

AD-A 211 136

**SOME PAGES OF THIS DOCUMENT ARE BELOW
LEGIBILITY STANDARDS REQUIRED FOR
PREPARATION OF SATISFACTORY MICROFICHE.
THESE PAGES ARE INCLUDED. IF YOU REQUIRE
THESE PAGES, PLEASE REQUEST THE LOAN OF
THE ORIGINAL FROM:**

National Defence Headquarters
Ottawa, Canada, K1A 0K2
Attn: Customer Services Centre/DSIS/CRAD.

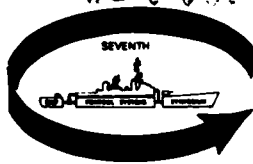
VOL. IV

**BEST
AVAILABLE COPY**

**QUELQUES PAGES DU PRÉSENT DOCUMENT SONT
INFÉRIEURES AUX NORMES DE LISIBILITÉ REQUISES
POUR LA RÉALISATION DE MICROFICHES SATISFAISANTES.
LES PAGES EN QUESTION SONT INCLUSES. SI VOUS
AVEZ BESOIN DE CES PAGES, DEMANDEZ QUE VOUS
SOIT PRÊTÉ L'ORIGINAL AUPRÈS DE:**

Quartier général de la Défense nationale
Ottawa, Canada, K1A 0K2
Compétence: Services aux clients SISD/CR Dev.

Vol 4 Proceedings of the Seventh Ship Control Systems Symposium 1984



Proceedings

Seventh ⁷¹⁴³ BR 94199
Ship Control System
Symposium

24-27 September 1984
Bath, United Kingdom

Volume 4 UNLIMITED



VOL 1 BR 94796
VOL 2 BR 94199
VOL 3 BR 94199
VOL 4 BR 94199
VOL 5 BR 94200
Indices & Supp. BR 94201

UNLIMITED



UNLIMITED

1984
Bath, UK



Printed at the University of Bath

UNLI

BR 94199

PUBLICATION INFORMATION

These papers were printed just as received from the authors in order to ensure their availability for the Symposium.

Statements and opinions contained therein are those of the authors and are not to be construed as official or reflecting the views of the United Kingdom Ministry of Defence.

Authors have given permission on behalf of themselves and/or their employers for the United Kingdom Ministry of Defence to publish their paper(s) in the Proceedings of the Seventh Ship Control Systems Symposium. No material contained within these Proceedings may be reproduced without the prior permission of the publishers.

Requests for further information regarding the Proceedings and the Symposium should be addressed to:

B BLOCK, MINISTRY OF DEFENCE (PE), FOXHILL, BATH, BA1 5AB, UK:
ATTENTION: ADMG.

Volume 4

Contents

Software for Simulation and Optimisation Applied to Ship Control Systems: Dr J van Amerongen, Dr P P J van der Bosch, Delft University of Technology, Netherlands	1
The Management of Software with a Long Life Span: Mr I J Sinclair, YARD Ltd, UK	11
Software Guidelines for Development of Digital Control Systems: Dr J R Ware, ORI Inc, USA	23
Statistical Analysis of Autopilot Performance: Mr J B D Rush, Dr D R Broome, University College London, UK	37
Estimating the Wave Spectrum and the Roll Angle and its Derivates with an Extended Kalman Filter: Ir ing P W Hoogenraad, Royal Netherlands Navy, Dr J van Amerongen, Delft University of Technology, Netherlands Prof Ir Hans van Nauta Lemke, Delfte University of Technology, Netherlands	73
Comparison of Steering Control Algorithms for Optimized Autopilots: Dr G J Thaler, Lt V Garcia USN Naval Postgraduate School Monterey, USA	87
Data Dependent Statistical Analysis for Ship Optimal Control: Mr K Ohtsu, Tokyo University of Mercantile Marine, Japan	101
A Microprocessor Based Pitch Indicating System for the DDH 280 Class: R K Santo, Department of National Defence, Canada	see vol 5
Design of a Man-Machine Interface for Supervisory and Manual Ship Systems: J Vermeulen, Institute for Perception, Netherlands	see vol 5
The Machinery Control Room for Onboard Training: Mr G M Primrose, Mr K M Glen, YARD Ltd, UK	117
Flexible Controls for a Flexible Ship: Mr C T Marwood, HSDE Ltd, UK, Mr T Munk, Naval Material Command, Denmark	131
Surface Effect Ship Ride Control System: Mr J D Adams, Maritime Dynamics Inc, USA, Mr A W Ernest, Ernest Engineering, USA, Mr W N White, Naval Sea Systems Command, USA	147
Integrated Damage Control System: Mr D W Geer, United Technologies Norden Systems, USA	177

SOFTWARE FOR SIMULATION AND OPTIMISATION APPLIED TO SHIP CONTROL SYSTEMS

by P.P.J. van den Bosch and J. van Amerongen,
Control Laboratory
Department of Electrical Engineering
Delft University of Technology
P.O. box 5031, 2600 GA Delft, The Netherlands

ABSTRACT

A comparison is given between mathematically-oriented methods and methods for control systems design based on simulation and optimisation. It turns out that the last approach can be more flexible and powerful, if supported by appropriated software. A description is given of the interactive simulation program PSI that allows the application of proposed approach.

1. INTRODUCTION

In this paper the use of simulation for the analysis and design of control systems is discussed. This approach is compared with other, mathematically-oriented analysis and design methods. The latter methods make use of, for example, the linearity of the system so that mathematical solution techniques are available. Then a quantitative judgement may exist on the range of validity of the results, for both this and other comparable systems.

In general, systems do not satisfy assumptions such as linearity, order information, etc., so that either a mathematical approach cannot be applied or a simplified, linear model has to be used.

An analysis or design approach based on simulation and optimisation is much more flexible and can deal with nearly any system description, linear or nonlinear, continuous or discrete or any mixture of differential, difference, algebraic or logical equations. This flexibility has to be paid for by extra calculation time. Suitable software has to be available in order to use such an approach.

In this paper the proposed simulation and optimisation approach for analysis and design of control systems will be compared with other, mathematically-oriented methods. Requirements will be derived for the software and a simulation program, PSI, that has been designed to realise nearly all these requirements will be described.

2. SYSTEM ANALYSIS

Identification methods are generally based on an analysis of the input and output signals of the system that has to be identified. Estimates of the parameters of a model, whose structure and order are determined in advance, can be calculated. Depending on the identification method a priori information, based on noise

characteristics, can also be included in the algorithm in order to obtain better estimates. Let us briefly consider the Least Squares Method (LS).

The LS-method assumes the process to be described by the linear, discrete transfer function $H(z) = A(z)/B(z)$. The LS method calculates the parameters a_i and b_i ($A(z) = a_0 + a_1z + \dots$, $B(z) = b_0 + b_1z + \dots$) model very fast by solving a set of n linear equations with n unknown parameters a and b . LS estimates only unbiased parameters a and b , if the noise $n(k)$ satisfies several conditions, for example, $n(k)$ has to be coloured noise, arising from white noise filtered by means of $B(z)$. If this condition is not met, the noise characteristics have to be estimated too. This can be achieved by, for example, the Extended Matrix Estimator. This extension introduces an iterative solution procedure which increases the calculation time considerably. When a-priori information about the structure or the parameters of the process is available, it is difficult or even impossible to use this knowledge.

Another approach to system identification is the use of simulation and optimisation which enables simulation and optimisation in some a-priori information of the system to be taken into account (for example, some knowledge about the internal structure, some known parameters, in some cases the shape or even the exact values of a non-linearity, etc.). This a priori information can be obtained from additional measurements or from the understanding of the physical laws that describe the system under consideration. This additional a-priori information can be very useful in finding an appropriate model of the system.

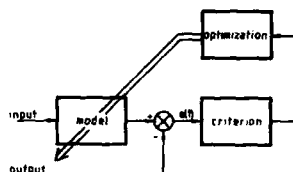


Fig. 1. Identification via simulation and optimisation.

By means of simulation and optimisation we can calculate the "best-fit" model of the system. Linearity assumptions are no longer necessary. Such an approach is illustrated in Fig. 1. A criterion is defined, based on the error between the output of the process and the output of the (adjustable) model. Both are excited by the same input signal. The output of the model is obtained by using simulation techniques. Therefore, the model may be described by continuous parts, discrete parts, non-linear or logical elements or any combination of these. Then an optimisation algorithm is able to find optimal model parameters of a (non-)linear model with a user-defined structure and a user-defined criterion. For example, if we know in advance that the system under consideration has two time constants (and thus two real poles) this knowledge can be used in the identification scheme of Fig. 1, but not in the LS method.

This flexibility is achieved at the expense of calculation time. Optimisation is inherently a non-linear iterative procedure. Each iteration requires a complete simulation run so that, more calculation time is needed than with the LS method. For system analysis this is not a real limitation. However, real-time identification for adaptive control poses strict limitations on the calculation-time requirements, so that the proposed identification method can not always be used. For interactive use of this facility, the number of parameters has to be limited to a maximum of about 5 to 10.

3. SYSTEM DESIGN

There are many ways to design a control system such that it satisfies pre-defined design requirements. In general, some control structure has to be implemented to improve the system behavior. In designing such a controller we can use several graphical representations of the system in order to study its dynamic behaviour and to find ways to define controllers such that the system behaviour will improve. Linear single-input single-output systems can be designed by using the Bode or Nyquist diagrams or root loci. Linear multivariable systems can be treated by graphic design methods such as the Inverse Nyquist Array method, the Characteristic Locus Design method, etc. For non-linear systems the describing function method and the circle criterion are available, although they are rather conservative. These graphic design methods offer much qualitative and quantitative information about the system behaviour. Nevertheless, if the system is complex much experience and knowledge is required to be able to design an appropriate controller which satisfies the design requirements.

Another approach to designing systems is to formulate the design problem in terms of an optimisation problem: formulate a criterion, parameters of a controller that have to be optimised and constraints. The criterion has to satisfy two requirements, namely it has to express the design objectives and has to be easy to calculate. In choosing a mathematically-oriented criterion the optimisation process can be quite fast, but the link with the design objectives, such as overshoot, rise time, damping etc., may be weak or even non-existent. For example, the linear optimal state feedback matrix, according to the quadratic functional J:

$$J = \int_0^{\infty} (x^T Q x + u^T R u) dt$$

taking into account the state x and the input u , can be easily found by solving a Riccati equation. There also exist fast algorithms for pole placement, etc.

Output feedback, instead of state feedback, complicates the optimisation considerably. Then, relatively simple expressions exist to calculate both the functional J and its gradient with respect to the coefficients of the feedback matrix. Hirzinger (1975) has proposed an usable output-feedback configuration for multivariable systems. His dynamic controller has both feedback and feedforward. The design requirements placed on dynamic behaviour and decoupling are expressed in a parallel reference model, which causes an unconstrained optimisation problem to arise with functional J as criterion. The dimension of the state now becomes the sum of the dimension of the

states of the original system, of the controller and of the parallel model. The value of J and its gradient are calculated by solving Lyapunov equations.

Methods which allow other, non-quadratic criteria to be used, yield more flexibility at the expense of additional computational effort (Zakian, 1979; Mayne and Polak, 1982).

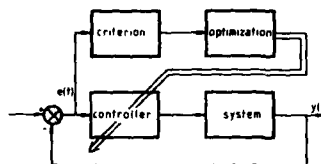


Fig. 2. System design using simulation and optimisation.

Even more flexible is the approach based on simulation and optimisation (Fig. 2). Simulation techniques are used to simulate the controller and the process and to calculate the error signal e . A criterion is defined, based on this error signal and/or the output, which can be optimised with respect to the parameters of the controller. So, any (non-)linear system and any controller configuration can be used with any criterion. Finite or infinite-dimensional constraints can be included, via penalty functions, in the criterion. Even the combination of a discrete controller which controls a (non-)linear continuous system offers no problems.

Van den Bosch (1982) has illustrated that calculation-time requirements of the simulation and optimisation approach are comparable with solving the linear output feedback problem.

From the point of view of accuracy, simulation suffers less from numerical errors. Especially for high-order systems, numerical solution methods for Riccati or Lyapunov equations may lead to inaccurate or erroneous results.

Therefore, it may be concluded that, even when an analysis or design is otherwise possible, it may still be profitable to use simulation and optimisation due to its inherent flexibility.

4. REQUIREMENTS FOR SIMULATION PROGRAMS

In this section we will concentrate on the requirements to be put on the simulation facility. Both digital and hybrid computers can be used. Due to the many advantages of the digital computer when compared with a hybrid one (price, availability, size of the problem, etc.) we shall focus our attention on the requirements for simulation programs intended for digital computers.

Integration Methods

Simulation programs calculate the solution of sets of linear or non-linear differential and/or difference equations. Digital computers calculate a variable only as a sequence of values at discrete time intervals, determined by the integration interval. Therefore, the continuous integrator has to be approximated. The accuracy with which this approximation can be realised determines the accuracy of the simulation and depends both on the integration method and the integration interval. With a small integration interval and a complex, higher-order integration method more accurate results can be expected than with a larger integration interval and a simpler integration method. But, both a small integration interval and a higher-order integration method increase calculation time. So, a compromise is possible between calculation time and accuracy. In using fixed-step integration methods, the second and fourth-order Runge Kutta integration methods are widely accepted.

Algebraic Loops

A second problem arises in solving a parallel defined system with a sequentially-oriented digital computer. This problem can be solved by using a proper sorting procedure, except when there is an algebraic loop (an equation in which a variable is an algebraic function of its own value), for example, $x = \sin(x) + y$. It is always advisable to avoid algebraic loops. If they cannot be avoided they have to be solved with the aid of time-consuming, iterative algorithms, which can be used not only for the solution of algebraic loops, but also for the solution of any general, non-linear algebraic equation.

Multi-Run Facilities

There is an important distinction between preprocessor-like programs (in general batch-oriented) and interpreter-like programs (in general interactive) for simulation purposes. The former allows statements of a high-level programming language to be included in the simulation model description. Therefore, these programs can be made as flexible as, for example, a Fortran program. Interpreter-like programs lack this facility, so that special measures have to be taken to realise, for example, multi-run facilities such as optimisation, comparison of variables between different runs, initial, dynamic and terminal calculations, etc.

User Interface

If a simulation program has very attractive mathematical aids and perfect multi-run facilities, it may still be inferior with respect to control system design when the interaction between the program and the user is not accepted by the user. This interaction is determined by a number of factors, but especially by the communication between the user and the program and the presentation of graphic information (Van den Bosch and Bruijn (1979)). Only an interactive program can support a designer-oriented environment. Either a command language or a question/answer approach can take care of the communication between the program and the user. A command language offers much more flexibility but lacks the guidance available in a question/answer approach.

In designing control systems, graphic representations of the system behaviour are of paramount importance. Although numbers are much more exact, design considerations mainly deal with graphic representations of a system. For example, linear optimal state feedback, linear optimal output feedback or pole placement are well-established, mathematically-oriented methods for control system design. However, whether or not such a design meets the ultimate design requirements cannot be judged by looking at only the value of the criterion or at the feedback matrix. In general, only time (or frequency) responses offer enough information to make a judgement of the ultimate system behaviour possible.

So, a graphics display, which is very fast, or a plotter is almost unavoidable when analyzing or designing systems.

5. THE PROGRAM PSI

Up to now facilities enabling a simulation program to be used for interactive system analysis and system design have been discussed. At the Laboratory for Control Engineering an Interactive Simulation Program (PSI), (Van den Bosch (1979,1984)), has been designed and realised. This interpreter-like, block-oriented simulation program offers, for example, the following facilities:

Facilities

- About 90 commands support the user to realise his design objectives.
- Five numerical integration methods are available, namely four fixed-step methods (Euler, Adams Bashfort 2, Runge Kutta 2 and 4) and one variable-step-size methods (Runge Kutta 4).
- Solution of algebraic equations is realised by a fast Newton-Raphson algorithm. If this procedure fails, a more reliable, although slower, optimisation algorithm is used.
- Optimisation with scaling and constraints is supported. In PSI the user can define the output of an arbitrary block as the criterion and up to eight arbitrary parameters of the simulation model as parameters of the optimisation. The parameters that offer the smallest value of the criterion will be accepted as the solution of the optimisation procedure. Pattern search (Hooke and Jeeves (1961), equipped with a premature stop of the simulation run has been selected as non-linear optimisation procedure, due to its robustness and lack of a line-minimisation procedure.

Although Pattern Search adjusts its search step size according to the "shape" of the criterion, improvement of the speed of convergence can be obtained by using scaling. Scaling can make each parameter about equally important for the optimisation algorithm. Not only is scaling supported by PSI, but constraints are also allowed. Each parameter may have an upper and a lower limit. The optimisation algorithm will only search for an optimum in the feasible region of the parameter space.

- Multi-run facilities are available. For example, run-control blocks, comparison of signals between several runs, etc. With the aid of storage variables PSI offers the initial-dynamic-terminal facility of CSMP III. At the end of a simulation run, this run can be continued without resetting the time and the initial conditions.

- Extensive tests on all user-supplied information is implemented. Each error is indicated by a meaningful error message of which there are about 60.

- About 50 powerful block types are available, among which integrators (limited, mode-controlled, resettable), continuous and discrete PI- and PD-controllers, Pulse-Width Modulation, etc. Fortran programming in a non-interactive mode is required to define new block types. The user only needs to write a subroutine in which the output is defined as a function of the input(s) and parameter(s), compile it and after a link step, his block is available.

- There are memories to store signals during a simulation run. These signals can be studied after the simulation run, can be saved on disk or can be used as inputs for future runs. These signals can be redrawn on the screen, as responses or as phase trajectories, after which a cursor, controlled by keyboard commands, can "walk along" these responses. The numerical values appear directly on the screen, so that overshoot, rise time or accuracy can be determined both quantitatively and qualitatively.

- Symbolic block names can be used. Instead of numbers each block or variable can be assigned a user-selected name of up to eight characters. So blocks can get meaningful names like PRESSURE, SPEED or OUTPUT instead of abstract numbers like block 13, 91 or 512, etc.

This section has described a number of facilities which makes programs, such as PSI, highly suited to the analysis and design of control systems. PSI is able to solve (non-linear) differential, difference, algebraic and logical, Boolean equations or any mixture of them. Moreover, an attractive and powerful interaction is realised between the user and the program.

Limitations

Still, PSI has limitations. These limitations arise from the minimum hardware requirements and the bounded facilities supported by PSI. As a consequence of the choice of Fortran as programming language, the many tests on input data and the extensive error messages, PSI has become quite large (approximately 200k). Therefore, PSI has to run in a 56k bytes computer in an overlay environment, so that a fast background memory, for example floppy disk or hard disk, is a prerequisite. The minimum hardware configuration consists of a processor with 56k bytes memory, a terminal and a floppy-disk unit. A display is not necessary but very valuable. Implementations of PSI run on mainframes (Cyber), Superminis (VAX), minicomputers (PDP 11/x, HP 1000) and 16-bit microcomputers (TULIP I, IBM-PC, both with MS-DOS).

Like most other interactive, block-oriented simulation programs, PSI does not support special facilities to solve partial differential equations, stiff systems and polynomial or matrix equations. These programs deal with single-valued variables, and consequently not with vectors and matrices. The solution of the Ricatti equation of a second-order system is possible, but the solution of this equation of higher-order systems cannot be obtained easily.

Yet, the designer should be aware of the limitations and pitfalls of simulation and optimisation. Although a design is supposed to be

optimal, it is only optimal for the selected design environment, namely structure of the model or controller, selected criterion, final time, integration interval, selected input signals, etc. Such an optimal design can yield an undesirable control behavior. The designer has to recognize that simulation and optimisation is a design tool, not a decision maker.

6. APPLICATION TO SHIP'S STEERING

The simulation program described in this paper has extensively been used during the design of a 'rudder-roll stabilisation' system (Van Amerongen, Van der Klugt and Pieffers, 1984). The problem is to design a controller for the system of Figure 3.

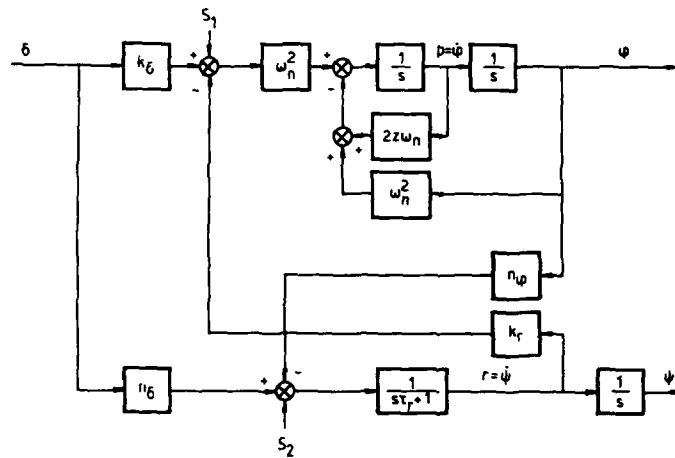


Fig. 3 Simple model to describe the transfer between the rudder and the roll and yaw motions of a ship

Both the heading (ψ) and the roll (ϕ) have to be controlled by one single input, the rudder (δ). Because the rudder angle and the rudder speed are both limited, the process to be controlled is essentially non-linear. Therefore, it is essential that the model of figure 3 be extended with the steering machine dynamics. A simplified block diagram of the steering machine, including both limiters is given in figure 4.

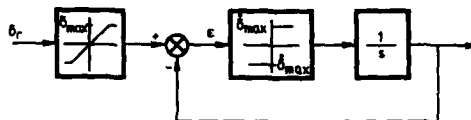


Fig. 4 Simplified block diagram of the steering machine

The whole system, including the steering machine dynamics is of the fifth order, which implies that five feedback loops are needed in order to realise complete state feedback.

The following controller is considered:

$$\delta = K_1\psi + K_2\dot{\psi} + K_3\phi + K_4\dot{\phi} + K_5\ddot{\phi}$$

where $\dot{\psi}$ denotes $d\psi/dt$ and so on.

For the system of figure 3 the feedback gains can be computed with the LQ approach, by solving a Ricatti equation, after definition of a quadratic criterion. In that case the steering machine dynamics have to be approximated, for instance, by a linear first-order transfer function which has to be added to the system of figure 3. However, because of the non-linearities in the steering machine, in practice the controller will only work satisfactorily for small disturbances. For large disturbances the performance will quickly deteriorate, especially because of the limited rudder speed.

Another limitation of the analytical approach is that a quadratic criterion has to be chosen. For the rudder-roll-stabilisation system it is desirable however, to use the criterion:

$$J = 2.\max |\phi| + 5.\max |\psi| \quad \text{for } 0 < t < T$$

where $\max(x)$ denotes the maximum value of x during the considered time interval.

As mentioned before, the design of a controller via the program PSI, by means of simulation and optimisation is just as easy with the 'max-criterion' as it is with a quadratic criterion. Also the non-linearities can easily be taken into account. The program can be used to decide which rudder speeds and rudder limits are allowable to realise the required reduction in a certain sea state. Finally, the program can be used to test the influence of discretisation of the controller, while the model of the ship still describes a continuous system. The results obtained with the design procedure are extensively described by Van Amerongen, Van der Klugt and Pieffers (1984).

CONCLUSIONS

The value of simulation and optimisation for system analysis and system design has been discussed. It appears that many systems can only be studied by using simulation techniques. But even when analytical methods are available, simulation and optimisation have their own unique merits. Yet, it has to be stressed that a user of these facilities should be aware of the potentialities as well as of the limitations and pitfalls of the proposed analysis and design method.

Facilities which allow the use of both simulation and optimisation in an interactive way have to be available. It has been illustrated that interactive simulation programs such as PSI are very well suited to use in interactive analysis and design of control systems.

REFERENCES

Amerongen, J. van, P.G.M. van der Klugt and J.B.M. Pieffers (1984). Model tests and full-scale trials with a rudder-roll stabilisation system. Seventh Ship Control Systems Symposium, Bath, UK, September 1984.

Bosch, P.P.J. van den (1979). PSI-An Extended, Interactive Block-Oriented Simulation Program. Proceedings IFAC Symposium on Computer Aided Design of Control Systems. Zurich (223-228).

