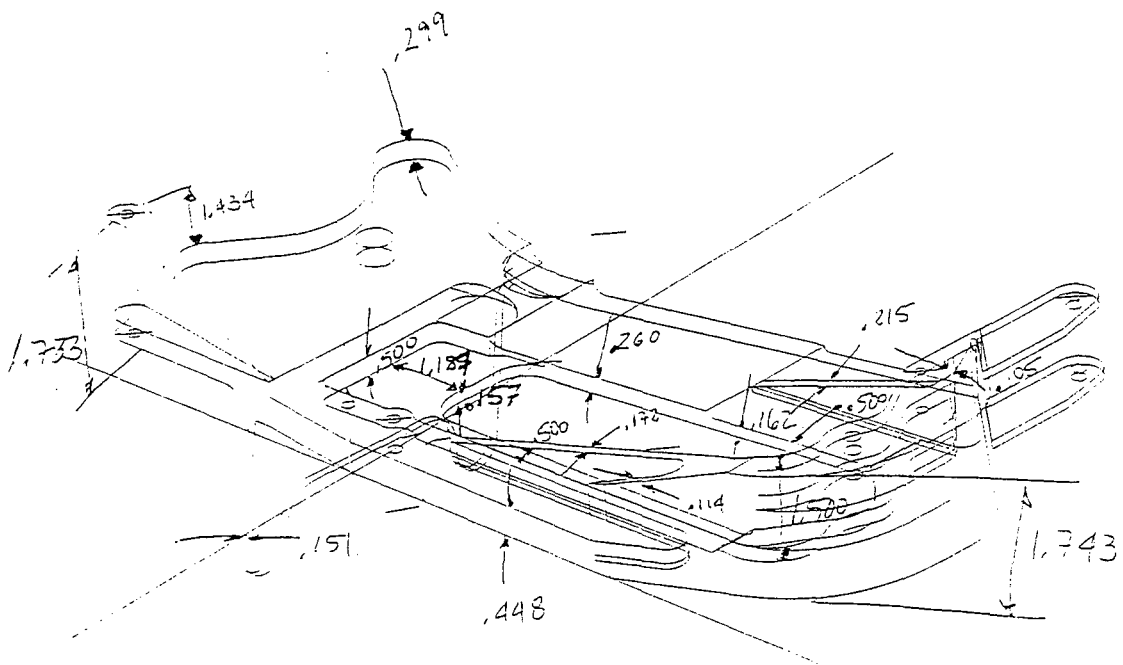

Figure 8. The discrepancies between the original and duplicated forward latch fitting

The locations where these dimensions were taken are shown in Figure 9.

(a)

(b)

Figure 9. Locations where the measurements (Data 1-17 in Table 2) were taken (a) wax part and (b) original sample

- **Evaluated white light triangulation laser digitization tool**

Although the principal investigator reached a conclusion earlier on that the Hymarc scanner is the best laser scanner on the market. He still kept his eyes open for a newer and better digitization tool. Recently, the principal investigator learned that the laser digitization method based on the white light triangulation is also very accurate and fast. He contacted Steinbichler Corp. (the vendor for this technology) and asked them to digitize a sample part to evaluate the effectiveness of their system. Steinbichler digitized a leading edge rib and returned the data to the principal investigator for evaluation and surfacing.

A careful examination of Steinbichler point data using Surfacer revealed that their raw data are extremely dense and fairly clean. The high density can be attributed to the large amount of data (400,000 points) taken during each measurement. In spite of huge amount of data, each measurement took only 2 minutes to complete. Another important characteristic is that their data were collected with both the part and the sensor remaining stationary, as opposed to a moving part or sensor in a traditional laser scanner. They claimed this feature would increase the accuracy of their digitization process. The comparison between the point data and the actual part, as shown in Table 3 and Figure 10 respectively, indicates that Steinbichler data are indeed very accurate (≥ 0.003").

Table 3. A comparison between the Steinbichler point data and original leading edge rib

Leading Edge Rib (units=inch)

Data No.	Locations	Point Data (Steinbichler) (PD)	Original Sample (OS)	PD-OS
1	A	1.181	1.182	-0.001
2	B	4.176	4.177	-0.001
3	C	0.601	0.598	0.003
4	D	1.623	1.62	0.003
5	E	0.245	0.245	0

The locations, A-E, where Data 1-5 were taken can be found in Figures 3-4.

Figure 10. A comparison between Steinbichler point data and original sample for the leding edge rib

Discrepancies between Steinbichler Point Data and Original Sample for Leading Edge Rib
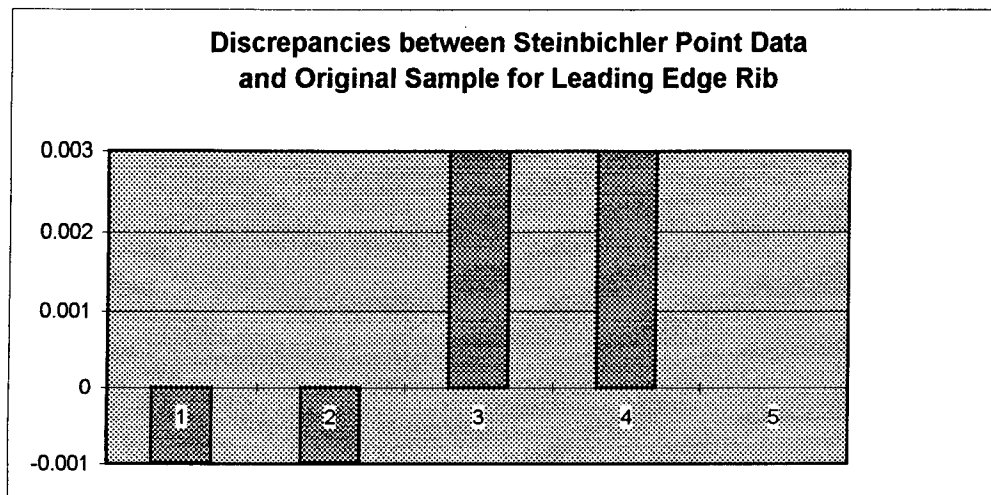
- **Devised a plan to duplicate a canopy fitting**

To duplicate a canopy fitting by machining, a 5-axis machine tool will be required. Since FIU does not have a 5-axis machine tool, machining has to be performed at an outside machine shop. Earlier this summer, the principal investigator contacted Mr. Terry Peters in Atlanta GA for this purpose. Mr. Peters, a representative for the CAD/CAM Cimatron software, showed great interest in this project and agreed to assist us in machining a canopy fitting. The plan to machine a canopy fitting consists of the following steps : (1) surface the scan data by a Cimatron application engineer, (2) generate toolpaths by Mr. Peters, and (3) perform machining at Georgia Tech using their 5-axis machine tool.

- **Use a CMM to inspect the parts**

Two methods are available to check the dimensions of two identical parts: using a manual device, such as dial caliper, or a coordinate measuring machine (CMM). With a CMM,

measurements can be taken from all the part surfaces, especially those complex free form surfaces. Furthermore, the CMM data are more accurate and repeatable, compared to manual measuring tools. FIU recently acquired a high-precision Brown and Sharpe Xcel 7107 UHA CMM. During his 1997 summer tour at WR-ALC, the principal investigator attended a training class for this machine and passed the class materials to his students at FIU. Currently, his students are learning how to use this machine and will write CMM programs to compare duplicated parts with the original ones when they become familiar with this equipment.

4. Discussions

In general, the dimensions of the duplicated parts have poor agreements with those of the originals. Compared to the allowed tolerances ($\pm$ 0.010"), approximately 50% of the measured dimensions in the duplicated parts are out of this range. For the forward latch fitting, the dimensions that are out of tolerance tend to be excessive (as much as 0.100"). The main reason for this large error is due to poor quality of the sensor used. Among all of the laser scanners evaluated, the Kreon laser scanner is probably the least accurate one. Kreon data was still used to produce a replicated part because the objective was to become familiar with the surfacing and machining processes for this part. A better scan data for this part will be obtained in the future. At that time, a forward latch fitting can be duplicated very quickly using the procedure already established.

For the leading edge rib, the discrepancies, though not as large as those of the forward latch fitting, are still very significant (as much as 0.045"). The scanning for this part was done at LDI using one of their laser scanners two years ago. Following scanning, the scan data were surfaced at their site by the Camand software. Since the principal investigator did not get involved in the scanning and surfacing processes for this part, the accuracy of the model creation process could not be determined. Again, the main reason to machine this part was to establish an evaluation procedure and apply this procedure to a better scan data later, such as Steinbichler point data.

THIS
PAGE
IS
MISSING
IN
ORIGINAL
DOCUMENT

20-18

*Note: This part will be scanned at WR-ALC using the EOIS sensor at TINPE.

The Re-Manufacturing Approach:

	Leading edge rib (Steinbichler data)	Forward latch fitting
Toolpathing	November	November
Machining	December	December
Measuring	December	December

6. Conclusions

The procedure of duplicating aircraft structural components using a laser scanner consists of several major tasks: scanning, toolpathing, machining, and/or modeling. All these tasks require extensive experience, specific software and equipment. During the summers of 1995 and 1996, the essential tools required to conduct this research were not readily available and the personnel involved in the project did not have required skills. This summer, because of more software and equipment options and financial supports from Wright Lab and WR-ALC, we could perform most of the tasks ourselves and gain more control over each process. This advantage will not only allow the principal investigator to conduct this research to WR-ALC's satisfaction and also complete this project by the end of this year.

7. Bibliography

1. J. Chow and Ramesh M. C., "A Feasibility Study for Re-Manufacturing Aircraft Structural Components Using Laser Scanning And X-Ray Computed Tomography", Technical Report to Air Force Office of Scientific Research, September 1995.

2. J. Chow "Re-Engineering and Re-Manufacturing Aircraft Structural Components Using Laser Scanning" Technical Report to Air Force Office of Scientific Research, September 1996.