Bosch, P.P.J. van den and P.M. Bruijn (1979). Requirements and Use of CAD Programs for Control System Design. Proceedings IFAC Symposium on Computer Aided Design of Control Systems. Zurich (459-464).

Bosch, P.P.J. van den (1982). Interactive System Analysis and System Design Using Simulation and Optimization. Proceedings IFAC Symposium on Computer Aided Design of Multivariable Technological Systems. Pergamon press, Oxford (225-232).

Bosch, P.P.J. van den (1984). Manual of PSI. Laboratory for Control Engineering, Delft University of Technology (100 pages).

Hirzinger, G. (1975). Decoupling Multivariable Systems by Optimal Control Techniques. Int. J. of Control, vol 22, no 2 (157).

Hooke, R and T.A. Jeeves (1961). Direct Search Solution of Numerical and Statistical Problems. JACM, vol 8, (212-229).

Mayne, D.Q., Polak, E and Sangiovanni Vincentelli, A (1982). Computer Aided Design via Optimization: A Review. Automatica, vol 18, no 2 (147-154).

Sirisena, H.R. and Choi (1977). Minimal Order Compensators for Decoupling and Arbitrary Pole-Placement in Linear Multivariable Systems. Int.J. of Control, vol 25, no 5

Zakian, V. (1979). New Formulation for the Method of Inequalities. Proc. IEE, vol 126, no 6 (579-584).

THE MANAGEMENT OF SOFTWARE WITH A LONG LIFE SPAN

by
I.J.Sinclair, YARD LTD.

ABSTRACT

Experience, particularly in the U.S. military field, has highlighted the problems of escalating software support costs in computer based systems. This has resulted in the realisation that software should be developed with the aim of minimising through life support costs rather than initial procurement costs, thus allowing software to be supported over periods of up to 20 years and be transported over successive generations of hardware without the major cost of replacement of the software.

Although the problems of designing and implementing software which is capable of a long life span are becoming better understood, the problems associated with the management of software to achieve this life span are less well understood.

This paper discusses topics such as quality plans, documentation standards and configuration management and describes the methods which may be applied to the management of software to enable the benefits of the initial investment to be realised.

The Management Problem

A project involving computers will rarely involve the development or procurement of software alone. More often a project will be aimed at procuring and installing a complete 'turn-key' system i.e. a system which comprises all the component parts necessary to make it achieve the functional requirements and any other requirements (e.g. satisfying environmental constraints) laid down in the original project specification.

The production of such a 'turn-key' system is likely to involve specialist expertise from many disciplines. For example, the development of a computer-based Ship's Machinery Control system will involve various engineering disciplines. Firstly there will be those who have the knowledge of exactly how the Ships Machinery actually functions and hence what the computer-based system will have to do to control the machinery. Expertise in ergonomics will also be required to ensure that the man-machine interface including displays, panels, keyboards etc, is effective and that the overall manning philosophy works, particularly under damage conditions. These requirements are in addition to the computer expertise necessary to design the computer system and software and to develop and test it. Even here it would be wrong to consider computer expertise as a single discipline. The field of computing is vast and "computer experts" are likely to be experts only in one of a large number of areas. Terms such as hardware design, communications, real-time software, high-level software, low-level software are examples of just a few of the areas in which computer expertise may be categorised.

A single project is therefore going to embrace skills from a wide range of disciplines. However, a single project manager (or project officer or project leader - terminology varies from organisation to organisation) will have overall responsibility for the successful completion of the project. In an ideal world this individual would be a multi-disciplined expert in all relevant technical areas. This would be in addition, of course, to having extensive project management skills. Not surprisingly, in this far from ideal world, such individuals are an extremely scarce commodity. Accordingly, the project responsibility is likely to be placed in the hands of someone with project management skill and, perhaps, some expertise in one or more of the relevant disciplines.

Of all the unfamiliar disciplines that a project manager may find himself obliged to control, software engineering is arguably the one likely to cause the greatest problems. Software engineering is in its infancy relative to most other engineering areas. It can be considered to be little more than 20 years old, and the distinction between good and bad software engineering practice has only begun to be properly established within the last ten years. Also, because of the lack of maturity of the industry, expertise in this area is in short supply and tends to lie with younger people who have been educated in an environment where the existence and importance of computing to the future has been realised. The disadvantage of this expertise lying with youth is that youth tends to lack the broad engineering background and experience that would be of such great value to a multi-disciplined project.

The management problem, therefore, is that the project manager may find himself responsible for controlling an area of which he has little understanding and in which the experts can tend to be rather blinkered to their own problems which they believe, largely through lack of experience in the broader engineering context, to be rather unique.

The Black Art of Software

The view of software as a 'black art', the domain of a new breed of experts, has been enhanced by the belief that software is intangible, that its development cannot be seen in the same way as a piece of machinery with the progressive development, test, and integration of its component parts. This belief is one which is quite ill-founded. A project manager should not accept that he cannot monitor progress on software development and that he must sit back and hope that at the end of the day the contents of the 'black-box' do what they are supposed to, cost what was budgeted for, and are delivered when scheduled.

This paper aims to destroy the 'black art' image and to highlight to those, not necessarily familiar with software, the techniques that can be used to manage a software development that will help to ensure that objectives, not only of meeting initial procurement budgets and schedules but also of keeping through-life support costs down, can be achieved.

The Definition of "Software"

Before discussing the management of software it is necessary first of all to decide upon a definition of software. Like so many other terms appearing in the lexicon of computing jargon it is not well defined and even experts in the field of software engineering may find themselves disagreeing as to the precise definition. NES 620 "Requirements for Software for use with Digital Processors", (Ref 1) defines software to be "instructions and data code for programmable digital processors". Others might take software to be anything associated with a computer system that is represented on paper rather than being a physical object and is therefore "soft" rather than "hard". This would include all documentation associated with the project and not just that associated with "instructions and data code for digital processors".

For the purposes of this paper the NES 620 definition will be used, but, in order to gain a better understanding of the problems associated with the management of software, it is necessary to draw a further distinction between "binary" software and "source" software. Digital processors are effectively sophisticated pieces of electronics capable of performing different functions determined by the set of instructions (or "program") that is loaded into them. These instructions are represented by patterns of binary digits (0 or 1) mapped onto appropriate 2-state electronics. A program as it exists inside a digital processor can be viewed simplistically as a sequence of binary numbers and is said to be in "binary" form.

However, writing a program in zeros and ones is an extremely onerous task and, to avoid the need for this, "assembly" languages and higher level languages (e.g. CORAL66, FORTRAN, ADA) have been developed. These permit programs to be written in ways much more suited to Homo Sapiens; namely using characters and English-language-like words. This has been achieved by providing a means of translating the higher level forms of "source" software into the binary form. Other programs ('software tools') referred to as compilers and assemblers are used to carry out this translation process.

The important distinction to make between binary software and source software is that while the binary software may be the fundamental thing that makes a digital processor operate, it is the source software in which lies the human understanding necessary to carry out modifications or correct faults. It is therefore of much greater importance to control the source software, and the means used to create the binary software from it, than to control the binary software itself.

The Life Span of Software

This paper is aimed primarily at the management of software with a long life span. This begs the question: Why should the life span of a set of coded instructions vary? One can envisage a piece of hardware corroding and its life span being dependent on the corrosion rate of the particular material used in its construction. There appears to be no equivalent for software.

Let us take the example of a computer-based control and surveillance system for a class of ships. If we assume that the lifetime of the ship will be say, 25 years, then the lifetime of the computer system must be at least that. However, there are several reasons why it may not be.

Firstly, at the rate at which digital processor technology is advancing, the computers on which the system is based are likely to become obsolete and unsupported by their manufacturers within perhaps ten years of initial procurement. They will require to be replaced by more up-to-date versions (or possibly completely different makes of computer, due e.g. to political decisions or manufacturer bankruptcy) and, if the software cannot run on these new computers, it will need to be replaced. While the trend in computer hardware is one of steadily falling prices, it is quite the reverse for software development which is manpower intensive. Replacing hardware may not have too great a financial impact, but replacing software is very likely to impact heavily. Therefore, if software is to have a long life span, it must be transportable between different types of computer.

Secondly, it would not be reasonable to assume that all ships in the class are going to have identical equipment to be controlled and monitored. Neither would it be reasonable to assume that the equipment on a particular ship will remain unaltered throughout its lifetime. If the equipment to be controlled or monitored changes, then the computer system must be capable of changing with it. Otherwise it will need to be replaced. Therefore to have a long life span the software must be flexible.

Thirdly, cost considerations come into play. Software may be flexible in terms of ease of introducing modifications or corrections, but if the cost of maintaining the facilities necessary to make this possible, plus the manpower cost of actually implementing the changes is exorbitant, then there is a large problem. It may well prove less expensive to discard the existing software and completely rewrite it. (The early experience of the U.S. Department of Defence's ever-escalating support costs for computer systems highlighted the problems of having to support and maintain a wide range of computer systems, written in different languages. This led to the current trends towards standardisation on languages such as ADA.) To have a long life span software must be cost-effective to maintain and support. Therefore, where computer systems are concerned, the through life costs must be examined, and not simply the initial procurement costs.

If a project manager is to control a project which is aiming to produce a system with a long life span then he must try to ensure that the software contained in the system has the characteristics of flexibility, portability and will be cost-effective to maintain over the intended lifetime of the system. How can this be achieved when the project manager lacks sufficient understanding of software to know what is required?

The Invocation of Standards

The usual approach be adopted by a project manager when controlling a particular discipline with which he is unfamiliar is to track down the recognised standards relevant to that discipline and ensure that it is a requirement that they are adhered to.

There are two problems with this approach. Firstly, software engineering is a young industry which is steadily learning from experience. As a result what constitutes good engineering practice is rapidly evolving and official standards produced tend to quickly lag behind the state of the art.

Secondly, it is one thing to invoke standards but quite another to monitor what is being done in order to gain confidence that the standards are being adhered to. The aim of project management is to ensure that all objectives (cost, timescale, functionality etc.) are met. As anyone with project management experience will know, the key to doing this is to pick up as early as possible any problems in any area of the project. If picked up early there is a good chance that some remedial action can be found.

If simple invocation of standards is not of much assistance, what else can be done to give confidence that a software development will be successful?

Definition of Standards

While it has been pointed out that standards relating to software engineering may rapidly become outdated, it is not suggested that they be abandoned. What is suggested is that relevant standards are not blindly invoked, but rather examined and qualified if necessary before being invoked for a particular project. The key to success in software engineering is not substantially different from other engineering areas. Good and useful standards must be defined and effective quality control must be introduced to ensure that they are complied with.

There is a range of standards associated with the production of software with a long life span. To gain an appreciation of these it is necessary to return to the basic characteristics of long life software. These were identified as portability, flexibility and cost-effective maintenance. Some of the ways in which a software development can go wrong are:

- (i) poor design: portability and flexibility do not just naturally come about. The software must be designed with these objectives very much in mind.
- (ii) poor documentation: the staff maintaining the software are very likely to change over the life span of the project. If all the knowledge of how the software functions is contained in the heads of the "gurus" who originally developed it, then maintenance costs are likely to be extremely high. Working out how a complex system works can be extremely difficult if the source-code listings are not backed up by comprehensive design and implementation documentation.
- iii) loss of configuration control: the distinction between binary software and source software was made earlier in this paper. In a large system the binary software which runs in the various processors will have been created by a complex translation and construction process involving possibly thousands of source software modules. Only in recent years have the problems of managing large quantities of source software and documentation come to the fore. If one cannot identify precisely the modules (or possibly versions of modules)

which have been used to create the binary software and also reproduce exactly the procedure used to create the binary software, then it becomes impossible to maintain the software. This is known as loss of configuration control. A typical symptom of its occurrence is that when an attempt is made to remedy a particular fault, two distinct faults previously thought to have been remedied are reintroduced.

Standards do exist, or will soon exist, covering each of these areas. Examples of British Ministry of Defence Standards are Naval Engineering Standard 620 (Ref 1) which in turn invokes the use of MASCOT, (RSRE 'Official Handbook of MASCOT', Ref 2) and the CORAL66 language (BS5905, Ref 3) with a view to defining standards for flexible and portable software design. It further invokes Joint Services Publication 188 (JSP 188, Ref 4) which aims at laying down standards for the "Documentation of Software in Military Operational Realtime Computer Systems". In the area of Configuration Control, DEF STAN 05-57 (Ref 5) will include standards for the very important subject of Software Configuration Management. Examples of similar U.S. Department of Defence standards are DoD-STD-480 on Configuration Control (Ref 6) and DoD-STD-1679 on Software Development (Ref 7).

To summarise, the technical skills do exist to permit software capable of having a long life span to be developed and attempts have been made to lay down standards to achieve this. However, a project manager should seek advice from someone with experience of major software developments as to relevant standards and whether to invoke them directly or with some qualification. It is quite likely that some of the standards will be under review and that he can obtain more useful standards by capitalising on the latest developments in this area.

Having established a set of standards applicable to the work, he should turn his attention to gaining confidence that these standards are being adhered to. In broad terms this is the subject of Software Quality Control and it is this that will now be addressed.

Software Quality Control

A set of standards will not by itself control a software development. It must go hand in hand with a project quality plan. A set of standards may be general and applicable to a range of projects. A quality plan on the other hand needs to be project specific and closely tied to the overall project plan. It should define what is to be done in the way of checking for compliance with standards and also precisely when it is to be done.

There may be a temptation for a project manager to say - "there is a department set up to handle quality control and I will leave this in their hands". It must be borne in mind, however, that the sort of quality control techniques applied in traditional engineering areas are not applicable to software and it is most unlikely that the skills necessary for carrying out software quality control will be available in a traditional engineering quality control department.

A software quality plan need not be a large document. Ideally it should be concise, preferably not more than a few pages in length. It is suggested that a Software Quality Plan should comprise the following:

- (i) A statement of the standards to be applied to the work, which may be in-house standards, or other standards (e.g. British, MOD or DoD) which are a contractual requirement. Reference should be made to the relevant documents. This section is effectively a statement of "Quality Requirements" for the work.
- (ii) A list of visible outputs that will be produced at various stages of the project and the formats in which they will be presented for visual inspection. The formats will usually be defined by reference to relevant standards (e.g. for presentation of design sketches or software modules). Like project management, quality control is impossible without visibility and therefore a very important function of the quality plan is to spell out exactly what is to be available for inspection and in what format it is expected to be presented.
- (iii) A schedule, closely related to the project schedule, showing when the visible outputs are expected to be available and allocating time and effort for the quality checking activities required. The actual activities to be carried out will be the reviewing of the visible outputs from either the technical or non-technical point of view. The subject of reviewing is discussed in more detail below.
- (iv) A definition of quality responsibilities and where they lie. This should include: who is to be responsible for carrying out the Quality Plan (quality control responsibility); who is responsible for ensuring that the quality control is taking place (quality assurance responsibility); who is responsible for carrying out each of the activities identified in the Quality Plan.

Reviewing

There are two different aspects to reviewing during a software development and it is worth differentiating between them when constructing a Quality Plan, as different levels of expertise are required in carrying them out. One is effectively checking the 'syntax' of what is being produced e.g. is this sketch presented in the format defined for sketches in the relevant standard?; or does the description of this software module contain an entry in each of the fields as required by the standard laid down for documenting software modules?

The other type of reviewing requires considerably greater expertise and could be considered to be 'semantic' reviewing. This involves answering questions such as: is this software design capable of meeting the functional requirements of this system? Are there any omissions? While the presentation of the design may comply with all standards, and the design approach may comply with standards, the design may still be inadequate. It should, however, be the case that the visibility resulting from the standards adopted should enable an appropriate expert to detect early if all is not well.

This second type of reviewing is of critical importance to the management of a software project and it is the contention of this paper that it is probably the activity most likely to prevent software developments ending in disasters. The topic of reviewing is discussed further in Ref.8.

Quality Responsibility

The aim of this paper is to give assistance to the project manager unfamiliar with software and yet it has been argued that the most important activity, quality control through reviewing, requires software expertise. How is this to be reconciled? The answer is that the expertise required to carry out the majority of the quality control activities should exist within the software development team carrying out the work. The person responsible for the software development work should be required to produce the Quality Plan on day 1 and it should be his responsibility to carry it out. The most experienced technical staff should be identified to be involved in the design reviewing. If the overall project manager is aware that there are highly competent software engineers in the team, then he can proceed with much greater confidence in the knowledge that the mode of working he has enforced will ensure their involvement. If he feels the necessary software expertise is not available he may choose to involve third-party consultants in the quality review process. In either event the result should be visibility and increased confidence for the project manager. Each review should result in a report recording any deficiencies and identifying the remedial actions required before the next review. This 'audit trail' should be quite comprehensible to someone not familiar with software and should provide him with confidence that what needs to be done, is being done.

Note that the approach being suggested does not tie-up the most experienced software personnel on a full-time basis. In fact it utilises this scarce (and expensive) resource in a sparing, well controlled and cost-effective manner.

Visibility

As the definition of visible outputs is critical to the quality control process it is worth making a few further observations in this area. The project manager might expect the visible outputs he wishes to specify in the Quality Plan to be easily definable by reference to an appropriate standard. Unfortunately current documentation standards tend to concentrate too much on the format and detailed content of particular areas. Many do not adequately address the problem of an overall project documentation scheme and how all the documentation related to software should fit into such a scheme. It is worth obtaining some expert assistance at this first stage of the project to define an overall documentation scheme and thereby enable visible outputs compliant with this scheme to be defined within the Quality Plan.

The quantity of documentation required to adequately describe a large software development can be considerable. To give some appreciation of the diversity of documentation required, it is suggested that a Quality Plan for a major software development should define visible outputs for at least the following:

- Functional Requirements
- Design Constraints
- Formal Software Design Documentation (independent of hardware)
- Actual System Documentation (describing the mapping of the
software design onto the hardware)
- Resource Modelling Documentation (a justification that the
chosen hardware has adequate
resources to support the
software design and meet the

functional requirements,
particularly those of
response times)

Test Software Documentation
Integration and Test Strategy
Environment Documentation (all facilities, hardware and
software, necessary to build the
final system from sources)

Project Control Documentation (work plans, etc)
Quality Control Documentation (Standards, the audit trail)

Configuration Control Documentation (everything associated with
ensuring the software is in a
known and reproducible state)

User Documentation (User manuals etc)

Configuration Management

Another problem area worth further comment is that of configuration management. It is a common mistake in implementing quality control on a software development to consider reviews to include only early phases of the design of the software. Reviewing should start before the design begins and continue into the in-service use. This is because many failures in software based systems (perhaps the majority) result from faulty statements of requirements, faulty design adjustments to correct defects or by poor in-service configuration control.

It is suggested that particular attention be paid to reviewing the methods to be used for issuing the software and maintaining configuration control, not only of the software sources, but also of the environment used to create and maintain it. The environment will comprise all of the documentation (it is important for maintenance that documentation reflects the current state of the software) and all of the software 'tools' (compilers, linkers etc) used to create the binary issue of the software.

Configuration control of software is a particular area where recent experience has highlighted the problems that can occur and where adequate standards are just beginning to be defined. To give some guidance, the procedures to be adopted for configuration and issue control should address at least the following topics

- (i) what objects in what formats are to be placed under configuration control
- (ii) how modules and different versions of modules are uniquely identified
- (iii) how modules are bound together into software packages and how packages are uniquely identified
- (iv) how issued packages are protected from accidental update
- v how faults discovered after delivery are reported, how the necessary modification to software and documentation packages is carried out and how the fact that the fault no longer exists is reported.

Configuration management is discussed further in Ref.9.

Software Issue Control

With regard to the issuing of software, a surprisingly high number of software faults arise through human error in the process of creating a binary issue from sources. On large systems this construction process can be quite lengthy and complicated and every effort should be made to automate it as far as possible and minimise the risk of human error. The procedures to be adopted for issuing software should state how this is to be achieved and also state how each issue of software is to be uniquely defined and how it is to be bound to the environment necessary to create it and maintain it.

The problems of binding together an issue of software and its environment are reduced if the items being bound together are in the same format. For this reason it is worth considering treating all the documentation associated with software as software modules and maintaining them in the same way. Software source modules are normally held as character-based files on a computer system. If documentation is maintained in the same form, then some flexibility will be lost in the way that diagrams and equations can be represented but the gains in ease of configuration control will be considerable.

Summary

The important messages which this paper attempts to convey are the following:

- (i) Software is not a black art and should be treated similarly to other engineering disciplines
- (ii) Enforcing up-to-date standards and effective quality control should be of prime concern in managing a software development
- (iii) A quality plan should be project specific and define visible outputs, how and when they are to be reviewed, and by whom they are to be reviewed
- (iv) To have a long life span software must be transportable between different computer types, flexible, and cost-effective to maintain
- (v) An overall project documentation scheme, defined at the start of the project, will considerably assist the quality control task
- (vi) Configuration management is a large problem area to which careful consideration should be given at the start of any new project.

References

1. Naval Engineering Standard 620, Requirements for Software for use with Digital Processors.
2. RSRE 'Official Handbook of MASCOT'
3. BS5905 - Computer Programming Language CORAL66.

4. JSP188 - Requirements for the Documentation of Software in Military Operational Real time Computer Systems
5. DEF STAN 05-57, Issue 2. Configuration Management Policy & Procedures for Defence Material. (Anticipated issue late 1984)
6. DoD-STD-480A, 12th April 1978 - Military Standard, Configuration Control & Engineering Changes Deviations and Wavers.
7. DoD-STD-1679, December 1978 - Military Standard, Weapons System Software Development.
8. Design Reviews, R.Foulkes and A.J.M.Tilbury. Paper presented at the Joint Conference on Engineering Quality. November 1983.
9. Software Configuration Management and its contribution to Reliability Program Management, R.Foulkes and M.P.Mills, IWFE Transactions, August 1983.

ACKNOWLEDGEMENTS

The encouragement given by the directors of YARD LTD is gratefully acknowledged, as is the generous assistance of the author's colleagues. Opinions expressed are those of the author.

SOFTWARE GUIDELINES FOR THE DEVELOPMENT OF DIGITAL CONTROL SYSTEMS

John R. Ware
ORI, Inc.
1375 Piccard Drive
Rockville, MD 20850
USA

ABSTRACT

As the complexity of control and monitoring systems grows and, as the requirements for software implementation, documentation and test increase, it becomes ever more unlikely that the control algorithm development team will be directly involved in the implementation process. This paper puts emphasis on the place of the control algorithm in the software cycle and the importance of establishing quality assurance procedures and standards for algorithm development and delivery in order to bridge the gap between development and implementation. The basic tenets of quality assurance are discussed together with recommended guidelines and methods for attaining the required integrity of the final product. Specific examples are given based upon the internal quality assurance procedures developed by and in use at ORI.

INTRODUCTION

This paper is addressed primarily to the control algorithm designer; but also to the manager who may have overall responsibility for control system implementation and installation. Over the past 10 to 15 years, there has been a virtual revolution in our approaches to control system design. This revolution was partially fomented by the development of optimal control and estimation theory, but the major driving force was the development of small, rugged, and powerful digital computers. The availability of these devices removed many of the restrictions that had been imposed in the analog days. Control systems became ever more complex with on-line estimation, complex adaption schemes, seaway adaptive filtering, trajectory optimization, and so forth. However, as with all revolutions, a new order is required. In the U.S., this new order has taken the form of ever increasing requirements on software development and a proliferation of Navy software specifications, beginning with the (in?) famous WS-8506 down to the present SECNAV 3560.1.

One of the results of the introduction of software requirements, besides the increase in software cost (and, hopefully, quality), is that it is becoming more and more unlikely that the control algorithm designer will be involved in the software/hardware implementation of his algorithm. It is even probable that a different organization or company will be tasked with implementation due to the diversity of talents required for implementation and documentation. A major question facing the algorithm designer is: What can he do to assure that his algorithms are properly implemented with a minimum probability of error? A similar problem faces the overall project manager: How does he know that the algorithms delivered in the final report will meet his functional requirements, are accurately represented in the report, and can be implemented with a reasonable effort and minimum error? This paper will

address some of the ways and means that can be employed to assure the level of quality, implementability, traceability, and maintainability of the control software. In this regard, it is assumed that the underlying theoretical approach to the control design is valid, applied properly, and will actually perform its intended function satisfactorily if it is actually implemented properly.

In order to make this distinction a little more clear, a typical scenario will be presented. This scenario assumes that the common U.S. Navy control development procedures are followed. That is, the Navy either tasks a control design organization (which may or may not be an element of the Navy) to provide a control algorithm which is then provided to an implementation contractor as GFI (Government Furnished Information); or, a contract is let to develop the entire control system from scratch. In this latter case, the winner of the contract will most probably be a large hardware-oriented firm (because the majority of costs and potential profit lie in hardware) which will subcontract to another organization for the control algorithms. It should be realized that the "customer" in the following scenario is the overall project manager. He has most likely been selected for his position based on his managerial skills and not for any knowledge of control system design. His concern is with the final product and the algorithms may be only a small part of that deliverable. The scenario:

1. The control system designer receives a task to develop a control algorithm for some system. He reads and understands the specifications, if any, and understands the customers needs.
2. He selects a design approach, for example, optimal control theory, and begins formulating specific equations.
3. When he feels that he has enough information to begin, the equations are programmed and interfaced with a simulation of the dynamics being controlled.
4. Simulation results are analyzed and modifications are made to the control equations, parameters, or whatever until the designer is satisfied with the results.
5. The control designer writes a report describing the control algorithms and, perhaps, providing simulation results and a derivation of the equations.
6. The report is delivered to the customer.
7. The customer turns the report over to a software developer.

A major question that can be asked with regard to the above process is: How does the customer (and, indeed, the designer himself) insure that the delivered algorithms in the report conform to (a)- the theory developed and (b) the simulation used for evaluation? Of less immediate importance to the designer and his customer, but of very great long-term interest are:

1. How will the reported algorithms be turned into software and what can be done to minimize the potential errors and decrease cost?

2. Should the specifications or needs change, or should "bugs" be found in the algorithm how can they be corrected with minimum cost and maximum assurance of success?

The remainder of this paper will discuss the relative importance of the control algorithms in the total software effort, the potential impact of "minor" errors, and, of most importance, how to maximize product assurance.

PUTTING CONTROL ALGORITHMS IN PERSPECTIVE

More and more software systems are being put under rigid development specifications and this is nowhere more true than in the U.S. Navy combat system environment. These specifications require an extremely formalized approach to software development, documentation, and test. In this environment, it is common for delivered software to cost \$100 to \$200 dollars per line of code even when starting with a set of programmable algorithms. Add to this that, even if the entire software and hardware system being built has as its sole purpose the control of a particular set of dynamics, it is not uncommon that peripheral systems (such as operating system, input data conversion and checking, fault location, self-check, data recording, output processing, etc.) will occupy 80 to 90 percent of the total system resources in terms of both memory and computing time. Thus, even though we control system algorithm developers tend to consider that the world revolves around us, the actual cost of algorithm development may be considerably less than 5 percent of the total system cost. Is it any wonder that Navy program managers spend relatively little time and effort in the algorithm development area?

Nevertheless, the control algorithms are the central feature of the final system. Without them there is just a black box with blinking lights. Further, we algorithm designers can have a tremendous impact on schedule and costs if we make even the smallest mistakes. As an example, in one control algorithm delivered to the Navy and turned over to a software/hardware developer, it was not discovered until late in the effort that a "less than" should have been a "less than/equal". In other words, one simple horizontal line was left out inadvertently. However, this single stroke of a pen necessitated the issuance of an ECP (engineering change proposal), review by the software configuration management board, updating of all pertinent software documents (performance specification, design specification, subprogram design document, test plan, and test procedures) with their attendant quality control, and software retest. Cost to the Navy: at least 25 thousand dollars and one week of schedule time!

In other words, while nobody cares if we're right; everybody cares when we're wrong. Now, what can we do to make things better for our customer and incidentally, make us look and feel better ourselves? The answer is to develop and follow a product assurance plan and stick with it throughout the project.

WHAT IS THE PRODUCT?

The first question to answer is: What is the product we should be producing and delivering? In the "good old days" it was enough to deliver that simple control algorithm report with some derivations and a few simulation results. The report might look not unlike a technical paper

delivered at the control symposium. That will no longer suffice. We must expand our areas of technical responsibility and expertise to cover, at least, the mapping of our control algorithms into the proper format for digitization and the development of test scenarios. Additionally, we should provide logic to detect invalid data and schedule algorithm actions when invalid data is encountered and to check operator inputs for consistency and reasonableness.

At first glance, some of these topics might not appear appropriate for the algorithm designer to consider and not only must we convince our customer of their relevance, we must also convince ourselves. However, with a little more thought we can see how each of these items is naturally the province of the control engineer. After all who better to know whether the input data is of such a poor quality that it can no longer be used for control? Who better to know what the algorithms should do when invalid data is present? Who, with his detailed knowledge of the algorithms, can better design test scenarios and predict results?

PRODUCT ASSURANCE

Now that we know the product that is being delivered, we can consider what we must do to convince ourselves and the customer that his deliverable is what it should be and that it will be capable of future "maintenance" actions. The three major items of product assurance are: visibility, traceability, and integrity.

Making a control algorithm visible can be a difficult task because of its abstract nature. Things that are abstract tend to be ignored by management, and even the customer; and it is part of the algorithm developer's task to keep his project visible by carefully defining the product and its extent and limitations so there are no surprises in the final delivery. Of even greater importance is to insure that management is visible to the personnel working on the project so that they are aware of the importance of the project and the importance of the product assurance methods that are being applied.

Traceability means that we can identify the origin of each part of the control algorithm and follow its development thorough various modifications and identify the causes for the modifications and the rationale behind them.

Product integrity means that the deliverable meets the contract requirements, fulfills the users expectations, meets all specified performance criteria, and is as accurate and as error free as possible.

In the following sections, we discuss the techniques and tools for attaining these goals.

TECHNIQUES FOR PRODUCT ASSURANCE

The techniques for product assurance for a control algorithm are similar to those used for product assurance for software with a few exceptions. The major items consist of evaluation, configuration control, configuration auditing, and product test.

Evaluation is what is normally done in the development of an algorithm in that the algorithms are simulated and interfaced with a simulated dynamic system and environment to determine performance. The resulting performance is then compared to any written specifications for performance (e.g., RMS course error shall be less than 2.5 degrees in a Sea State 5 quartering sea) and/or the subjective opinions of the algorithm designer and the customer. The major difference between what is normally done and what must be done to "assure the product" is that there must be proof that what is being evaluated corresponds with the theory developed and the delivered report.

Configuration control means that all of the descriptors and representations of the algorithm must be controlled in such a way that no unauthorized and unreviewed changes are made and that all authorized changes are made to all the descriptors in exactly the same manner. Normally there are several representations of the control algorithms, depending upon the stage of the development process. For example, there will usually be the developers derivation notes, a more detailed description of these notes to be used by the programmer (i.e., a translation from the cryptic mathematical notation into some understandable form), the program being used in the simulation evaluation process, and a draft of the final report. Every possible precaution must be taken to assure that each of these representations are equivalent.

Configuration auditing is the process of checking the algorithm descriptors, described in the previous paragraph, for uniformity. While, to a great extent, this uniformity should be assured by careful adherence to configuration control policies, the possibility of error still exists and should be reduced as much as possible. A configuration audit should be performed at the point at which it is felt that the algorithm and simulation programs have undergone all evaluation steps and are ready for final testing. The audit consists of a review of all the documentation and program listings for consistency and should, in an ideal situation, be performed by a person or group who has not been involved in the algorithm development or programming. If an independent group is not used it is very likely that mistakes and misunderstandings will perpetuate themselves and may not be discovered until much later in the process, very likely sometime after final delivery during the implementation phase.

Product test is used to describe the testing that is done, not to assure that the algorithms provide the necessary control actions, but to assure that the deliverable documentation represents the original control algorithms and corresponds to the simulation program that was used during the evaluation phase. It is very easy to become complacent with regard to the assumed accuracy of all representations of the control algorithm, particularly if the schedule is tight and time and dollar constraints come into play. Additionally, the layout of this type of testing and the required computations are extremely tedious and require an exacting attention to detail; attributes which are not always found in control algorithm designers. Especially for these reasons, this portion of the quality assurance process requires an extraordinary effort and discipline from the entire team. It is the portion of the process that is most likely to be ignored.

In the following section, we will discuss some tools and methods that may be applied to achieve the quality assurance needed and to accomplish the techniques listed above.

TOOLS AND METHODS

The tools and methods that will be described in this section to attain the quality assurance desired and then execute the major steps in the product assurance procedure are the result of almost 15 years of experience in developing digital control systems. They have evolved slowly, primarily because, while I am now a major advocate of the process; I initially resisted every step (even the ones that I proposed). After all, I reasoned, my group does not make mistakes; why should we go to all the extra bother, time and expense of plodding through these additional procedures that contribute little to the final product (in terms of theoretical development). We have found that nothing could be further from the truth. We have found that, in actual practice, no matter what method we use to develop the algorithms, whether optimal control or frequency response methods; no matter how clever and inventive we were in their application; we probably will not derive a control algorithm that improves performance over previous automatic or manual systems by more than a factor of two or so. While this may seem significant, it may mean very little in actual practice and it may be difficult or impossible to assess the improvement in combat capability that results. However, if we fail to transmit the new control algorithm correctly to the implementation contractor, we not only lose the potential improvement; we also may cost the Navy a large amount of money and time and we most certainly lose some of the confidence that our customers have felt in our capability. In other words, the best control algorithm in the world does very little good if it is never utilized.

In order to focus the discussion of the methods to be described, an example will be used that was partially reported in a previous ship control symposium. This controller was to be used to control lateral and longitudinal separation of ships during underway replenishment as well as providing control during approach. Thus, the system had to control both the rudder and the engine, including acceleration/deceleration to insure rapid closing to the UNREP condition. A further requirement was that the algorithms must filter certain sensed variables for display to the conning officer and provide quickened helm and heading information if requested. While this was not an extremely complicated control algorithm, as compared to control algorithms for submarines, it, nevertheless, will serve to illustrate the major tools used.

It might appear, at first reading, that use of the tools and methods described below would impose a considerable burden on the algorithm development process. However, particularly if the design team is small, the process can be relatively informal (when compared to military specification requirements for software documentation) and guided by common sense. The sole reason for the additional effort is to assure the integrity of the final deliverable. A little extra effort up front will pay for itself many times over as the delivery deadline approaches.

SOFTWARE SPECIFICATIONS

Before actually discussing the tools in some detail, a word about software specifications. You, as the algorithm developer, are not delivering software. But, what you are delivering will be software someday and that software will be written to some specification. Much of that specification is not relevant to you, in fact, you could argue that none of it is.

Nevertheless, you can make a substantial contribution to the implementation process by understanding those few elements of the applicable specification that will effect your work. These would include structured programming requirements, module size, variable name rules, and so forth. Very little effort up front on your part can pay big dividends to your customer later on.

MASTERBOOK

We use the idea of a "Masterbook" for control algorithm development and documentation. The Masterbook contains all relevant information with regard to the project and is maintained by the project technical leader. For convenience, it is usually in the form of a loose-leaf binder (or set of binders) which permit the easy insertion of additional material as required. The information in the Masterbook consists of:

- o Relevant contractual information (operational specifications)
- o Action items resulting from various meetings as the contract progresses
- o Changes to functional requirements defined in the original contract
- o Internal and external schedules and milestones
- o Project responsibility assignments
- o Definition of control modules
- o Rough draft of final algorithm report
- o All theoretical derivation and investigations even if they should prove to be rejected for any reason (including gross errors).

No information is allowed into the Masterbook until it has been reviewed by the project leader and no changes to information in the Masterbook can be made by anyone except the project leader. Thus, even if a team member derives the original algorithm for some portion of the controller, once this algorithm is entered in the Masterbook, errors can only be corrected by the project leader. Changes are dated and documented on insertion sheets adjacent to the original flawed pages. Flawed pages are not discarded but are left intact to document the work.

Thus, the Masterbook, as simple as the concept appears, provides the keystone for the product assurance effort. It is the single repository for all information relevant to the project and can be consulted by any team member. For example, if the person responsible for developing the parameter scheduling module should discover what he feels to be an error in some other element, he can consult the Masterbook to determine the origin of the information as well as the person to contact to discuss the problem. In a small project, such as the UNREP study, this is not critical but, as the size of the project increases, it becomes correspondingly more important.

MODULARIZATION

After attaining a full understanding of the functional requirements of the control algorithm, by study of the contractual requirements and via meetings with customer representatives, the control algorithm is separated into modules, each of which will execute one or more of the basic functions required. The purpose of modularization is twofold: first, by defining the functions of each module the resulting function list can be checked against the functional requirements to see if all requirements will be addressed. Second, by dividing the algorithm into modules, later testing and configuration auditing is made much simpler and more efficient.

This modularization and definition need not be extensive in order to accomplish these goals. Table 1 shows the modules and functions used in the UNREP controller. Each module has its own section of the Masterbook which typically contains a detailed discussion on the internal processing required to accomplish its requisite functions, initialization, processing, Input/Output Tables, and nomenclature tables.

NOMENCLATURE AND FORMAT

Even as simple an item as nomenclature and presentation format can be the source of errors and complication in the implementation process. In our early control system algorithm reports, equations were written in much the same way as they appeared in the handwritten derivations. That is, they contained Greek letters, superscripts, subscripts, "hats", tildas, and so forth. Not only did this cause significant confusion with the initial typing of the reports but errors, even with the most diligent proofreading, were inevitable (proof reading assumes that the original can be properly interpreted). Additionally, during the implementation process many questions arose with regard to interpretation of equations (e.g., $X = AB$. Does this mean X is equal to A times B or X is equal to the variable AB ?). Additionally, with the advent of word processors, it was found that many of these were limited in the complexity of the symbolism available with the result that certain symbols would have to be added to the text by hand or by a second pass through the word processor. More problems occurred if revisions were required because even simple revisions might necessitate a retyping of the report with a significant expense due to the tremendous amount of hand work needed.

For these reasons, nomenclature and format rules were established that would effectively preclude hand work from the final report and would result in a report that could be programmed on virtually any word processor. These rules were:

- o No symbols other than the standard English capitals and arabic numbers plus the symbols: * () - + = /
- o No superscripts or subscripts, all symbols for a given equation must be on the same type line
- o Every variable must be separated from another variable by an operation sign.

Table 1. Module Functional Definitions for UNREP Control
Algorithm Development

1. INPUT PROCESSING MODULE (IPM)
 - o Initialization
 - o Preprocessing
 - o Signal filtering
 - o Invalid data handling
 - o Input conversion
2. PARAMETER SCHEDULING MODULE (PSM)
 - o Generate control gains as functions of ship speed
 - o Generate filter weightings as functions of speed and display mode
3. LATERAL RANGE MODULE (LRM)
 - o Generate rudder commands to provide correct lateral separation
4. SPEED CONTROL MODULE (SCM)
 - o Generate time optimal speed/longitudinal separation trajectory for approach
 - o Generate appropriate Power Lever Angle commands for engine control during approach and while maintaining station
5. DISPLAY MODE MODULE (DMM)
 - o Filter lateral distance, longitudinal distance, relative heading and their rates for display to the conning officer
 - o Provide aided helm display for helmsman
 - o Provide quickened absolute heading recommendation for conning officer
6. SYSTEMS ROUTINES MODULE (SRM)
 - o Defines all arithmetic, algebraic, and trigonometric notation to prevent any possible ambiguities of interpretation.

- o Every variable name must follow the variable name rules for the expected implementation computer. In the absence of knowledge of the computer and its language, the current FORTRAN 77 rules apply.
- o No dimensioned variables
- o Variable names should be readily interpretable, where possible. (E.g., RBDT for relative bearing rate.)

An example page for the UNREP report shows the result of application of these rules, Table 2. Note that, while the format appears cumbersome and space consuming, even a very moderately skilled programmer could use this document to establish his implementation program directly from the report without the necessity of an intermediate translation into "computerese". Notice also that the filter equation (the last equation in the Table) is already in discrete format. This brings us to the next topic.

DIGITIZATION

All differential equation digitization should be done by the control algorithm design team, not by the implementation contractor. The typical implementation contractor software employee knows how to program. He does not know control theory, z-transform theory, anything about stability margins, or about a host of other topics which are required to transform an algorithm into viable code. Therefore, it is strongly recommended that, whenever possible, do not submit control algorithms in the form of differential equations or in Laplace transform notation for implementation. They may be submitted in this form in a report which documents the derivation of the algorithms, but this submittal should be separable from the equations to be implemented.

NOMENCLATURE AND I/O LISTS

It has been stated that, if you define the inputs and outputs of a system, you have defined its functional capabilities. While this statement may be an exaggeration to some extent, it does indicate the importance of defining all the required system inputs and the system outputs. By assembling the system I/O list as early in the process as possible, the customer and the implementation contractor can be made aware of any special sensor or preprocessing requirements that may be needed. The I/O list will also define the orientation of the input coordinate axes system as well as defining the units of input and output.

A nomenclature list, while strictly speaking not required for implementation purposes, will be extremely helpful during module testing, code auditing, debugging, and testing. Items in the nomenclature list should be identified either inputs to the program, outputs from the program, variables within the program, or constants within the program. Variables should be identified as to which module is their source and, if applicable, which module is their destination.

Table 2. Page From Typical Control Algorithm Final Report.

LATERAL ERROR FILTERING

Lateral error is filtered by a notch filter tuned to own ship's roll period.

Initialization of Filters

$XLATE(-1) = XLATE$ IF ((LPWR = 1 AND LSLNT = 0)
 $XLATE(0) = XLATE$ OR (ON TRANSITION FROM LSLNT = 1
 $XLATEF(-1) = XLATE$ TO LSLNT = 0))
 $XLATEF(0) = XLATE$

Computation

$TLAT = CTLAT * DT$
 $FLAT1 = EXP(-CNF1 * TLAT)$
 $FLAT2 = COS(TLAT * CNF2)$
 $FLAT3 = COS(TLAT)$
 $KLAT1 = FLAT1 * FLAT1$
 $KLAT2 = 2 * FLAT1 * FLAT2$
 $KLAT3 = 2 * FLAT3$
 $KLAT4 = (1 - KLAT2 + KLAT1) / (2 - KLAT3)$

 $XLATEF(N) = KLAT4 * (XLATE(N) - KLAT3 * XLATE(N-1)$
 $\quad + XLATE(N-2) + KLAT2 * XLATEF(N-1)$
 $\quad - KLAT1 * XLATEF(N-2)$

NOTATIONAL DEFINITIONS

It is important to define all notation and operational symbols used in the control algorithm report no matter how self-evident they appear. The importance of the inclusion of this feature cannot be overestimated and we have established a standard format and definitions which cover the following topics:

1. Function definitions which covers such items as LIM, ABS, SIN, COS, ASIN, and so on with input and output units defined as required.
2. Mathematical operation notation including multiply, divide, and so forth. In our notation, for example, the "equal" sign is used to define replacement as it is used in programming languages.
3. Relational symbols such as .LT. (less than), .GT., and so forth.
4. Logical (or Boolean) operations including IF, AND, and OR.
5. Difference equation and initialization notation for discrete filters.

TEST SCENARIOS

Whenever possible, module and total algorithm test results should be generated by the algorithm developer and published in the algorithm report or in a special Test Procedures document. The reason for this is that the algorithm developer knows the algorithms intimately and is most qualified to produce test scenarios which will trigger various algorithm options in ways that will highlight possible errors. The test scenarios serve two purposes: first, they provide a method for checking the algorithm delivery; and, second, if done properly, can be used directly in the Program Test Plan and/or Program Test Procedures (PTPL, PTPR) that will be issued by the implementation contractor.

Tests should be conducted at the module level first and this is best done by constructing "event drivers". An event in this context is a complete set of module inputs together with a time of occurrence. A single test consists of one or more events with the event driver constructed in such a way that all previous inputs are held constant until the next timed event takes place. This procedure allows for simple tests, such as step inputs, to be run using a single event. Module tests should be constructed to test every element in the module and, to this end, inputs need not be restricted to those that would naturally occur in service. For example, consecutive ship headings one second apart of 10, 20, and 30 degrees would not be possible in real life, but might make checkout of a particular module segment easier than a more normal test sequence.

Once the choice of test events has been determined, the test results are computed via two different methods. The first is a hand calculation using as a basis the original control algorithms from the Masterbook. These are checked against results obtained from a programmed version of the algorithms which is derived from the final report. Any discrepancies that occur must be resolved by the project leader who should write up a description of the discrepancy and its resolution. The calculation of test outputs by hand is a tedious task which can be substantially speeded up using hand calculators. However, using the computer program to generate its own test results is begging the question.

The purpose of the module tests is to assure that all computations are being performed correctly. Once this is assured, total algorithm tests must be performed to assure that information required as inputs from other modules is properly transferred. Because of the previous care taken with regard to module tests, it should now be relatively easy to design test scenarios to test the entire control algorithm.

It is very important that the test scenarios be totally driven by an external event driver and not be closed loop using a simulation of the controlled dynamics. If this is not done, then the controlled dynamics must also be put under configuration control and be part of the delivery to the implementation contractor. This will, of course, substantially increase costs. Another problem that occurs when closed loop simulations are used to test software is that the actions of the control system tend to mask errors in programming. The reason for this is that control algorithms are usually designed to minimize sensitivity to parameter variations with the result that closed loop behavior can be nearly invariant even if control coefficients are incorrectly computed by even a substantial amount.

THE DESIGNER AND THE CUSTOMER

What should you do as a control algorithm designer to attain quality assurance in your product and please your customer?

1. Convince yourself and your staff that these procedures are worthwhile and will pay off in the long run.
2. Educate your customer that the extra time and effort that will be expended will pay for itself many times over during implementation and test.
3. Establish an internal configuration management procedure and quality assurance plan (or use the one that I proposed) and stick with it. This may require extreme discipline particularly when the going gets rough and deadlines are near. That's the time you need it most!
4. Convince your customer that the algorithm design team should be involved in the implementation, documentation, and test process. Your participation should be as consultant and reviewer. Many times the implementation contractor will face what he feels are overwhelming problems that can be solved with some insight into the control design process.

What should you as a customer do to guarantee that you receive a quality product that you can turn over to another organization for implementation?

1. Insist that your contractor establish and follow quality assurance and configuration management procedures.
2. Demonstrate to your contractor that you are interested in his effort and are convinced of its importance.
3. Insist on demonstrations of quality assurance such as code audits, module tests, and acceptance tests.
4. Insist that the algorithm developer maintain contact with his work as the implementation progresses.

SUMMARY

Whether we, as control algorithm designers like it or not, our work is considered as part of the software development process and often not even a very important part of that process. Even if the entire software/hardware system is designed to house the control algorithms and they are the reason why the system was built, total algorithm development costs may be only 5 to 10 percent of the total system cost, if that much. Whatever is spent on the creative work of control design will be dwarfed by the amount of time and dollars spent on implementation, documentation, and test. The major reason for this is the proliferation of software development specifications particularly for military applications. Because of the increasing complexity of these specifications, new skills are required and it is extremely unlikely that the control algorithm designers will either have these skills, be willing to acquire these skills, or be able to win a contract for implementation. The trend in the U.S. for some years has been that control algorithms are turned over to an implementation contractor and become part of Government Furnished Information. This places a tremendous burden upon the algorithm developers to present their work in a form and fashion that assures successful implementation.

At a minimum, this means that the work should be as error free as possible. However, we can and should go well beyond this minimum by following rigid procedures to control the development process, its simulation software used for evaluation, and the final deliverable. After all, what is at stake is our reputation; and, in the last analysis, that is all we really have.

STATISTICAL ANALYSIS OF AUTOPILOT PERFORMANCE

by J.B.D. Rush BSc
and D.R. Broome PhD BSc
University College London

INTRODUCTION

It is recognised that the proportional and derivative feedback gains of a conventional PID autopilot may be tuned to give a minimum fuel consumption for a ship, when travelling between fixed points at a predetermined speed. There is however no agreement as to the magnitude of the constraints in the quadratic cost function which is usually used and against which adaptive autopilots may be optimised. The magnitudes of the weighting factors in the cost function may be derived from a detailed knowledge of the hydrodynamic behaviour of the ship, but the complexity and uncertainty of this approach to the problem is such that it has yet to find widespread application. Furthermore, the behaviour of the ship is also subject to change with such things as loading and sea condition.

From the results of model trials and extensive simulation work it is proposed that a method exists whereby autopilot tuning for minimum drag may be achieved, which does not require a knowledge of the ship's hydrodynamic derivatives. Furthermore, the method is an on-line process, and is sensitive to the effects of changes in sea state and ship's loading condition, as they affect the optimum tuning of the autopilot.

SIMULATION CONSTRUCTION

Simulation Philosophy

A digital simulation of a fast container ship was written, and was used to investigate the effect of autopilot setting on the ship's fuel cost. As such the simulation included a full propulsion system, with the prime mover characteristics taken from a manufacturers published Diesel engine data. The nonlinear behaviour of the propeller was simulated using lookup tables and interpolation so that the characteristics of a real propeller could be reproduced more exactly. The ship included a full PID autopilot although for these studies the integral feedback was not used, and only the effects of varying the proportional and derivative gains was investigated. The autopilot included dead bands in the sensing signal lines and in the error signal delivered to the steering engine. The rudder mechanism was simulated as a dynamic system rather than simply giving a rudder angle proportional to the error signal, and included stiction and friction, rate of turn limiting and rudder inertia. To simulate the motions of the hull both the yaw, sway and surge equations were used, with a random wave series impinging on the hull and developing perturbations in rudder inflow velocity due to water circulation.

4.1 Motions

The ship's yaw and sway response to the external wave forces and its own steering actions may be described by the usual linearized equations of motion for these two degrees of freedom. That is:-

$$m(\dot{v} + ur) = Y_v v + Y_\delta \dot{\delta} + Y_r r + Y_{\dot{r}} \dot{r} + Y_\delta \delta + Y_w \quad (1)$$

and

$$I \dot{r} = N_r r + N_{\dot{r}} \dot{r} + N_v v + N_{\dot{v}} \dot{v} + N_\delta \delta + N_w \quad (2)$$

where

m = ship's mass
 I = ship's moment of inertia about the z axis
 δ = rudder position
 Y = response factor in sway to the parameter x
 N_x = response factor in yaw to the parameter x
 v = ship's sway velocity
 u = ship's forward velocity
 r = ship's yaw rate
 Y_w = lateral external wave force on the ship
 N_w = torsional external wave force on the ship.
 The dot signifies a time derivative. The motions of the hull are assumed to be such that the ship turns about a point just aft of the bows, so that the sway velocity is proportional to and in phase with the yawing motions of the ship. Laplace transforms may be taken of these first two equations, and the sway velocity and its derivatives eliminated to give an equation of yaw motion in terms of the ship parameters, the rudder actions and the external wave forces on the hull.

$$\begin{aligned}
 & \{[(m - Y_v)(I - N_r) - N_v Y_r] s^2 - [N_r(m - Y_v) + N_v Y_r + N_v(Y_r - mu)] \\
 & + Y_v(I - N_r)] s + N_r Y_v - N_v(Y_r - mu)\} r = \{[N_v Y_\delta(m - Y_v)] s + N_v Y_\delta - Y_v N_\delta\} \delta \\
 & + \{N_v s + N_v\} N_w + \{(m - Y_v) s - Y_v\} Y_w
 \end{aligned} \quad (3)$$

This equation is frequently simplified by assuming a constant forward velocity and thereby allowing lumped constants to be formed from the various terms. In this simulation the speed was not assumed to be constant, but was derived from the surge equation given below.

$$m \dot{u} = T_p - X_{uu} u^2 - X_{uu\delta} u^2 \delta^2 - .765 m L r^2 + X_w \quad (4)$$

where T_p = propeller thrust
 X_w = wave forces in longitudinal direction

The numerical term in equation 4 is derived from the yaw drag, as can be seen, in terms of the distance from the turning point of the vessel to the centre of gravity (0.45L) and the mass and added mass of the ship (1.7m). The sway velocity is often omitted from ship simu-

lations because it does not appear directly in the surge equation. In single screw vessels however it does affect the propeller inflow velocity and thus causes a perturbation in the propeller thrust. Having found the ships yaw rate and its derivatives the sway velocity may then be calculated from equation (1).

The wave forces on the ship's hull in the previous equations are calculated from an integration of the pressure forces acting on the hull which are caused by differences in water height around the perimeter of the ship. This assumes that the ship is travelling in a fully developed sea of long crested gravity waves, and that the wave forms are themselves not distorted by the ship. The simulation does not account for any dynamic exchange of energy between the vessel and the waves. On this basis the forces are given by the following equations:-

$$Y_w = -2aLs \frac{\sin b \cdot \text{sinc}}{b} \quad (5)$$

$$X_w = 2aBs \frac{\sin b \cdot \text{sinc}}{c} \quad (6)$$

$$N_w = a \cdot d \cdot \frac{L^2 \sin b (c \cos c - \text{sinc}) - B^2 \text{sinc} (b \cos b - \text{sinc})}{c^2 b^2} \quad (7)$$

where

Y_w = lateral thrust on the hull
 X_w = forward thrust on the hull
 N_w = turning moment on the hull

and

$$a = \rho g (1 - \exp(-kT))/k$$

$$b = \frac{kL}{2} \cos \chi$$

$$c = \frac{kB}{2} \sin \chi$$

$$s = \frac{kh}{2} \sin (\omega_e t)$$

$$d = \frac{kh}{2} \cos (\omega_e t)$$

and ω_e = wave encounter frequency for the ship

that

$$\omega_e = \omega - k u \cos \chi + k v \sin \chi$$

and

k = wave number

ω = wave frequency relative to inertial axes

u = ship's forward speed

v = ship's lateral velocity
 h = wave height
 ρ = water density
 χ = wave encounter angle
 L = ship's length
 T = ship's draught
 B = ship's breadth amidships
 g = acceleration due to gravity
 t = elapsed time

Equations (5) and (6) above may not be used in a digital simulation at encounter angles of 90° and 0° respectively, since in both cases the denominators of the equations tends to zero whilst the forces on the hull are not zero. To circumvent this, each equation may be simplified by allowing the encounter angle to tend to the relevant limit and simplifying. This gives the following two equations:-

$$X_w = \rho g L B T \cos \chi \frac{kh}{2} \sin(\omega_e t) \quad (8)$$

as $\chi \rightarrow 0$

and

$$Y_w = \rho g L B T \sin \chi \frac{kh}{2} \sin(\omega_e t) \quad (9)$$

where the terms in the above equation have the meanings given previously. The elapsed time in the above equations is the time since the start of the wave coincided with the mid-ships position.

Wave data

The wave data in the simulation has three principle variables, wave direction, wave frequency and sea state. Of these three variables the first two may be set to a suitable value for each simulation run, but remain fixed whilst the simulation is running. The third variable, sea state, is determined by the selection of a wave file containing a random sequence of wave heights whose distribution corresponds to the wave height distribution of standard (ITTC) wave spectra. Table 1 shows, for a global average, the relative frequencies of different wave heights and the representative wave periods for the sea states that were encoded into the random wave files. The wave period data is not included in the data files themselves, but may be used as required in the simulation in terms of the wave frequency. Some approximations were made in terms of the wave excitation apart from the simplifications of a constant wave direction and constant frequency and the modelling of a fully developed sea. It was also assumed that the waves were longer than the ship and that the ship therefore would encounter only one wave at a time. This is clearly not true no matter how long the waves are, since the ship must move from one wave to the next as it progresses relative to the waves. The above equations are such that they assume that a continuous series of waves of a given height 'h' are impinging on the ship. Since the wave data files contain a series of random and different wave heights there will be a step in the calculated wave forces on the hull each time a new and different wave height is taken from the wave data file. The next wave height is taken from the file when the $(\omega_e t)$ terms in equations 5+9 reach a value exceeding 2π . Although this inaccuracy exists in the simulation it was felt to be an acceptable error, since a random perturbation of the ship's course is nevertheless still being generated by forces acting on the hull within the expected range of frequencies.

Table 1. Wave Height Distribution For Sea State Files

Wave Height (m)	Wave Period (Seconds)			
	2.5	6.5	8.5	10.5
0 + 1	58	16	7	5
1 + 2	37	58	40	27
2 + 3	4	21	39	42
3 + 4	1	5	14	26
Total%	100	100	100	100
Sea State	1	2	3	4

This same argument may be extended to the omission of wind and current loading from the simulation, since a regimen which would minimise a ship's fuel consumption in the presence of wave loading could also be expected to be effective with other similar disturbances.

Propulsion System

The simulated propulsion system consist of a marine Diesel engine, a gearbox and a single propeller. The characteristics of marine Diesel engines and propellers have been widely investigated and published, and the behaviour of a specific example is reproduced in this part of the simulation. An important aspect of the Diesel engine is the proportionality that exists between speed and fuel flow rate to the cylinders, and this is included using a simulated governor with the following equation:-

$$mR_A \ddot{\theta} = (R_0 + R_A \sin \theta) \cos \theta (2\pi n)^2 - mg \sin \theta - k\theta - B\dot{\theta} \quad (10)$$

where R_0 = fixed minimum flyweight radius arm length
 R_A = moving flyweight radius arm length
 θ = radius arm angle
 m = flyweight mass
 k = opposing spring, spring constant
 B = radius arm damping factor
 n = engine speed

The input to this equation is the engine speed, so that for each iteration of the simulation a new value of the angular acceleration may be found from this and the previous values. Equations are also included to limit the angular range and angular velocity, and the constants in (10) are arranged to give a critically damped system. The output variable from the governor is the angular position θ , which is then used in conjunction with the engine speed to find the rate of flow of fuel, with the following equation:-

$$\dot{m} = K_1 n (X_0 - K_2 \sin \theta) \quad (11)$$

where $\dot{m}$ = fuel flow rate per engine cylinder
 n = engine speed
 K_1 and K_2 are constants and are arranged to give the correct

range of fuel flow rates for the range of engine speeds and engine torques described by the subsequent equations. The factor X corresponds to the input power control rack of the Diesel engine, and in other studies with this digital simulation it was linked to feedback loops sensitive to various engine and propeller parameters. In these studies of fuel consumption the input rack position X was controlled by a servomechanism designed to control the engine speed. Other research indicated that fuel consumption was affected by the choice of controlled parameter in the propulsion system and that speed was not the best choice of parameter. However, these effects were independent of the autopilot setting, which is the main topic of this paper, and these results with speed control of the engine may be regarded as being applicable with other types of propulsion control. The fuel flow rate to the engine is then taken to be:-

$$\dot{M} = \dot{m}N \quad (12)$$

where $\dot{M}$ = total fuel flow rate
 N = number of engine cylinders.

It is the total fuel flow rate that is logged by the simulation for the purpose of assessing the ship's fuel cost. It is possible under dynamic conditions that the governor will deliver more fuel to the engine than can be usefully used, and this was also accounted for, but in these studies did not occur. The new fuel flow rate and existing engine speed may be used to find a corresponding engine net output torque according to the engine characteristics. These are shown for a series of fuel flow rates in figure 1, as single cylinder output, so that the net torque from the engine equations must be multiplied by the number of cylinders. Within the simulation the new engine speed is produced by integration of the propulsion system angular acceleration, which is itself calculated from the engine torque, the propeller load torque, angular inertia and propulsion system losses.

The propeller torque and thrust are calculated in the usual way from values of K_T and K_Q referenced to the advance coefficient J , and the propeller characteristic curves are shown in figure 2. Determination of the inflow velocity takes account of the water circulation in the waves and the vessels sway velocity. For fully developed gravity waves, the horizontal component of the circulation velocity is given by:-

$$U_1 = \frac{h}{2} \omega \exp(-zk) \cos(kx - \omega_e t) \cos \chi \quad (13)$$

where U_1 = horizontal circulation velocity component
 ω = wave frequency
 k = wave number
 χ = wave encounter angle
 h = wave height
 t = wave elapsed time
 z = propeller depth below mean surface.

The velocity component given by (13) is taken to be additional to the normal inflow velocity caused by the ship's speed and the hull shape. It has been found empirically that for single-screw vessels the inflow velocity is affected by the vessel's sway, and that this

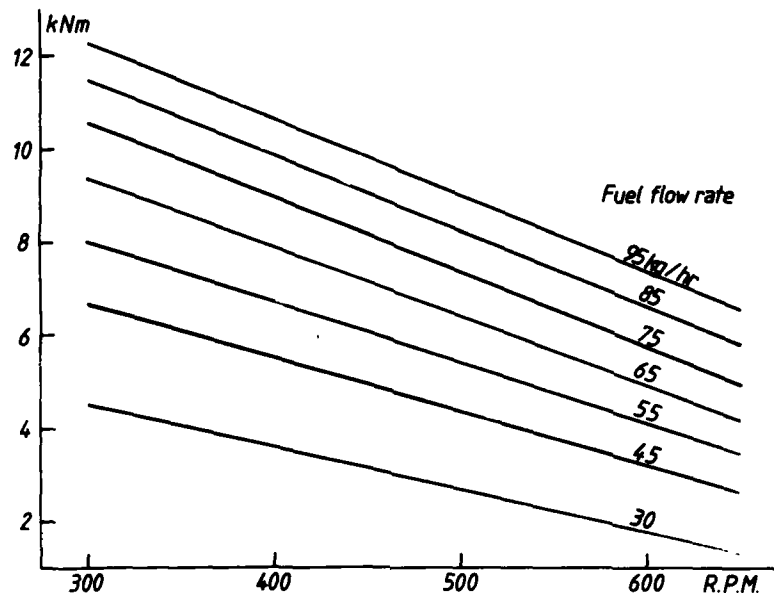


Figure 1 Diesel Engine Torque/Speed/Fuel Characteristics

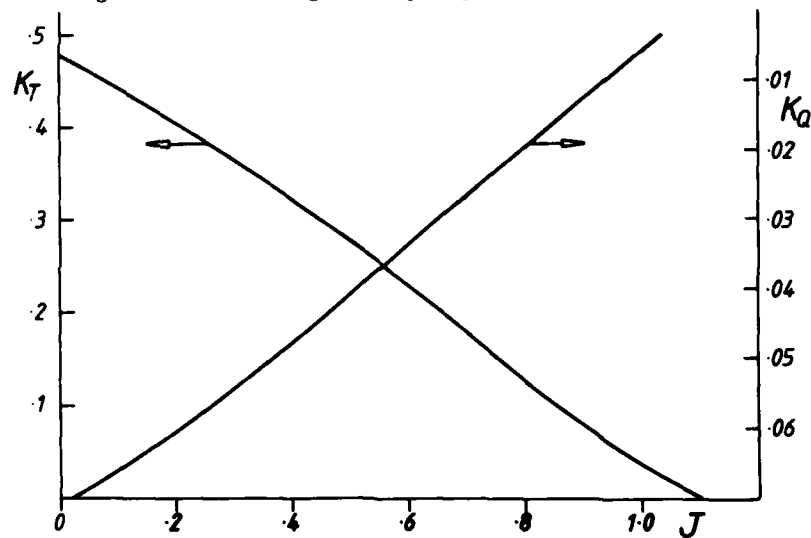


Figure 2 Simulation Propeller Characteristics

effect is both asymmetric and non-linear, giving a characteristic shown in figure 3. For low rates of sway this can be linearized and accounted for as a modification of the wake factor.

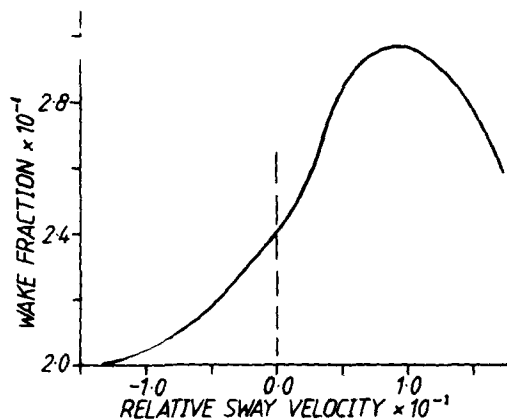


Figure 3. Wake fraction $W(r,v)$ plotted over relative sway velocity at the stern v_p/u for a Mariner type hull.

The net propeller inflow velocity V_A is thus given by:-

$$V_A = \frac{h}{2} \omega \exp(-zk) \cos(kx - \omega_a t) \cos \lambda + (1-W)u - W_v(v-xr)u \quad (14)$$

where W = wake factor
 u = ship's forward velocity
 v = ship's sway velocity
 r = ship's yaw rate
 W_v = wake factor versus sway velocity coefficient.

The advance coefficient, is then given by:-

$$J = V_A / nD \quad (15)$$

Where the terms have their usual meaning. The advance coefficient is then used in conjunction with lookup tables and interpolation routines to return values of torque and thrust coefficients. The thrust delivered by the propeller, less the thrust losses, is then used in the surge equation to determine the current speed of the vessel.

Autopilot and Steering Gear

The ship was a conventional PID autopilot, but for these studies the integral feedback was set to zero and only the proportional and derivative terms were used. The proportional signal is taken from

the heading error but includes a deadband of 0.5° , and the rate signal is taken from the yaw rate, with a deadband of $0.3^\circ/\text{second}$. This gives a demand rudder signal which actuates the rudder mechanism if it exceeds the deadband of 0.3° of rudder error. The error signal between the demand and actual rudder produces rudder acceleration, according to the rudder mass and against coulomb friction, but the magnitudes used in this part of the simulation are such that the acceleration is very rapid, as would be expected with hydraulically actuated steering gear. Although the acceleration rate is high the rudder velocity is immediately limited to a magnitude of $2^\circ/\text{second}$ to simulate the course keeping actions of the steering gear. The maximum rudder angle is $\pm 28^\circ$, and the programme also checks and corrects rudder velocity as necessary if the rudder angle approaches either of the end limits. In fact, in these course keeping trials the rudder limits were not approached.

It is also common practice in actual PID autopilot to include phase lag and phase lead networks so as to remove the inherent noise problems associated with derivative or rate feedback circuits. Such problems are not necessarily present in digital signal processing circuits, as will be seen below in the description of the scale model used for confirmation of the simulation results. For this reason frequency sensitive signal processing was not included in the autopilot simulation and the calculation of proportional and derivative gains was for that reason much simpler.

SIMULATION TEST PROCEDURE AND DATA ANALYSIS

Increased Fuel Cost Determination

The inclusion of the surge equation in the ship simulation gave a ship's speed that was determined by the propulsion machinery demand levels, the ship's environment and the simulated autopilot response to the external perturbations. For this reason the simulated ship's speed could not be predetermined exactly, but with experience could usually be made to fall within 10% or less of a standard speed of $11.3\text{m}/\text{second}$. The fuel cost results were made by comparison with a standard speed of $11.3\text{m}/\text{second}$ with no external perturbations and therefore no autopilot actions or disturbances to the propulsion plant. In order to make this comparison, it was necessary to find out what the fuel cost would have been if the ship had been running at the standard speed under perturbed conditions. This was done by obtaining two simulation runs for each setting of the autopilot, the only difference between the two being that the engine demand speeds were set to different levels to give two different ship speeds, both close to the standard speed. Because the simulation programmes were controlled by elapsed time, the results for each simulation run were first scaled to give results for a standard distance, in this case 5000m of travel, and then extrapolated between the two sets of results to give a single result for 5000m travelled at $11.3\text{m}/\text{second}$. The simulation programme not only logged the total mass flow of fuel to the engine, but also integrated the individual drag components in the surge equation. The final results therefore not only gave the amount of fuel used under perturbed conditions, as compared with a standard distance journey, but also gave the individual components of drag, i.e. hull, yaw and rudder, and it was therefore possible to see the effect of autopilot setting on each component. The hull drag component also provided a check since it should itself show very little variation for the normal range of autopilot settings, and in fact the

integrated hull drag terms were usually constant within 0.1%

The result of this part of the investigation was that it was found that the autopilot could be tuned to give a minimum fuel cost. This is illustrated in figures 4 and 5 for a fixed proportional gain against derivative gain, and also shown are the changes in the yaw and rudder drag components as percentages of the hull drag. Figure 4 is the curve for the ship at 24000 tons and figure 5 is for 20,000 tons. The zero disturbance fuel cost against which they were compared was for 20,000 tons. It will be seen that the increased weight shifts the minimum in the rudder drag and fuel cost curves to a slightly higher derivative gain. The effect of changing the proportional autopilot gain may be seen in figure 6 which gives the curves and several points for 20,000 tons for three different proportional gains. It is clear from these last three figures that for a minimum fuel cost at the required operating speed, the autopilot settings would be in the region of $K_D = 0.5$, $K_P = 8.0 + 10.0$. For a real ship these fuel cost graphs would not be available, but the ship's heading step response could be found. Step responses were also taken from the simulation, and two sets of graphs are shown in figures 7 and 8 for 20,000 tons. The step response at $K_P = 1.0$, which is not shown, was similar to figure 7, but gave a slower response. From the behaviour shown in these last two figures it is apparent that for optimum course keeping, i.e. quickest response with no overshoot, the autopilot should be set to $K_P = 1.4$, $K_D = 40$. Reference to the earlier fuel cost graphs will show that such an autopilot setting is likely to be very expensive in terms of fuel used.

Spectral Analysis of Rudder Position and Ship's Heading

The fuel cost graphs shown in figures 4 and 5 are typical of those that may be obtained under conditions in which the ship is travelling into the waves. The fuel cost is increased by the disturbances acting on the ship for all autopilot settings, although it will be seen that the increase in drag due to the rudder activity and yawing of the ship does not fully account for the full increase in fuel used. Later studies with only the propulsion system in the simulation gave results which indicated that the propulsion losses by themselves could account for a 10% increase in fuel cost, under the sea conditions used to produce figures 4 and 5.

The problem of obtaining the minimum fuel cost by tuning the autopilot, remains, as with the cost function approach, that of obtaining the correct balance of rudder and yaw activity to achieve the minimum net increase in drag. The relative drag due to the rudder and yaw shown in the figures above gives an indication as to how this may be done. The left hand region of the fuel cost graphs, below the optimum derivative gain, shows a region in which both yaw and rudder drag are high. The rudder response is slow in this region and the ship may only be controlled by excessive rudder angles. In this ship simulation the disturbing forces are added in as wave forces on the hull, so that increasing the derivative gain increases the sensitivity to yaw rate and with sufficient gain the rudder begins to correct the heading errors as they are generated by the wave forces. It is believed that if the disturbances were added to the simulation as a series of random heading errors, this would not be the case.

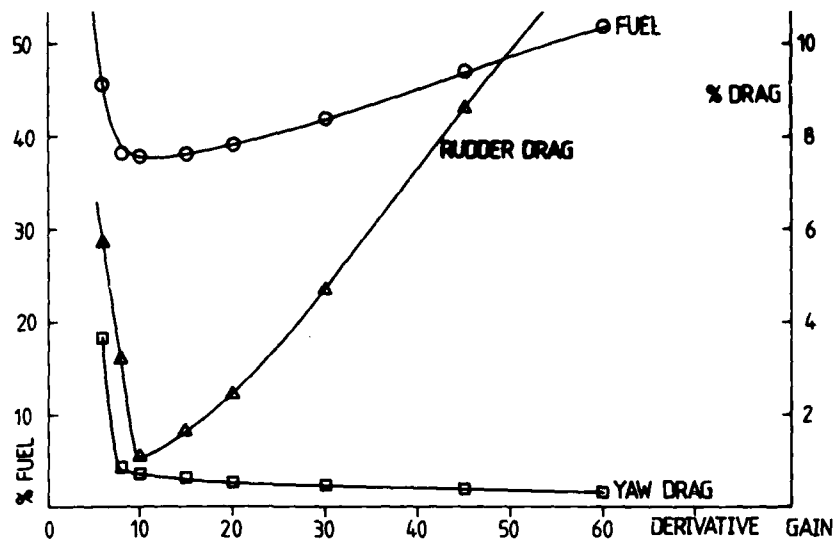


Figure 4 Simulated Drag Components and Excess Fuel for $m = 24,000$, $\omega = 0.2$, $\gamma = 0.75$, $K_p = 0.5$

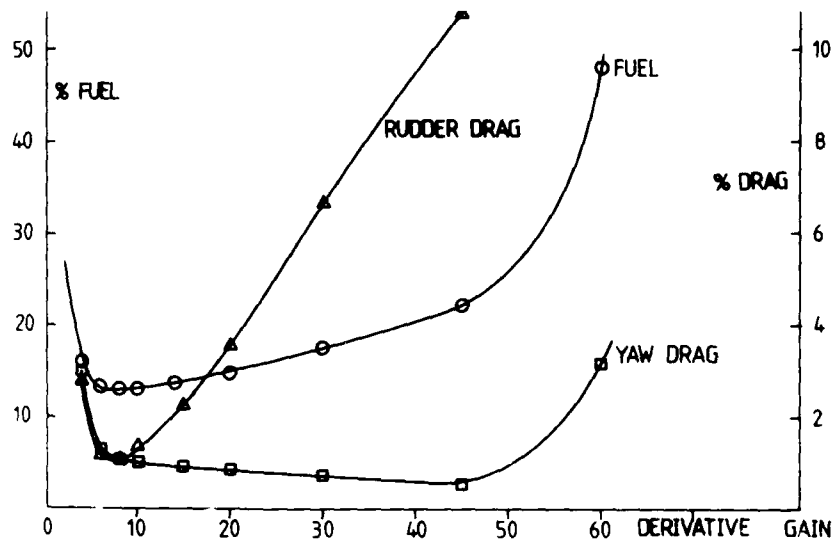


Figure 5 Simulated Drag Components and Excess Fuel for $m = 20,000$, $\omega = 0.2$, $\gamma = 0.75$, $K_p = 0.5$

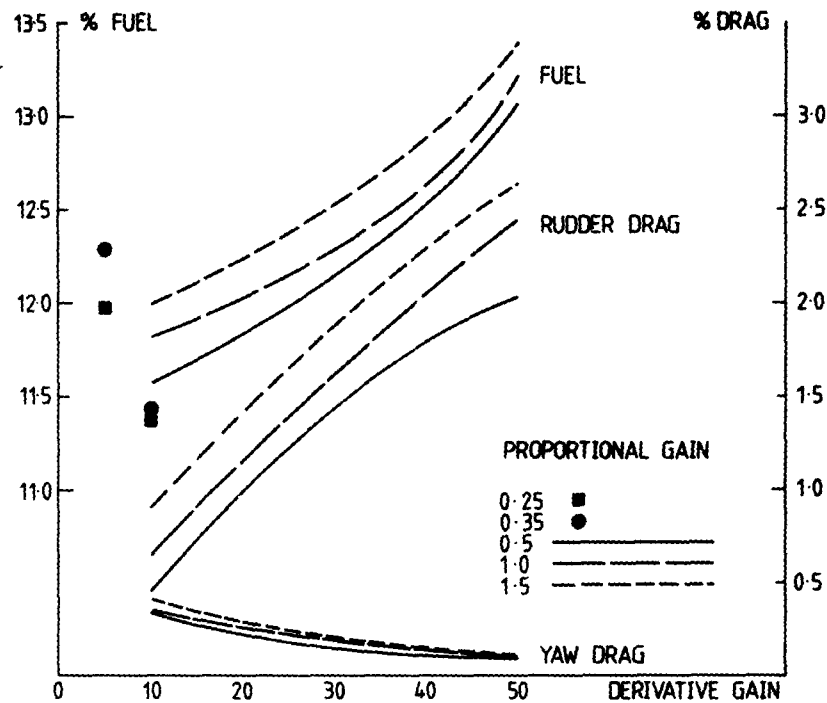


Figure 6. Changes in excess fuel, and rudder and yaw drag for a fixed mass at various proportional gains.

The increase in derivative gain brings about a minimum in the net drag and therefore a minimum in the fuel cost graph. If the derivative gain is increased beyond this point the rudder drag increases and the yaw drag decreases. This last fact is well known, but the magnitude of the change in the two components is completely different. There is very little reduction in the yaw drag, and therefore yaw rate, but a very sharp rise in rudder drag, and therefore in the average rudder deviation. Since there is so little reduction in the gross yawing of the ship in the high derivative gain area, and so much rudder activity, the question may be asked as to just what the rudder is responding to. The answer to this question was eventually furnished by spectral analysis of the rudder position and ship's heading records from the simulation.

The power spectral density plot of a time series is conventionally provided by first forming the autocorrelation of the time series, and then transferring this time plot to the frequency domain by Fourier analysis. The autocorrelation function of a random process may be defined as:-

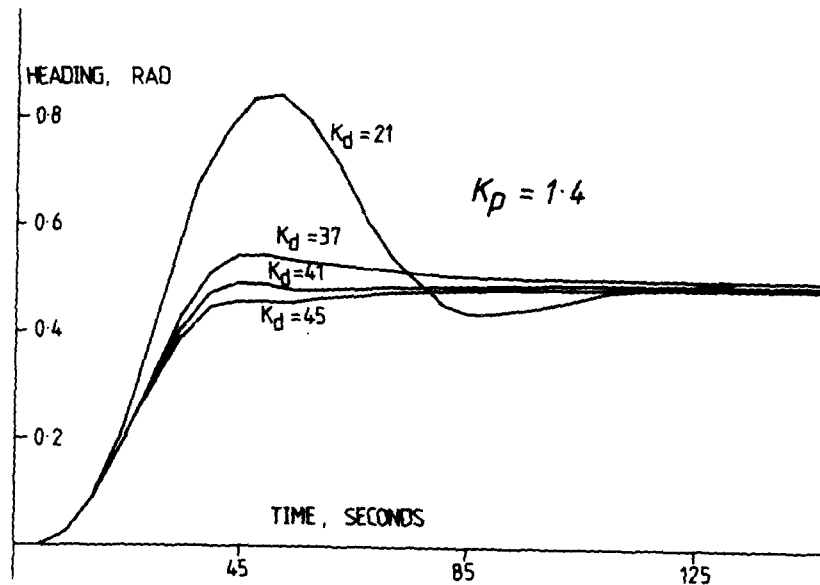


Figure 7 Simulated Ship's Heading Step Response for $K_p = 1.4$

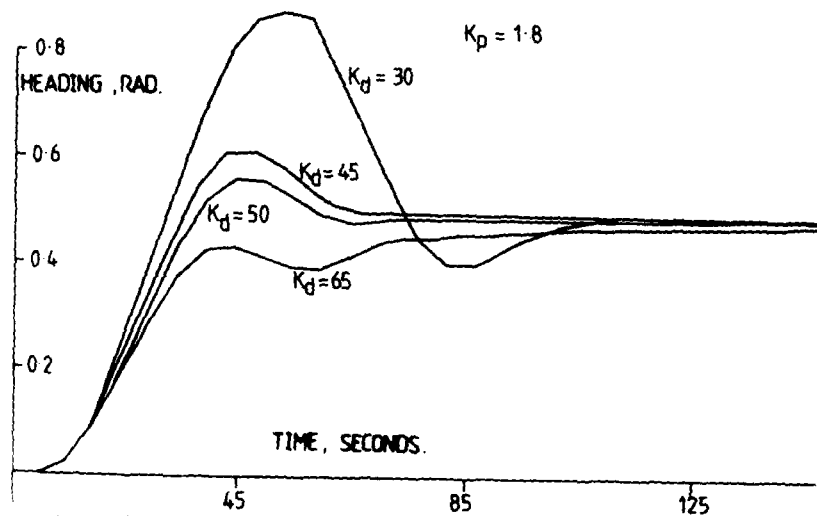


Figure 8 Simulated Ship's Heading Step Response for $K_p = 1.8$

$$R_{xx}(\tau) = \iint x_1 \cdot x_2 \cdot f(x_1, x_2; t_1, t_2) dx_1 dx_2 \quad (16)$$

but for a stationary ergodic time series the analysis may be done on a single record of the process to achieve the same function. Therefore,

$$R_{xx}(\tau) = \frac{1}{2\pi} \int_{-\pi}^{\pi} X(t) \cdot X(t+\tau) dt \quad (17)$$

For a discrete stationary ergodic time series this becomes:-

$$R_{xx}(\tau) = \frac{1}{N} \sum_{n=1}^N x(nt) \cdot X(nt+\tau) \quad (18)$$

That is, to form the autocorrelation function of a time series for a given fixed interval τ , one forms the sum of the product of all values after an interval τ . In equation 18, t is the sampling interval and N is sufficiently large to give enough samples for a representative average. To obtain any insight into the nature of a time series it is necessary to form the autocorrelation function for a sequence of intervals τ (sometimes called lags). The autocorrelation function of a pure sine wave is itself a sine wave, so that periodicity in the original time series may be detected by examining the autocorrelation function. The power spectrum may be derived by a fourier transform of the autocorrelation function, so that for a continuous time series:-

$$S(\omega) = \int_{-\infty}^{\infty} e^{-j\omega\tau} R(\tau) d\tau \quad (19)$$

For a discrete process this becomes

$$S(\omega_1) = \sum_{n=1}^N R(n\Delta t) [\cos(\omega_1 n\Delta t) + \sin(\omega_1 n\Delta t)] \quad (20)$$

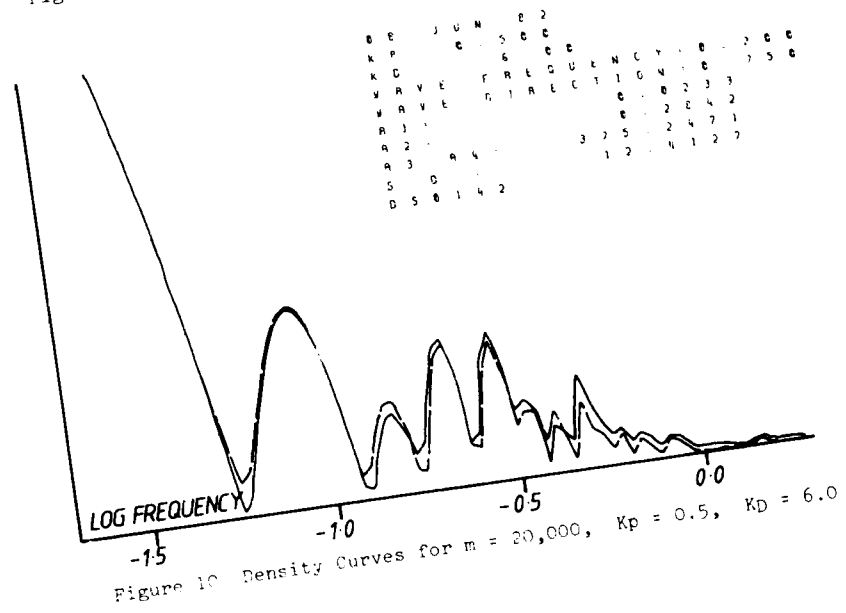
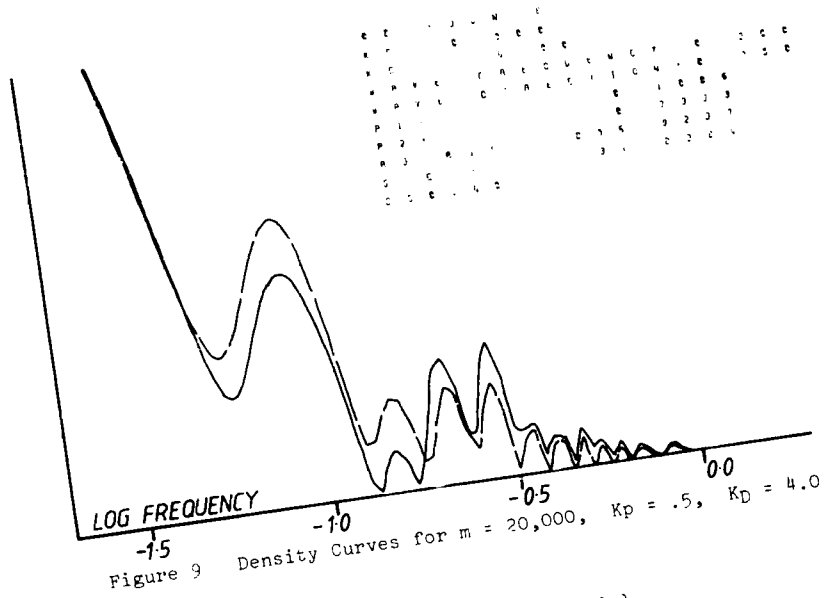
Again this forms the spectral density for one point, in this case one frequency ω_1 , and it is again usual to form the spectral density function for a range of frequencies. In the case of digital processing the frequencies themselves would also be discrete, and the interval Δt would correspond to the finite steps in the autocorrelation function.

Additional programming was written into the original ship simulation to form the autocorrelation functions of the rudder position and of the ship's heading, and these two functions were then available along with the fuel cost information. These functions were then transformed to the spectral densities and graphs produced. The sampling interval in the simulation, which corresponds to t in equation 18, was 1 second and functions were formed for values of τ

from 1 to 100 seconds. When transformed this gives a useable frequency range from 3 rad/sec to 0.03 rad/second for the power spectral density. The information was gathered from fuel cost simulation runs, so that in equation 18 the figure N is 490 approximately. The spectral density functions produced in this way did show effects which correlated with the earlier fuel cost and drag component graphs, but the limited span of the autocorrelation functions produced an apparent very low frequency component and false harmonic effects. At a later stage a method was found of eliminating this effect. Another function performed by the Fourier transform programming was to integrate the total area under both the rudder and heading spectral density curves. For a continuous function this gives a measure of the total power involved over the frequency range considered. Because the original heading and rudder motions were of different magnitudes the spectral density curves were originally also of different magnitudes. In order to show both curves on the same graph they were rescaled according to the very low frequency component, but this does not effect the calculated areas under the curves.

The spectral density curves so produced, for the sea conditions used to produce the fuel cost graph shown in figure 5, are shown in figures 9 to 16, and a similar set of curves were obtained for conditions given in figure 4. The density curves are plotted against the log of the frequency, and the areas under the spectral density curves are given as A1 for the rudder and A2 for ship's heading. Several effects may be seen in this set of spectral density curves, which were also present in those for the ship with a larger mass whose fuel cost graph is shown in figure 4. For the lowest derivative gain, i.e. $K_D = 4.0$, the two curves show a distinct separation in the low frequency region, which would be the frequency in this case of the ship/autopilot system, i.e. 0.08 rad/sec, period = 80 seconds. There is also very little power being dissipated at the higher frequencies, particularly the wave encounter frequency, which in this case is approximately 0.4 rad/sec ($\log \omega_e = -0.4$). When derivative gain is increased, as for figure 10, the two spectral density curves show much higher correlation, that is, the rudder activity corresponds more in frequency and magnitude to the activity of the hull. Examination of figure 5 will show that this autopilot setting was very close to giving the minimum fuel cost for this set of sea conditions, and ships heading. The areas under the two power curves are also very much less for figure 10 than they were in figure 9, indicating a corresponding reduction in the original rudder and yaw activity. The higher frequency activity in figure 10 appears to be higher than that for figure 9, but these graphs have been scaled independently, and only the areas A1 and A2 are indicative.

The further spectral density curves are for the higher derivative gains shown in figure 5, and several changes can be seen to take place. Correlation between the two curves is less in figures 11 and 12, but the separation is in the mid-frequency range. The area A2 is steadily reduced up to figure 15, which corresponds to the minimum yaw drag as expected, and then increases dramatically, as does the yaw drag curve. The area A1 however increases steadily from figure 10 onwards, and also shows a very large increase in figure 16. This again corresponds to the progress of the rudder drag curve. These changes in the power density curve areas could be predicted from the original drag curves. The changing shape of the curves as the derivative gain is increased reveals the source of the increased rudder activity. From figures 12 to 15 a large peak



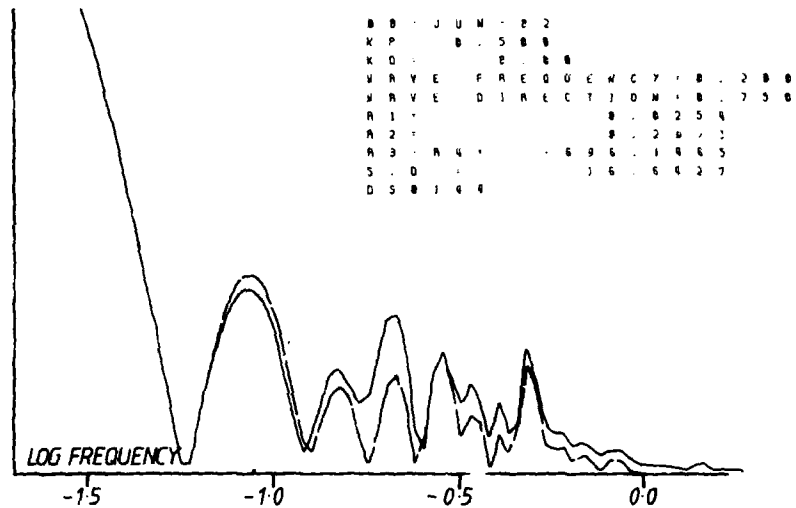


Figure 11 Density Curves for $m = 20,000$, $K_p = 0.5$, $K_p = 8.0$

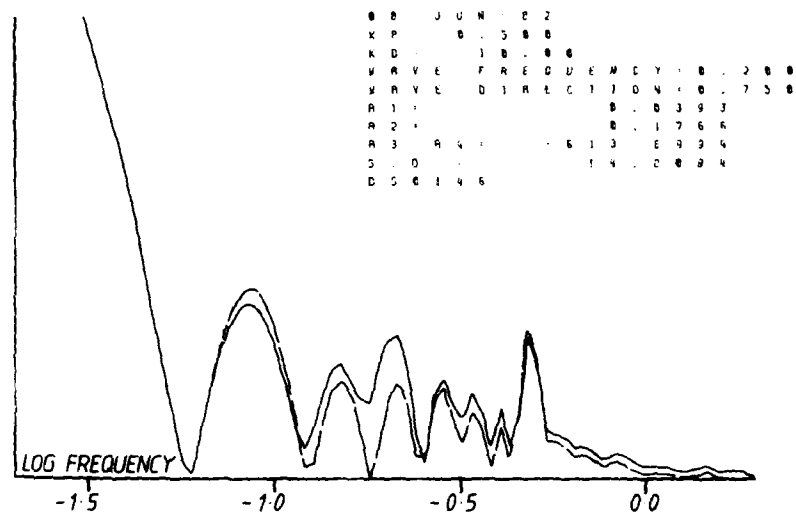


Figure 12 Density Curves for $m = 20,000$, $K_p = 0.5$, $K_p = 10.0$

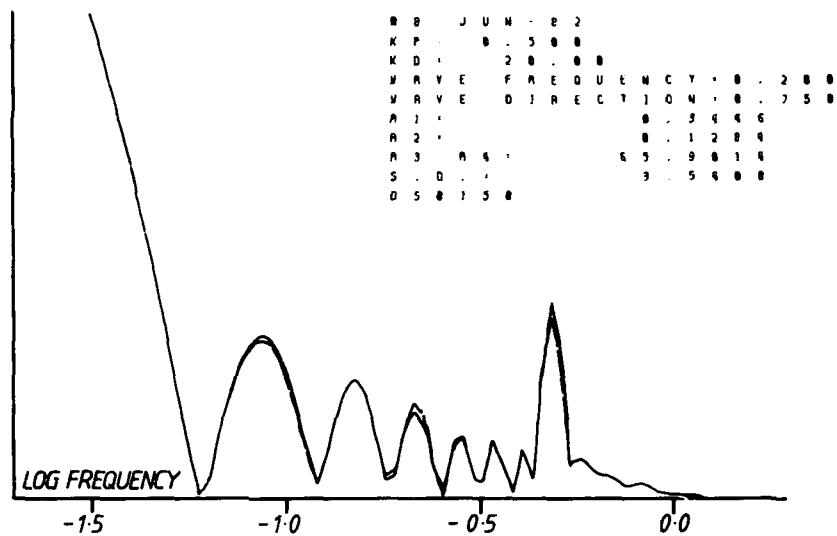


Figure 13 Density Curves for $m = 20,000$, $K_p = 0.5$, $K_D = 20.0$

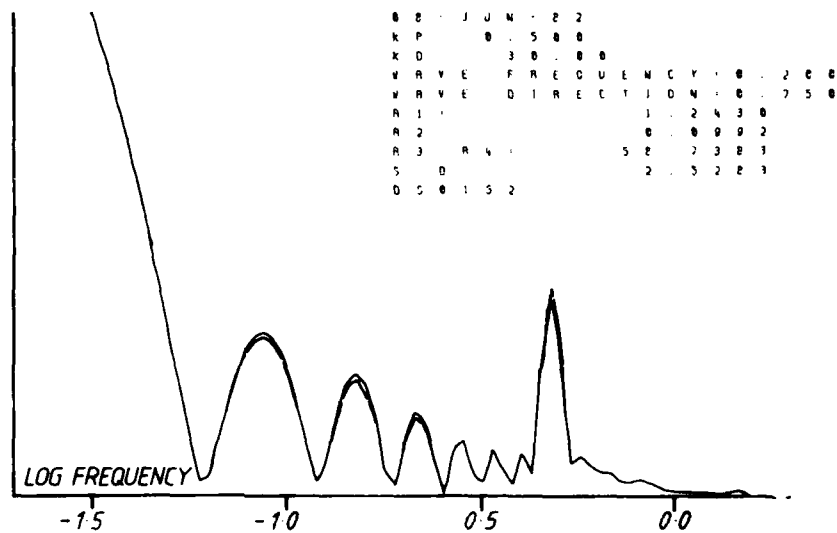
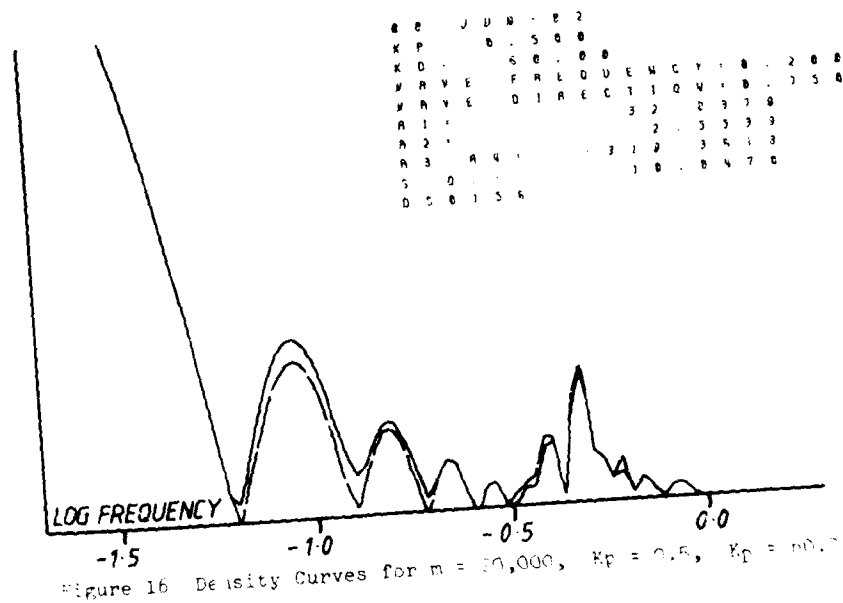
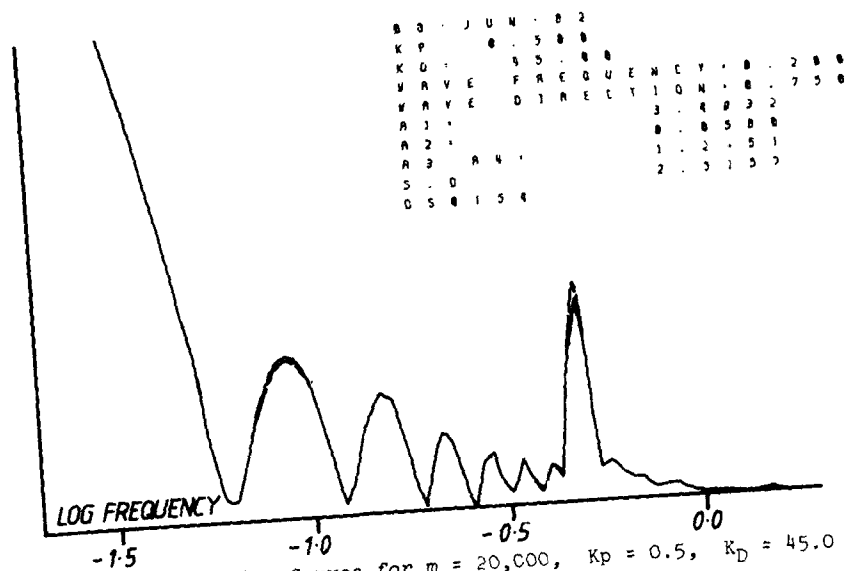


Figure 14 Density Curves for $m = 20,000$, $K_p = 0.5$, $K_D = 30.0$



develops which is at the wave encounter frequency, and it is the acute sensitivity of the autopilot to such frequencies which causes the high rudder activity and increased drag. For the very highest gain, figure 16, the curves show separation at the low frequency end of the spectrum, indicating instability. The spectral density curves for a ship's mass of 24,000 tons (figure 4) showed a similar pattern, with the maximum correlation between the two curves occurring at $K_D = 10$, and they did not show any dramatic increase in area, or curved separation at $K_D = 60$.

For this ship simulation the proportional gain, when varied over a reasonable range, had less effect in changing the fuel cost than did the derivative gain, as shown by the curves for three different proportional gains shown in figure 6. Changes in the ship's behaviour brought about by changes in the proportional autopilot gain also manifested themselves in the spectral density curves, with similar effects to the changes in derivative gain. This is illustrated by figures 17 and 18, which are the spectral density curves for the same sea conditions and derivative gain, but for different proportional gains as shown. Again the correlation between the rudder and heading curves is increased by a reduction in the proportional gain from 1.0 to 0.5, particularly in the low frequency range of yaw activity. There is also a reduction in the integrals of both power curves (areas A1 and A2) in moving to the lower gain. Since these areas reflect the degree of rudder and yaw activity this obviously implies a reduction in both of these drag components and therefore also in the fuel cost.

Spectral analysis curves were done for many different wave frequencies and wave encounter angles over ranges of both proportional and derivative autopilot gains. The effects illustrated in the figures were present to a greater or lesser degree in all of the sets of curves taken. These effects are concerned with changes in the spectral density curves with changes in autopilot setting, in terms of cross correlation between rudder and heading, and the power dissipation in the rudder and yaw activity; rather than with an absolute value at some setting of the autopilot. The ranges of autopilot gains included those for which the response of the ship to step changes in demand heading had been found, as shown in figure 7. These spectral density curves were taken from the behaviour of the ship when underway on a fixed course, with various wave conditions. Figures 19, 20 and 21 give the spectral density curves for a proportional autopilot gain of 1.4 and derivative gains of 20, 40 and 60. The step response indicated that $K_P = 1.4$, $K_D = 40$ gave an approximately critically damped response and the most rapid change of course. The curves given in these figures are for a following sea, and the wave encounter frequency is approximately 0.16 rad/sec. It can be seen from the figures that similar effects occur in these graphs as in the previous figures, that is correlation is increased and the areas under the curves are decreased in going from $K_D = 20$ to $K_D = 40$, but that further increase in K_D does not improve the correlation but does increase the rudder activity, and this activity shifts towards the wave encounter frequency ($\log 0.16 = 0.8$). It should be mentioned that the following sea condition was a 'worst case' so far as changes in the spectral density curves were concerned.

From the above information it may be proposed that spectral analysis of the rudder and the hull activity could be used to tune the autopilot to give a minimum of activity and thus a minimum in the

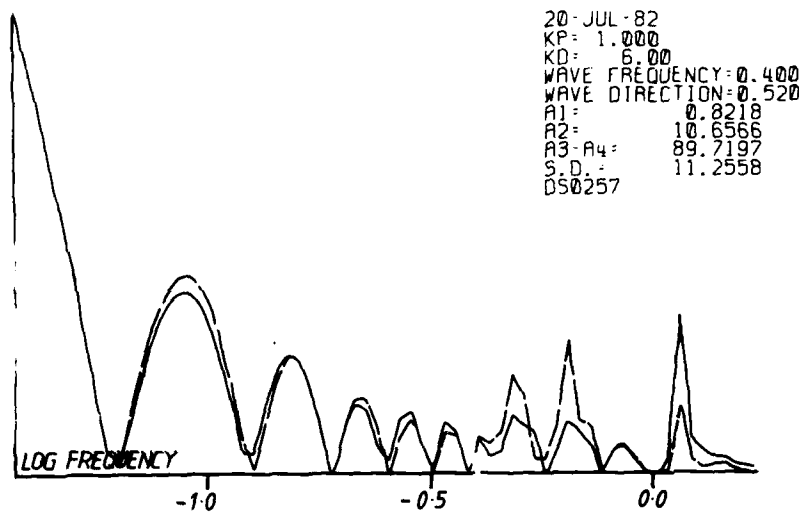


Figure 17 Density Curves for $m = 20,000$, $K_p = 1.0$, $K_D = 6.0$

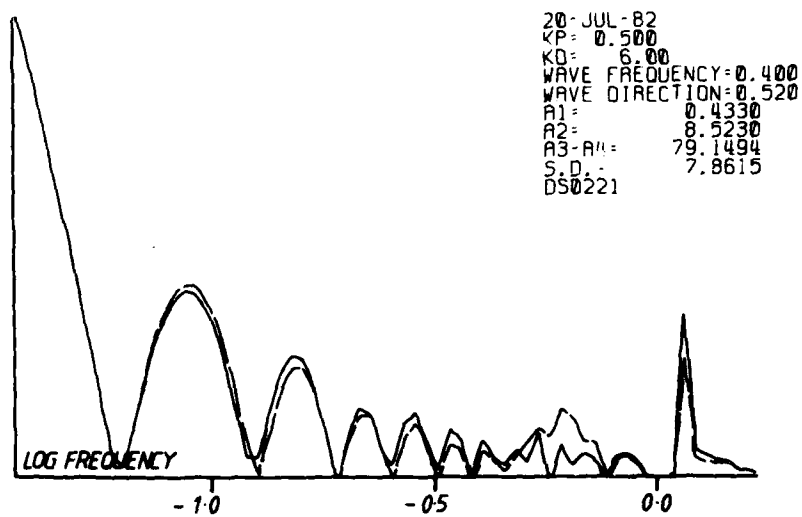


Figure 18 Density Curves for $m = 20,000$, $K_p = 0.5$, $K_D = 6.0$

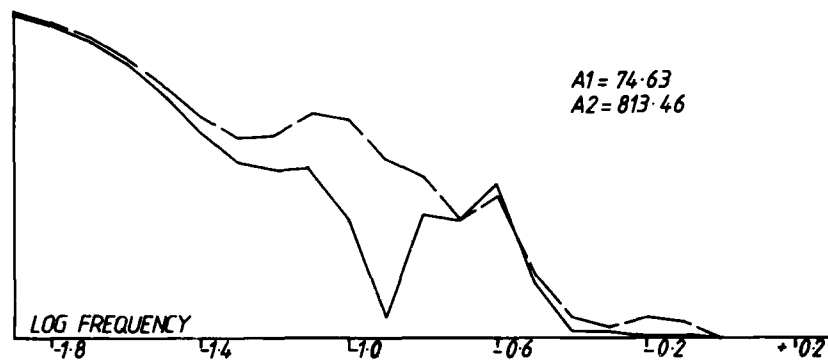


Figure 19 Density Curves for $m = 20,000$, $K_p = 1.4$, $K_D = 20.0$
Following Sea

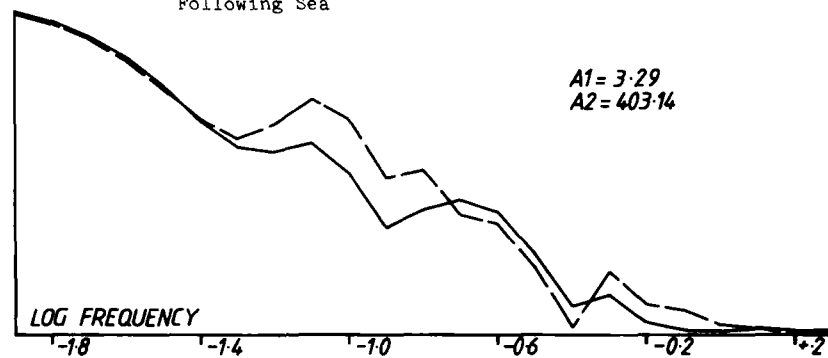


Figure 20 Density Curves for $m = 20,000$, $K_p = 1.4$, $K_D = 40.0$
Following Sea

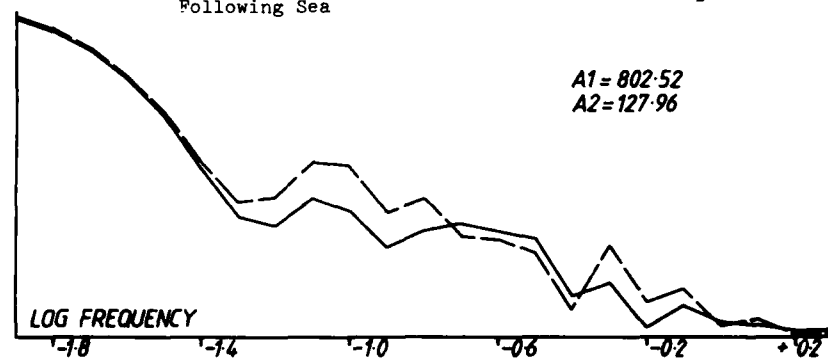


Figure 21 Density Curves for $m = 20,000$, $K_p = 1.4$, $K_D = 60.0$
Following Sea

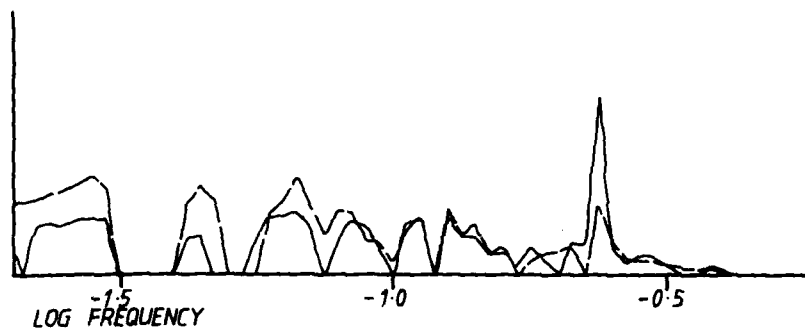


Figure 22 Density Curves for $m = 15,000$, $K_p = 0.5$, $K_D = 6.0$

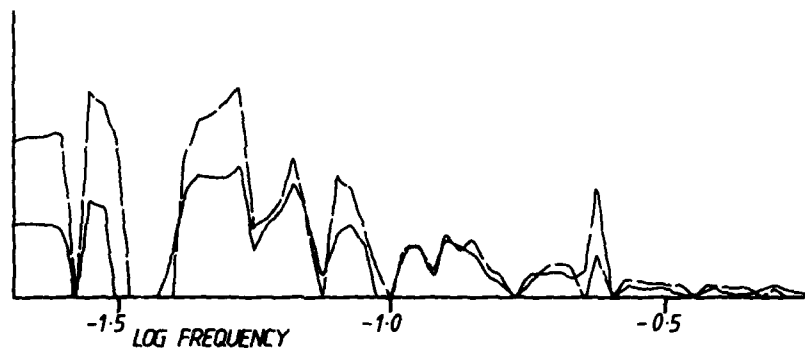


Figure 23 Density Curves for $m = 20,000$, $K_p = 0.5$, $K_D = 6.0$

drag and excess fuel cost. Fundamental to this is the fact that the ship response in yaw, for most sea conditions, is at a far lower frequency than the dominant frequency of the disturbing random signal. Furthermore, it is the yaw of the ship which is responsible for a fraction of the excess drag generated, and the purpose of the rudder is to minimise this drag without excessive rudder activity. If the autopilot tuning is incorrect it has either poor control of the ship, or responds to the disturbing frequency of the waves. The tuning is governed by the relative magnitude of the derivative and proportional gains of the autopilot, and for a given proportional gain a minimum drag is achieved by reducing the derivative gain to a point just prior to an underdamped response. Drag may be further reduced by reducing the proportional gain from a high level, (and retuning with the derivative gain) so long as such a reduction reduces the integral of the power spectral density curves. In the simulation tests it was found that a minimum proportional gain did exist, below which the yawing activity of the ship increased and the correlation between the spectral density curves rapidly decreased, i.e. the rudder loses control of the ship.

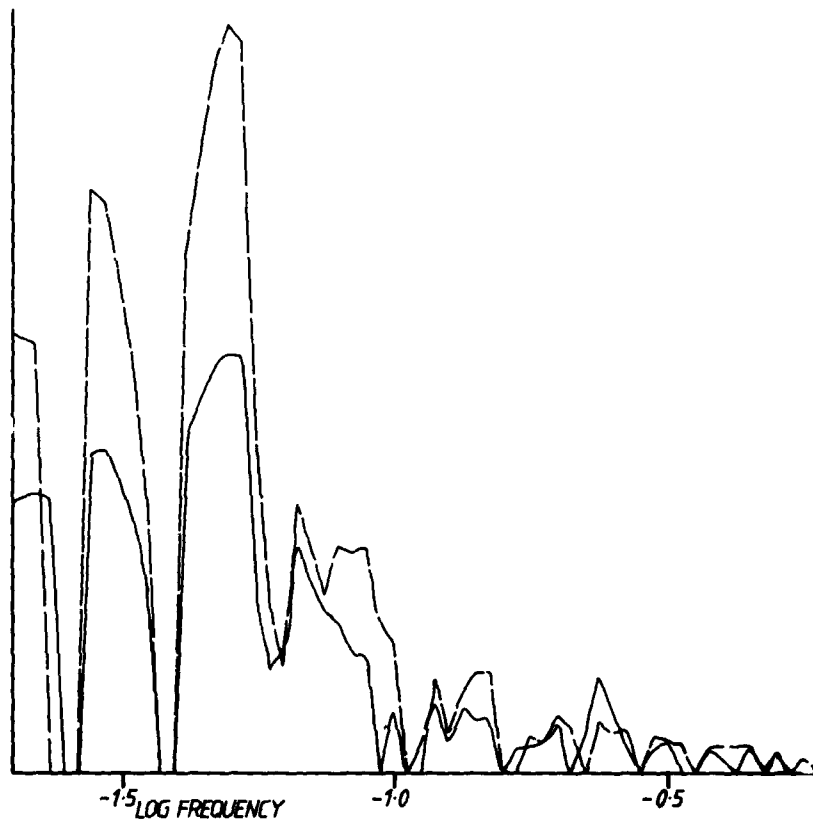


Figure 24 Density Curves for $m = 25,000$, $K_p = 0.5$, $K_D = 6.0$

There are two major factors which change the behaviour of the ship and would therefore be expected to change the autopilot tuning for maximum fuel economy. One of these factors is the condition of the sea, with particular reference to the ship's heading relative to the wave direction. In this respect spectral analysis would appear to provide a reliable method of identifying the correct autopilot tuning. The second major factor which influences the behaviour of a ship is its loading condition, and examination of figures 4 and 5, which are for ship masses of 24,000t and 20,000t, shows that the ship's loading influences optimum tuning. To be of any practical use in determining minimum fuel cost settings of the autopilot, the

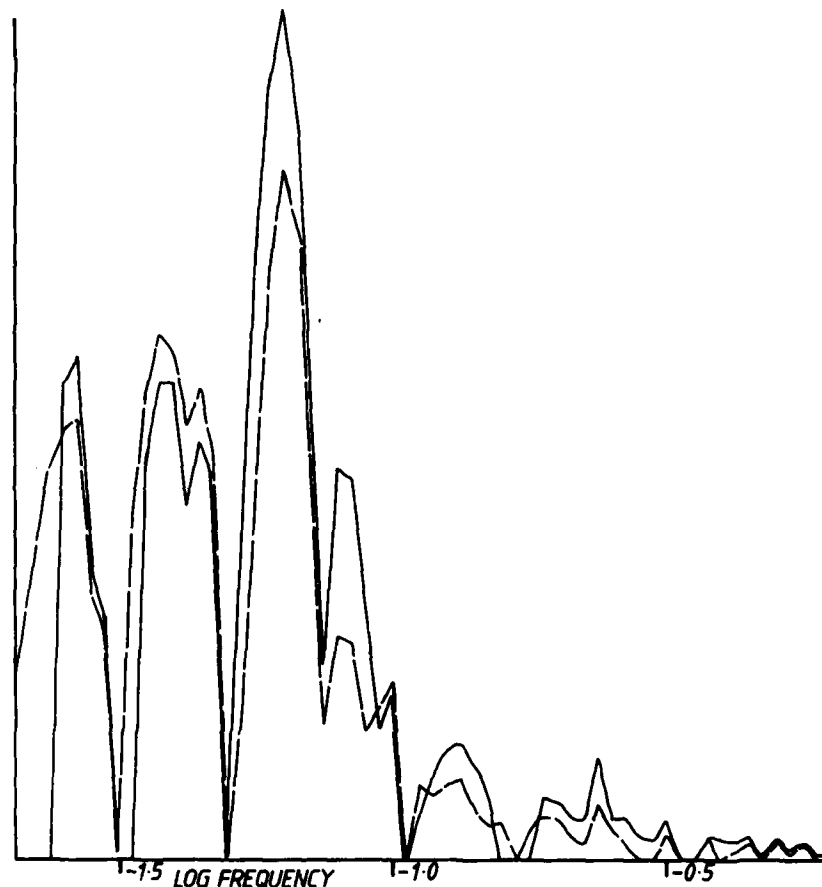


Figure 25 Density Curves for $m = 25,000$, $K_p = 1.0$, $K_D = 8.0$

Method of spectral analysis must also be sensitive to changes in ship loading. At the time that this effect was being investigated changes were also made in the method of calculation, which removed from the spectral analysis curves the false very low frequency component and periodicity associated with the limited time span of the autocorrelation functions. The curves shown in figures 22 to 24 are of the later type, and convey more information on ship behaviour than did the earlier kind. These curves are all for the same sea conditions, ship's heading and autopilot settings, but are for ship masses of 15,000t, 20,000t and 25,000t. There is a significant

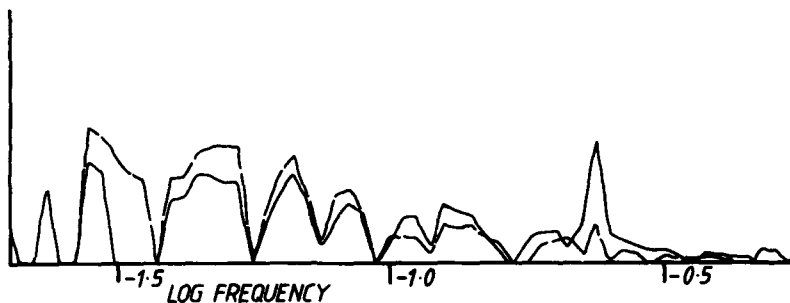


Figure 26 Density Curves for $m = 25,000$, $K_p = 0.5$, $K_D = 10.0$

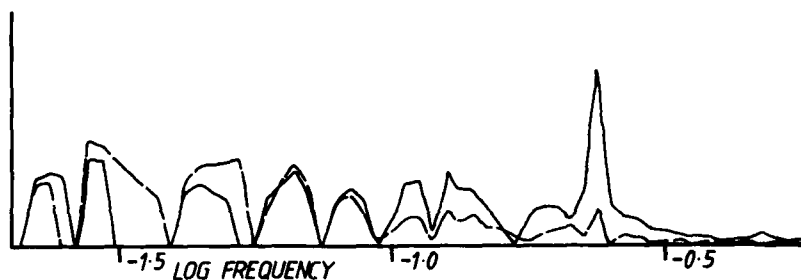


Figure 27 Density Curves for $m = 25,000$, $K_p = 0.5$, $K_D = 14.0$

change in the curves at the lower end, in the region of the yawing frequency of the vessel. According to the hypothesis figure 24 shows a ship which is significantly underdamped, and figure 4 indicates that the fuel cost of this autopilot setting and ship's mass would be high. To counteract this the autopilot may be retuned, but for a real ship, how much to change each of the autopilot gains may be in doubt. Figure 4 shows that the derivative gain should be increased to approximately 10, but this information would not be available. The spectral density curves of the rudder position and ship's heading do however indicate the new tuning point. Figure 25, for $K_p = 1.0$ $K_D = 8.0$ shows a worse condition, whilst figures 26 and 27 indicate that the proportional gain may remain unaltered and the derivative gain should be increased to between 10 and 14 seconds. This is in agreement with the fuel cost graph (for a slightly lower mass) shown in figure 4.

SCALE MODEL EXPERIMENTS

circuitry for this used a clock rate of 75Hz, and was capable of supplying a smoothed derivative signal from a 5V amplitude sine wave from 0.001Hz to 10Hz. The phase shift, which increases for increasing input frequency, was approximately 60° at 10Hz, and negligibly small at 0.001Hz. A digital circuit was chosen for the derivative channel because the proportion of derivative feedback from the channel is independent of frequency, which would not be the case for practical analogue circuits. Both the proportional and derivative signals were combined with the rudder position feedback signal in a summing amplifier, the output of which fed the rudder motor power amplifier.

Much effort was put to making the final model conform to the ideal scaled parameters from the actual ship. The hull shape was achieved by scaling the original body lines of the ship, so that the final hull shape conformed closely to that of the original, including the rudder and propeller locations. There was a discrepancy in the size of the propeller which was only 50mm diameter and should have been 55mm, but it was given the correct blade area ratio and pitch. The other important parameters of the model are shown in table two. Although the moment of inertia of the actual model was not measured, it is unlikely to be very much in error because the model mass is approximately correct, and the instrumentation and batteries etc which were responsible for most of the ship's mass, were distributed uniformly inside the model.

Table 2. Model and Ship Parameters

	Actual Ship	Ideal Model	Actual Model
Speed	18.4 knots	0.95 m/s	0.63 m/s
Mass	1×10^6 kg	16.95 kg	18.75 kg
Length	152 m	1.58 m	1.58 m
Inertia	2.13×10^{10} kgm ²	1.07 kgm ²	-
Roll period			1.5 seconds

Model Trials

The model was tested as a free sailing vessel under its own autopilot control on both open water and in an enclosed towing tank. The open water trials provided experience in using the model and its instrumentation, but because of the magnified effects of even slight breezes these tests could not be used as formal experiments. The trials in the enclosed towing tank were of course free from random wind effects, but it should be said that the wave disturbances on the model were not closely controlled. The wave conditions under which the tests were done were approximately constant, but when the effects of scaling to this degree are taken into account, the variation was sufficient to make the speed results from these tests unreliable. Added to this there was unfortunately a slight variation in the propulsive power delivered by the propeller drive motor, due to changes in battery voltage. It had been hoped that propulsion losses could be assessed from the model's speed under constant conditions, but since this was not so the alternative is to assess the net drag from the records of rudder position and ship's heading.

The scale model trials consisted essentially of a large number

of timed free sailing runs of the model under autopilot control. During the earliest trials the workable ranges of autopilot gains were found, and the electronics altered as necessary to give the maximum number of autopilot fixed gain setting within these workable ranges. Subsequent trials were then done for all of the combinations of proportional and derivative gain for which the autopilot was capable of maintaining a course. For these tests the speed of the vessel was determined by measuring the transit time over a fixed distance, usually 40m, and the miniature tape recorder was used to gain a record of the rudder position and ship's heading. On completion of these tests, the recordings were analysed using spectral analysis.

Data Analysis

The tape recordings obtained from the model were handled in two ways. The first process was to transfer the tapes to a continuous graphical recording so that the timed autopilot runs could be identified. The recordings were also played back on an instrumentation amplifier, and read into a digital computer, at a sampling rate of 25 Hz, as a continuous record of the ship's activity. Using a graphics programme this continuous record was then edited into a separate record for each model test at different autopilot settings. After this editing process each of the data files produced, containing a record of rudder position and ship's heading, represented approximately 40 seconds of sailing time for the model.

A spectral density plot was then obtained of the rudder position and heading for each of the data files, and plotted as a hard copy. This was done without the intervening step of calculating the autocorrelation function, since it can be shown that:-

$$X(\omega) = \int_{-T}^{+T} X(t) e^{i\omega t} dt \quad (21)$$

If T is sufficiently large in comparison with the time period of the frequency ω . In the spectral density curves generated from these records the artificial low frequency component produced by the finite length of the record occurs below the power density peaks for yaw and rudder activity. For a discrete signal, the above transformation becomes:-

$$X(\omega) = \sum_{n=1}^N X(n\Delta t) \exp(i\omega n\Delta t) \quad (22)$$

where Δt is the sampling interval. In practice the exponential is formed from a sum of sines and cosines. The resulting spectral density plots are given below, plotted on a log frequency scale, with frequency increasing from left to right. The actual figures have been omitted from the scale because there was some doubt as to the sampling frequency of the computer data input programme, since it did not take account of the time taken to read in the data. In the actual model tests however, it was observed that the yaw period of the model under autopilot control was between 5 and 10 seconds, depending on autopilot setting. The figures given for K_p and K_D in

the spectral density plots refer to the autopilot gain switch positions rather than the actual gain for each channel. The gains may be related to the switch positions according to table 3.

Table 3. Proportional and Derivative Autopilot Gains

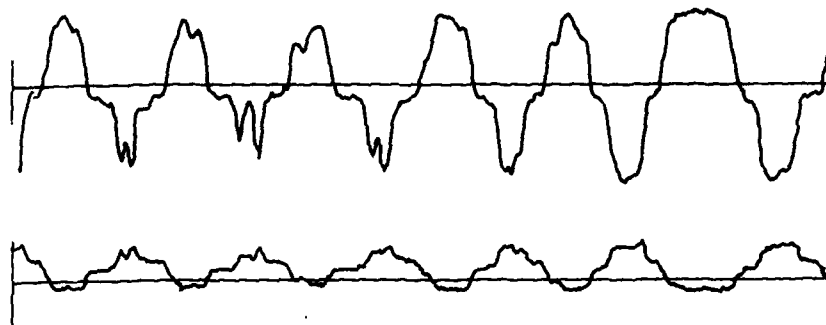
Switch Position	Proportional Gain K_p	Derivative Gain K_D
0	0	0 sec
1	0.82	0.95
2	1.09	1.24
3	1.63	1.86
4	2.17	2.50
5	3.26	3.70
6	4.61	4.80
7	6.78	7.70
8	9.05	10.2

Therefore, for the density curves marked $K_p = 6.0$, $K_D = 2$, the actual values of the two gains were 4.61 and 1.24 sec.

Test Results

Several of the records of ship's heading and rudder position, together with the corresponding spectral density curves are given in the following figures. In these diagrams the ship's heading is given by the upper record line, and the heading spectral density is shown by the solid line. Many of the ship's heading curves show a point of inflection at the midway position, which is in fact mostly due to a nonlinearity in the gyroscope output. However, since it is present in the electrical output of the gyro, it does influence the rudder and also tends to introduce high frequency harmonics into the spectral densities.

The model records and density curves are shown in the following figures for autopilot settings (K_p and K_D switch positions) of 6, 0; 7, 0; 6, 1 and 7, 1. An examination of the rudder and heading records shows that 7, 0 is a better setting than 6, 0, since both rudder and heading curves are smoother for 7, 0 than for the latter, and evidently the higher proportional gain removes the yawing influence of individual waves. A setting of 7, 0 is also more optimum than a setting of 6, 1; since it can be seen from the traces that the increase in derivative gain, from 6, 0 to 6, 1 increases the magnitude of both the rudder and yaw motions, and although it does reduce the perturbations from the yawing motion, the high frequency content is increased. For a gain setting of 7, 1 the ship's yaw frequency is higher than for any of the other three settings, and the rudder movements are large, which would indicate the highest added drag components, and for a real ship, the highest fuel cost. If the spectral density curves are examined for these runs it will be seen that the correlation between the two curves is highest for the 7, 0 setting, closely followed by the 6, 0 setting which has more higher frequency yaw components and therefore should also be subject to higher drag. Correlation between the two curves is worse for the 6, 1 and 7, 1 settings, and deteriorated as the derivative gain was increased, although in this case the rudder activity increased and yaw activity decreased at the lower end of the spectrum. In retrospect, having examined the spectral density curves, it is probable that the deriva-



K F 5
 K D 0
 RUN NUMBER 1
 S P L E D 0.631
 H A 1 R B
 2 4 - M A Y - 2 4

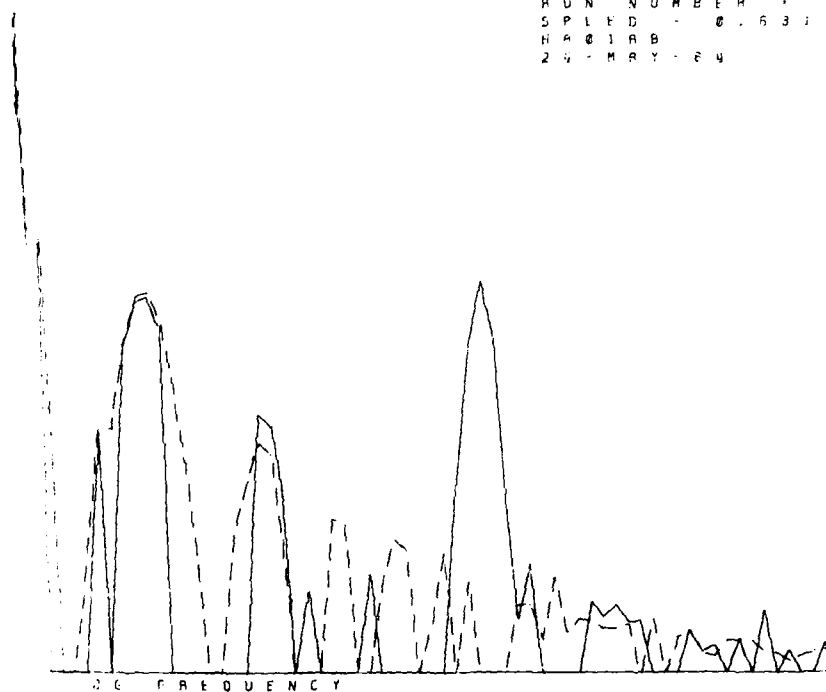


Figure 28 Heading (top) and Rudder Records and Spectral
 Density Curves from Scale Model. $K_p = 4.61$, $K_D = 0$

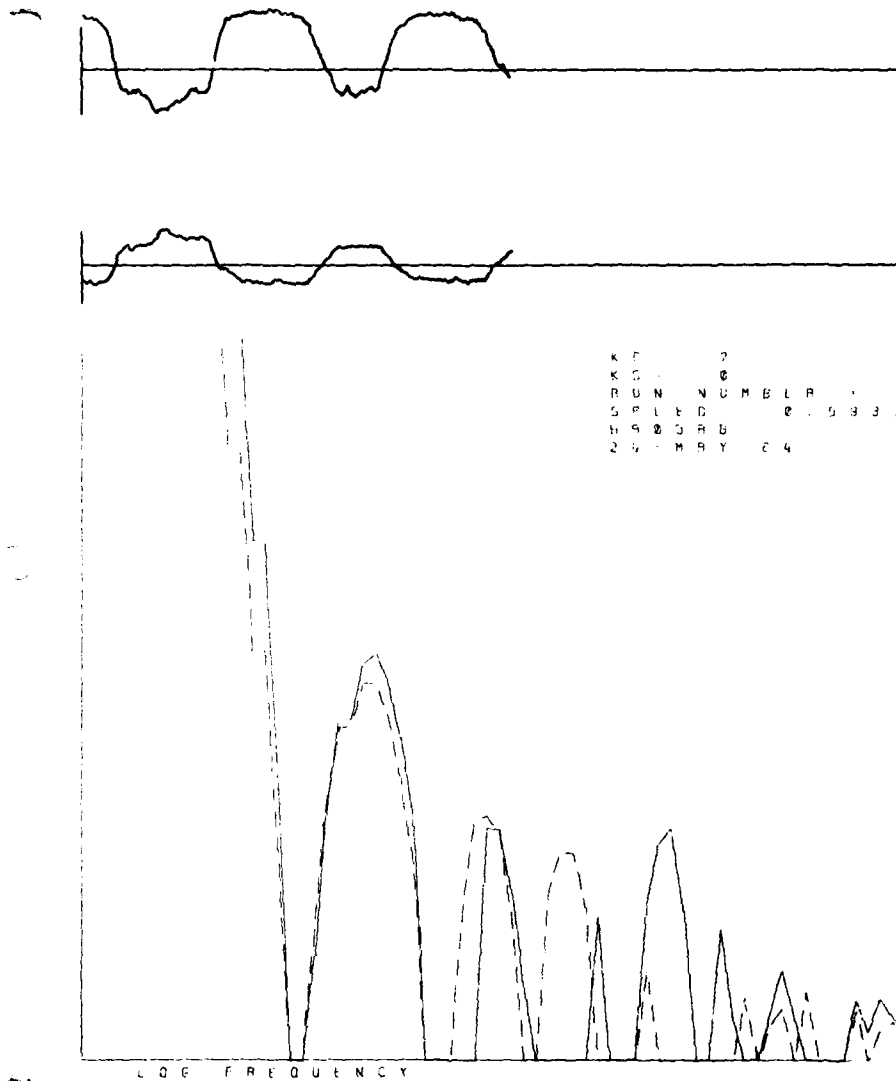
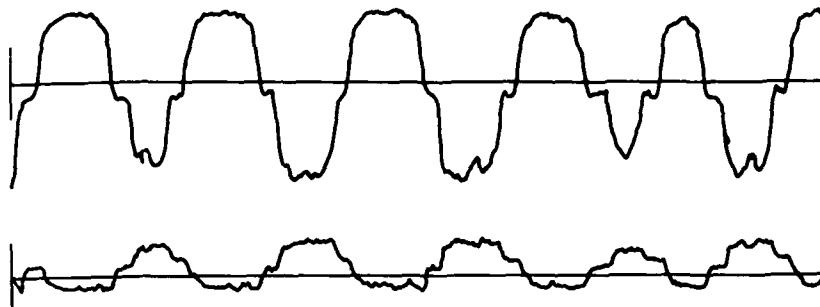


Figure 30 Heading (top) and Rudder Records and Spectral Density Curves from Scale Model. $K_p = 6.78$, $K_D = 0$



K P 6
 K D 1
 RUN NUMBER 1
 S P L E D 8.5651
 H A 2 R B
 2 4 - M A Y - 2 4

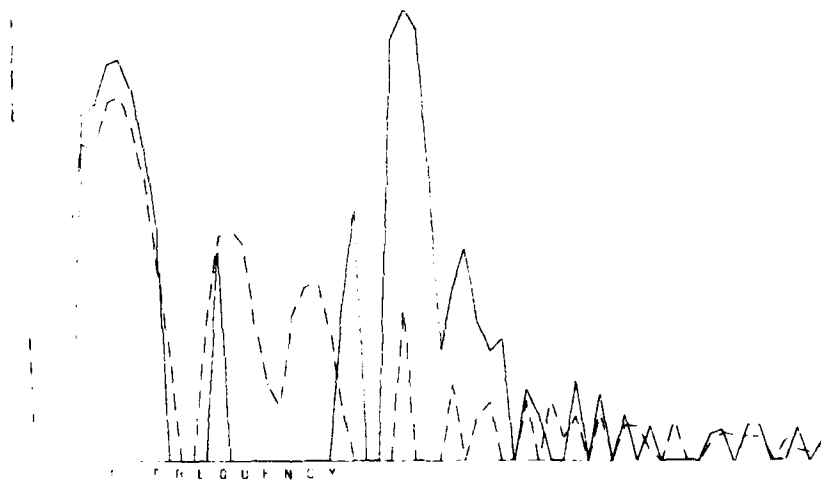


Figure 29 Heading (top) and Rudder Records and Spectral Density Curves from Scale Model. $K_p = 4.61$, $K_D = 0.95$

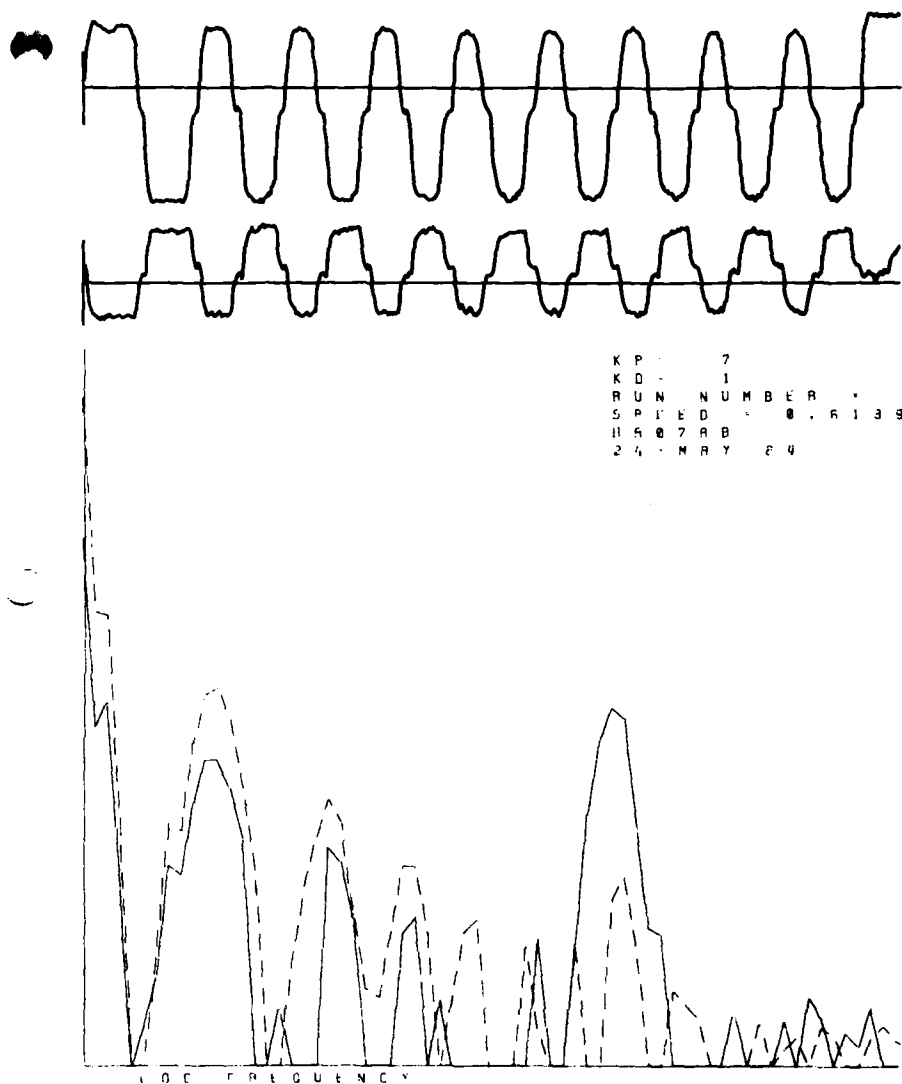


Figure 31 Heading (top) and Rudder Records and Spectral Density Curves from Scale Model. $K_p \approx 6.78$, $K_D = 0.95$

tive gain settings were too high, and that the optimum lay somewhere between 0 and 1. Despite this however, it is felt that the model trials confirmed the hypothesis that for optimum autopilot gain settings there will be the maximum correlation between the rudder and heading spectral density curves over the relevant parts of the frequency spectrum.

ESTIMATING THE WAVE SPECTRUM AND THE ROLL ANGLE AND
ITS DERIVATIVES WITH AN EXTENDED KALMAN FILTER

by J. van Amerongen
P.W. Hoogenraad *)
H.R. van Nauta Lemke

Control Laboratory, Electrical Eng. Dept.,
Delft University of Technology, P.O. Box 5031
2600 GA Delft, The Netherlands.

*) presently with the Royal Netherlands Navy

ABSTRACT

This paper describes a Kalman filter which estimates the wave spectrum, the roll rate and the roll acceleration from measurements of the roll angle. In addition, a filtered version of the roll angle itself is obtained.

It is demonstrated that for a good estimation of these signals not only a mathematical model of the ship, but also a model of the dynamics of the disturbances is essential. Because these dynamics depend on the angle of incidence of the waves, the wave dynamics have to be estimated as well. Therefore an extended Kalman filter has been designed. The paper describes the results obtained from experiments with a discrete and hybrid simulation of a naval ship.

1. INTRODUCTION

For surface ships it may be necessary to reduce the roll motions for several reasons, one of which could be the comfort of passengers and crew. For naval ships it is important to keep the ship fully operational in bad weather conditions.

Roll motions of ships are caused by the following disturbances:

- affect of waves on the ship
- movements of the rudder
- affect of wind on the ship.

In general roll motions caused by the first two disturbances will be of a higher frequency than roll motions caused by the last one. To reduce the roll motions caused by the first two disturbances several stabilisation systems have been developed. The latest development in stabilisation systems is Rudder Roll Stabilisation (RRS), in which roll motions are reduced by movements of the rudder.

In co-operation with the Royal Netherlands Navy and Van Rietschoten and Houwens, Delft University of Technology has recently developed an algorithm for

PRECEDING PAGE FOLLOWS

RRS with adaptive properties, described in a paper by Van Amerongen, Van der Klugt and Pieffers (1984). This algorithm is based on state feedback of the roll angle and its first and second derivatives. In case only measurements of the roll angle are available, its first and second derivatives have to be calculated. This can be achieved by using Kalman filtering techniques.

The following sections describe a Kalman filter which calculates the first and second derivatives of the roll angle. Section 2 deals with the basic Kalman filter theory. In section 3 this basic theory has been applied to calculate the roll angle and its derivatives. The model of the ship which is needed for the Kalman filtering techniques is described. Disturbances which cause roll (the waves) are modelled and implemented in the filter to get a better performance. In section 4 extended Kalman filtering is applied to estimate the parameters of the model which describe the disturbances. Section 5 describes the results of the Kalman filter which was developed in the preceding sections.

2. BASIC KALMAN FILTER THEORY

Kalman filtering is one of the techniques available to estimate all the state variables of a process from measurements of the available output signals. This technique is based on the following principle (fig. 1) (see e.g. Sage and Melsa, 1971).

The input signals, which actuate the process (which is assumed to be linear), are also input to a linear model of the process. The process and model are basically described by:

$$\dot{\underline{x}}(t) = A(t)\underline{x}(t) + B(t)\underline{u}(t) \quad (1)$$

In addition, the process states are disturbed by the process noise $R(t)\underline{v}(t)$, while the model has an additional input $K(t)\underline{e}(t)$, where:

$\underline{x}(t)$ is an n -dimensional state vector
 $\underline{u}(t)$ is an r -dimensional input vector
 $\underline{v}(t)$ is a j -dimensional unknown disturbance vector
 $A(t)$ is an $n \times n$ -matrix
 $B(t)$ is an $n \times r$ -matrix
 $R(t)$ is an $n \times j$ -matrix

Unmodelled inputs as well as modelling errors are represented by the process noise $\underline{v}(t)$, which is assumed to be white and to have a zero mean. Therefore:

$$\begin{aligned} E\{\underline{v}(t)\} &= 0 \\ \text{cov}\{\underline{v}(t), \underline{v}(t+\tau)\} &= V_v(t) \delta_d(\tau) \\ &\text{with } \delta_d \text{ the Dirac function} \end{aligned} \quad (2)$$

The observations are disturbed by observation noise $\underline{w}(t)$, which is also assumed to be white and have a zero mean.

$$\begin{aligned} E\{\underline{w}(t)\} &= 0 \\ \text{cov}\{\underline{w}(t), \underline{w}(t+\tau)\} &= V_w(t) \delta_d(\tau) \end{aligned} \quad (3)$$

The observation itself, which has to be a linear combination of the elements of the

state vector $\underline{x}$, is represented by the following observation model:

$$\underline{z}(t) = C\underline{x}(t) + \underline{w}(t) \quad (4)$$

where: $\underline{z}(t)$ is an m -dimensional observation vector
 $\underline{w}(t)$ is an m -dimensional disturbance vector
 C is an $m \times n$ -matrix

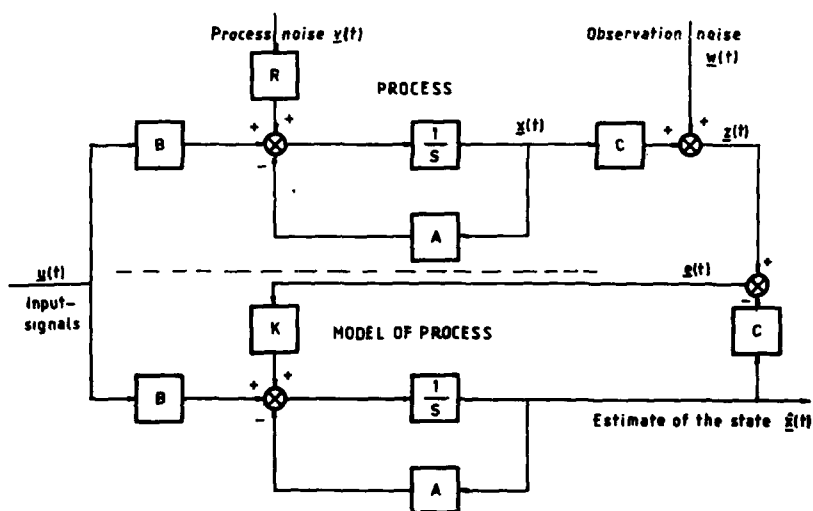


Figure 1

The model of the process produces estimates of $\underline{x}(t)$, denoted as $\hat{\underline{x}}(t)$. Generally, the actual status of the process, $\underline{x}(t)$, will differ from the estimated status due to unknown disturbances (process noise). The model is updated for the influence of process noise (including modelling errors) with the signal $\underline{e}(t) = \underline{z}(t) - C\hat{\underline{x}}(t)$ (see fig. 1). This signal is multiplied by the Kalman filter gain $K(t)$ whose value depends on the ratio between the process and observation noise. This leads to the following equation:

$$\dot{\hat{\underline{x}}}(t) = A(t)\hat{\underline{x}}(t) + B(t)\underline{u}(t) + K(t)\{\underline{z}(t) - C\hat{\underline{x}}(t)\} \quad (5)$$

Because the implementation of the Kalman filter is most convenient in a digital computer, it is necessary to transform the preceding equations into a discrete form. Equations (1) and (4) are transformed into:

$$\underline{x}(k+1) = A(k)\underline{x}(k) + B(k)\underline{u}(k) + R(k)\underline{v}(k) \quad (6)$$

$$\underline{z}(k+1) = C\underline{x}(k+1) + \underline{w}(k+1) \quad (7)$$

where T is the sampling interval and k is used as an abbreviation of $k(T)$. In a

discrete system it makes sense to distinguish between $\bar{x}(k+1)$, the value of $\underline{x}$ based on observations until $t=kT$, and $\hat{x}(k+1)$, based on observations until $t=(k+1)T$. Therefore equation(5) is transformed into:

$$\begin{aligned}\bar{x}(k+1) &= A(k)\underline{x}(k) + B(k)\underline{u}(k) \\ &= E\{\underline{x}(k+1) | \underline{z}(k)\}\end{aligned}\quad (8)$$

$$\begin{aligned}\hat{x}(k+1) &= \bar{x}(k+1) + K(k+1)\{\underline{z}(k+1) - C\bar{x}(k+1)\} \\ &= E\{\underline{x}(k+1) | \underline{z}(k+1)\}\end{aligned}\quad (9)$$

In eqn. (8) the state vector $\bar{x}(k+1)$ can be seen as the best estimation at this point of time when the observation $\underline{z}(k+1)$ is not yet known. This step is called the prediction step. In eqn. (9) the model is updated by making use of the observations $\underline{z}(k+1)$. This step is called the correction step. The discrete covariancies of the process and observation noise are related to the continuous ones by the following equations:

$$V_v(k) = V_v(t)/T \quad (10)$$

$$V_w(k) = V_w(t)/T$$

The variance matrices of $\underline{x}(k)$ before and after the observations $\underline{z}(k)$, respectively $M(k)$ and $P(k)$, are defined as:

$$M(k) = E\{(\underline{x}(k) - \bar{x}(k))(\underline{x}(k) - \bar{x}(k))^T\} \quad (11)$$

$$P(k) = E\{(\underline{x}(k) - \hat{x}(k))(\underline{x}(k) - \hat{x}(k))^T\} \quad (12)$$

where the superscript T indicates the transpose of a matrix.

When $\underline{z}(k)$ and therefore $P(k)$ is known, for $M(k+1)$ the following expression can be found:

$$M(k+1) = A(k)P(k)A(k)^T + R(k)V_v(k)R(k)^T \quad (13)$$

When it is assumed that observation noise, process noise and the system's initial state $\underline{x}(0)$ are independent, that is

$$\text{cov}\{\underline{w}(j), \underline{v}(k)\} = \text{cov}\{\underline{x}(0), \underline{w}(k)\} = \text{cov}\{\underline{x}(0), \underline{v}(k)\} = 0, \quad (14)$$

it is possible to derive an expression for the Kalman gain $K(k+1)$ such that the trace of the variance matrix $P(k+1)$ is a minimum, which implies that the variance of the errors between process states and model states are minimum.

This yields:

$$K(k+1) = M(k+1)C^T[M(k+1)C^T + V_w(k+1)]^{-1} \quad (15)$$

and for the variance matrix $P(k+1)$:

$$P(k+1) = [I - K(k+1)C]M(k+1) \quad (16)$$

The Kalman gain is thus calculated from eqns. (13), (15) and (16); with eqns. (8) and (9) and the observation z the estimation $\hat{x}$ of the state vector x can be calculated. Because this procedure is a sequential one, it can easily be implemented in a digital computer. To start the procedure initial estimates $\hat{x}(0) = E\{x(0)\}$ and $P(0)$ are needed. $P(0)$ can be seen as the uncertainty $\hat{x}(0)$.

This procedure yields the best estimates of the state vector when the probability function of process and observation noise are Gaussian. When these noises are not Gaussian it can be proven that the procedure yields the best linear estimates (see, for example Sage and Melsa, 1971).

3. ESTIMATION OF THE ROLL ANGLE AND ITS DERIVATIVES

The theory of the preceding section has been applied to estimate the roll angle and its derivatives from observations of the roll angle only. When the derivatives of the roll angle are also measured, Kalman filtering techniques can be used to get the best possible estimates of the various state variables.

The availability of a mathematical model of the ship is essential. This model can be obtained by considering the hydrodynamic forces and moments on the ship. The parameters of such a model can be calculated from full-scale trials. Van Amerongen en Van Cappelle (1981) propose the following model of the roll and yaw motions of a ship (fig. 2):

$$\begin{pmatrix} \dot{\phi} \\ \dot{p} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -\omega^2 & -2z\omega \end{pmatrix} \begin{pmatrix} \phi \\ p \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ k_\delta \omega^2 & -k_r \omega^2 \end{pmatrix} \begin{pmatrix} \delta \\ r \end{pmatrix} \quad (17)$$

$$p = \dot{\phi} \\ \tau \ddot{r} + r = n_\delta \delta - n_\phi \phi \quad (18)$$

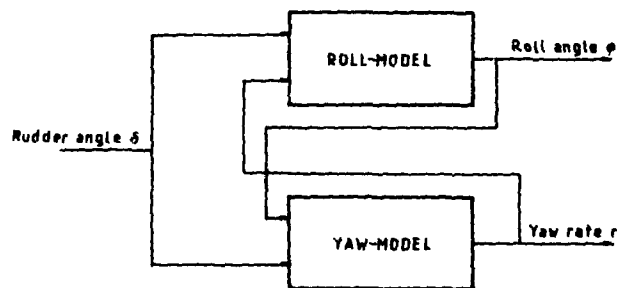


Figure 2.

For calculation of the roll angle and its derivatives, the model of eqn. (17) has been used. In this model the yaw rate r is assumed to be known.

The ship is disturbed by wind and waves. In general, the roll angles caused by wind are of a much lower frequency than the roll angle caused by waves. With respect to the Kalman filter design only the disturbances caused by the waves will be considered. The slowly varying, almost stationary roll angle caused by wind can also be estimated by means of a Kalman filter. In this paper, however, the static roll angle is calculated with a moving average filter. This mean value of the roll angle is subtracted from the measured values, which yields measurements with a zero mean for the Kalman filter.

The roll moment of the waves adds to the roll moments of the rudder angle and the yaw rate. Therefore the model described by eqn. (17) can be extended to the following model:

$$\begin{pmatrix} \dot{\phi} \\ \dot{p} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -\omega^2 & -2z\omega \end{pmatrix} \begin{pmatrix} \phi \\ p \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ k_\delta \omega^2 & -k_r \omega^2 \end{pmatrix} \begin{pmatrix} \delta \\ r \end{pmatrix} + \begin{pmatrix} 0 \\ K\omega^2 \end{pmatrix} v \quad (19)$$

In eqn. (19) v is the disturbance caused by the waves. In general, the factor K will be a function of several variables, such as the angle of encounter of the waves. In this model the factor K will be assumed to be constant. The waves can be described by the ITTC wave spectrum (see e.g. Bhattacharyya, 1978).

$$S_\zeta(\omega) = (172.8 H_s^2) / (\bar{T}^4 \omega^5) \exp(-691 / \bar{T}^4 \omega^4) \quad (20)$$

where H_s is the observed significant wave height
 $\bar{T}$ is the mean of the wave period

For $H_s = 4.85$ m and $\bar{T} = 8.4$ s the wave spectrum is drawn in figure 3.

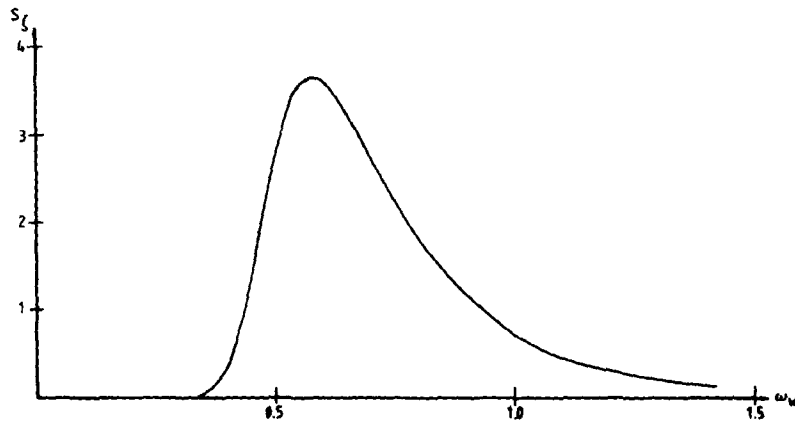


Figure 3.

When the waves are considered as a stationary stochastic process, the wave spectrum is the Fourier transform of the autocorrelation function $S_z(\tau)$ of the waves. So:

$$S_z(\tau) = 1/\pi \int_0^\infty S_z(\omega) \cos \omega \tau d\omega \quad (21)$$

From eqn. (21) it can be seen that this disturbance is non-white. This implies that the Kalman filter will not yield the best possible results unless the colouring of the noise is taken into account.

It is possible to see the non-white noise v as the output of a linear dynamic system driven by white noise (fig. 4).

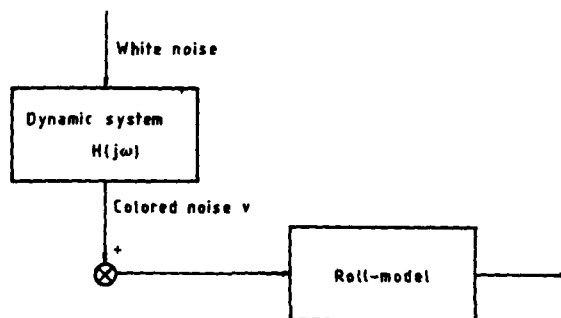


Figure 4.

By setting the Fourier transform of the dynamic system equal to $H(j\omega)$ and the Fourier transform of the autocorrelation function of the white noise by $R(\omega)$, the dynamic system can be determined by solving the following equation:

$$S_z(\omega) = H(j\omega)H(-j\omega)R(\omega) \quad (22)$$

Because the autocorrelation $R(\tau)$ of the input noise is equal to $V_r \delta_D(\tau)$, the Fourier transform of this function is equal to V_r , which is set equal to 1. When $H(j\omega)$ is restricted to a second-order transfer function, the general form of $H(j\omega)$ is:

$$H(j\omega) = \frac{k_w \omega_w^2 (a - b\omega^2 + j\omega)}{\omega_w^2 - \omega^2 + 2z_w \omega_w j\omega} \quad (23)$$

Equation (23) can be written in the form of a minimisation procedure which has the following form:

$$\min! \int_0^\infty [H(j\omega)H(-j\omega) - S_z(\omega)] d\omega \quad (24)$$

k_w, ω_w, z_w, a, b

Because of optimisation a and b are always zero. The parameters ω_w , z_w and k_w are dependent on the sea state, the ship's speed and the angle of encounter of the waves. They can be estimated with extended Kalman filtering. This is discussed in the next section.

One can now adjoin the preceding dynamic system of the noise to the model described in eqn. (19). When the state vector is augmented by ζ_1 and ζ_2 the following formulas are found:

$$\dot{\underline{x}} = A\underline{x} + B\underline{u} + Rv \quad (25)$$

with:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\omega^2 & -2z\omega & 0 & K\omega^2 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -\omega_w^2 & -2z_w\omega_w \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 0 \\ k_\delta\omega^2 & -k_r\omega^2 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

$$\underline{x} = (\phi \ p \ \zeta_1 \ \zeta_2)^T, \quad \underline{u} = (\delta \ r)^T \text{ and } R = (0 \ 0 \ 0 \ k_w\omega_w^2)^T$$

In this model the process noise v is assumed to be white with a variance V_r set equal to 1. On this model the Kalman filtering techniques, which have been discussed in the preceding section, can be applied.

Because the Kalman filter will be implemented in a digital computer, the continuous model has to be transformed into a discrete model. This can be done with the z-transform, which results in the computation of $\exp[AT]$, where A is the transition matrix and T the sample interval. However, the transition matrix A may vary, because of the varying terms ω_w and z_w . Therefore the z-transform should be computed every sample. However, when the sampling rate T is small with regard to the time constants of the system, it is sufficient to compute only the first three terms of the Taylor expansion of $\exp[AT]$. This results in the following discrete model:

$$\underline{x}(k+1) = A(k)\underline{x}(k) + B(k)\underline{u}(k) + R(k)v(k) \quad (26)$$

$$\text{where, } \underline{x} = (\phi \ p \ \zeta_1 \ \zeta_2)^T$$

$$\underline{u} = (\delta \ r)^T$$

$$A(k) = I + AT + A^2T^2/2$$

$$B(k) = BT + ABT^2/2$$

$$A, B \text{ matrices from eqn. (25)}$$

The observation of the roll angle, disregarding the static roll angle, is described by:

$$z(k) = Cx(k) + w(k) \quad (27)$$

where, $C = (1 \ 0 \ 0 \ 0)$

w is observation noise

z is measurement of the roll angle minus the static roll angle

The observation noise w can be split into two factors, namely

- quantisation noise caused by the analogue-to-digital conversion
- inaccuracy of the roll angle sensor

When N is the number of bits in which the roll angle is converted, it can be shown that by linear quantisation the variance of the quantisation noise is equal to:

$$1/3 \{ (Y_{\max} - Y_{\min})^2 \} / 2^{2N+2} \quad (28)$$

where, $Y_{\max}$ and $Y_{\min}$ are respectively the maximum and the minimum roll angle

When $Y_{\max} = 360^\circ$, $Y_{\min} = 0^\circ$ and $N = 14$, the variance of the quantisation noise is 4.10^{-5} . In general the quantisation noise will be non-white. However the variance of the observation noise caused by the inaccuracy of a common roll angle sensor, which is assumed to be white, will be in the order of $2.5 \cdot 10^{-3}$, which is much greater than the variance of the quantisation noise. Therefore, the total observation noise will be approximately white, which is necessary to use Kalman filtering techniques.

In this section it has been indicated how basic Kalman filter theory can be applied to estimate the roll angle and its derivatives. In this case the Kalman filter is mainly used for state estimation, rather than for suppressing observation noise. How to deal with non-white process noise has also been shown.

4. ESTIMATION OF THE PARAMETERS OF THE WAVE MODEL

The model developed in the preceding section, which is described by eqn. (26), has the following parameters: ω , z , ω_w , z_w , k_w and K . The parameters ω and z are ship dependent and will be approximately constant. K depends on several factors but is assumed to be constant. The parameters ω_w , z_w and k_w are dependent on the ship's speed, the sea state and the angle of encounter of the waves. For a good performance of the Kalman filter they have to be estimated. The parameters ω_w and z_w can be estimated by using an extended Kalman filter. The parameter k_w cannot be estimated by this method because it does not appear in the matrix $A(k)$ of eqn. (26). A possible way to estimate k_w is to calculate the variance of the roll angle. Together with the estimation of ω_w an estimation of k_w can be found. The parameter k_w can be seen as an amplifier of the process noise. The variance of the process noise must therefore be multiplied by a factor k_w^2 . In this paper the parameter k_w will be assumed to be constant.

The extended Kalman filter is derived as follows:

When the parameters ω_w and z_w are seen as additional states, the state vector of

eqn. (26) can be augmented with ω_w and z_w . The new state vector has the following form:

$$\underline{x} = (\phi \ p \ \zeta_1 \ \zeta_2 \ \omega_w \ z_w)^T \quad (29)$$

Supposing that at $t = kT$ the states ω_w and z_w have the correct value, the following equations must hold:

$$\begin{aligned} \omega_w(k+1) &= \omega_w(k) \\ z_w(k+1) &= z_w(k) \end{aligned} \quad (30)$$

With the aid of eqns. (29) and (30), the model of eqn. (26) can be transformed into the following form:

$$\underline{x}(k+1) = \underline{f}(\underline{x}(k), \underline{u}(k), k) + \underline{R}(\underline{x}(k), v(k), k) \quad (31)$$

with, $\underline{f}$ and $\underline{R}$ as non-linear vector functions.

Equation (31) describes a non-linear model. Therefore, the basic Kalman filtering techniques cannot be applied.

When the estimated state vector is denoted by $\hat{\underline{x}}$, the non-linear vector functions $\underline{f}$ and $\underline{R}$ can be expanded into a Taylor series. Taking only the first-order terms yields:

$$\underline{x}(k+1) = \underline{f}(\hat{\underline{x}}(k), \underline{u}(k), k) + \frac{\partial \underline{f}(\hat{\underline{x}}(k), \underline{u}(k), k)}{\partial \hat{\underline{x}}(k)} (\underline{x}(k) - \hat{\underline{x}}(k)) + \quad (32)$$

$$\underline{R}(\hat{\underline{x}}(k), v(k))$$

This can be rewritten as:

$$\underline{x}(k+1) = \underline{F}(k) \underline{x}(k) + \underline{R}(k) v(k) + \underline{u}^*(k) \quad (33)$$

$$\text{with } \underline{F}(k) = \frac{\partial \underline{f}(\hat{\underline{x}}(k), \underline{u}(k), k)}{\partial \hat{\underline{x}}(k)}$$

$$\underline{R}(k) = \underline{R}(\hat{\underline{x}}(k), k)$$

$$\underline{u}^*(k) = \underline{f}(\hat{\underline{x}}(k), \underline{u}(k), k) - \frac{\partial \underline{f}(\hat{\underline{x}}(k), \underline{u}(k), k)}{\partial \hat{\underline{x}}(k)} \hat{\underline{x}}(k)$$

Kalman filtering techniques can be applied to the model of eqn. (33) because this model is linear. The apriori estimation of the state vector (before the observation is known) $\bar{\underline{x}}$, can be calculated from the vector function $\underline{f}$. Therefore, the following equation must hold:

$$\bar{\underline{x}}(k+1) = \underline{f}(\bar{\underline{x}}(k), \underline{u}(k), k) \quad (34)$$

The extended Kalman filter not only gives an estimation of the states of the roll model, but also an estimation of the parameters ω_w and z_w . Equation (30) states

that after some time the parameters ω_w and z_w will not change anymore, even if the factors which define ω_w and z_w are changed. This will cause a decreased performance of the estimation of the states. To solve this problem the calculated variance matrix $M(i,j)$ is changed as follows:

$$M_{\text{new}}(k+1)(5,5) = M_{\text{old}}(k+1)(5,5) + a \quad (35)$$

$$M_{\text{new}}(k+1)(6,6) = M_{\text{old}}(k+1)(6,6) + b$$

The values of a and b have to be determined experimentally. It appears that too large values of a and b make the Kalman filter unstable. On the other hand, too low values lead to a slow adjustment after parameter changes. A good compromise appears to be:

$$\begin{aligned} a &= 3 \cdot 10^{-6} \\ b &= 7 \cdot 10^{-6} \end{aligned} \quad (36)$$

In this section it has been shown how the parameters ω_w and z_w can be estimated with an extended Kalman filter. Because the ω_w and z_w may vary in time, the basic Kalman filter is modified, so that it can adapt to these changes. This is done by slightly modifying the variance matrix M .

5. RESULTS

The filter described in the preceding sections has been implemented in a digital computer. Initially, simulations were carried out in which the ship was simulated by an analogue model similar to the one used for the Kalman filter design. These experiments yielded, for instance, the numerical values given in eqns. (35) and (36).

A more realistic situation was simulated by using data obtained from experiments with a more extensive model. This model was based on a hydrodynamical approach and implemented in another digital computer at the Netherlands Ship Model Basin (see Van Amerongen, Van der Klugt and Pieffers, 1984).

Because not only the roll angle, but also the roll rate and the roll acceleration were recorded during these experiments, the estimates of these signals could be compared with the actual state variables. In this computer simulation the measurements of the roll angle were not corrupted by measurement noise. Therefore, noise has been added to the roll angle used in the Kalman filter.

A comparison has been made between the performance of the extended Kalman filter (B) and a filter disregarding the wave dynamics (A). This is expressed by the performance index:

$$\text{Perf.} = \frac{\text{var}(\hat{e})_A}{\text{var}(\hat{e})_B} \quad (37)$$

A value greater than 1 indicates an improved performance of the extended Kalman filter. These experiments demonstrate that the extended Kalman filter is also able to improve the performance if a good estimate of the parameters of the wave model can be guaranteed. However, if the difference between the estimated parameters and the actual values is too large, the performance suffers. This is shown in figs. 6 and 7 for head seas, beam seas and following seas, respectively. These

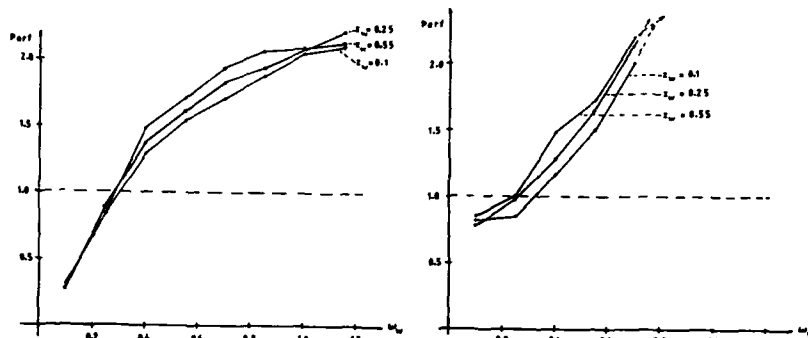


fig 5 Performance of the roll rate and roll acceleration respectively for head seas

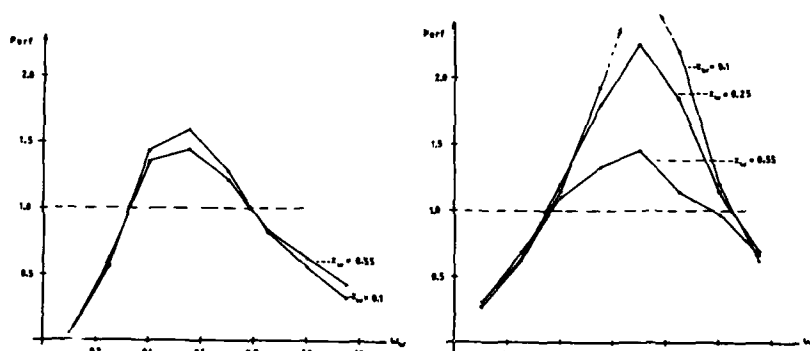
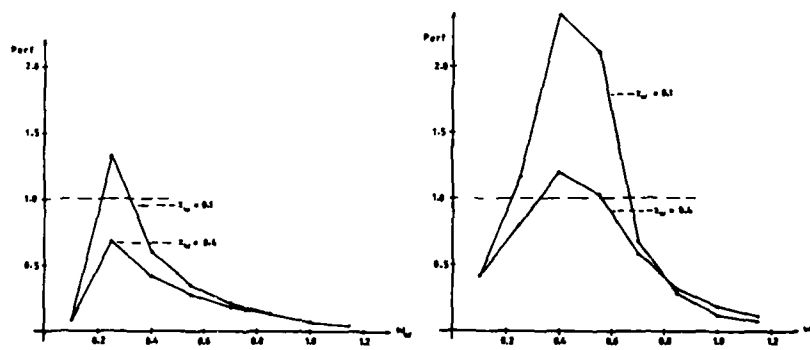


fig 6 Performance of the roll rate and roll acceleration respectively for beam seas



Performance of the roll rate and roll acceleration respectively for following seas

figures were obtained by systematically varying w_w and z_w . The performances are based on a comparison of both filters (A and B) during 15 minutes for each operating condition.

The results show that especially for following seas the parameter adjustment is critical. Because it appears that the extended Kalman filter does indeed yield good estimates of the parameters of the wave model (w_w and z_w), the extended Kalman filter is able to improve the estimation when compared with an "ordinary" Kalman filter.

6. CONCLUSIONS AND SUGGESTIONS

This paper has shown that an extended Kalman filter which is able to produce good estimates of the roll angle and its derivatives in a noisy environment can be designed, based on a relatively simple model of a ship and the disturbances. A sensitivity analysis demonstrates that the adjustment of the parameters of the wave model is not very critical except for following seas, where bad estimates of these parameters quickly deteriorate the performance.

In the present paper the coupling between the wave model and the ship's model has no dynamics (it only contains the gain K , see eqn. (25)). Until now no simple transfer function which further improved the performance of the estimation, could be found. This will be the subject of further research.

REFERENCES

- Amerongen J. van, van der Klugt P.C.M. and Pieffers J.B.M. Model tests and Full-Scale Trials with a Rudder Roll Stabilisation System. Proceedings Seventh Ship Control Systems Symposium, Bath, U.K. 1984.
- Sage A.P. and Melsa J.L. Estimation theory with application to communication and control, McGraw-Hill Book Company, New York, 1971.
- Bhattacharyya R. Dynamics of Marine Vehicles, John Wiley & Sons, New York, 1978.
- Amerongen J. van and van Cappelle J.C.L. Mathematical modelling for rudder roll stabilization. Sixth Ship Control Systems Symposium, Ottawa, Canada, 1981.

COMPARISON OF STEERING CONTROL ALGORITHMS FOR OPTIMIZED AUTOPILOTS

by Vicente C. Garcia, Lt USN
and George J. Thaler
Naval Postgraduate School

ABSTRACT

An automatic steering control is said to be optimized for minimum added resistance due to steering when the controller parameters are adjusted to minimize the cost function

$$J = \frac{1}{T} \int_0^T (\lambda \psi_e^2 + \delta^2) dt$$

where δ = rudder angle
 ψ_e = yaw error
 λ = a weighting factor

Since a variety of control algorithms are possible one must ask if one algorithm provides a lower minimum cost than another?

To study such questions a simulation was made of the SL-7 containership. Several steering control algorithms were inserted in the simulation, and the entire simulation was coupled to a function minimization subroutine. Use of this subroutine adjusted the parameters of each controller to minimize the indicated cost function. Results are tabulated and compared.

INTRODUCTION

In recent years many researchers (1-11) have studied the problem of optimizing an automatic ship steering controller for minimum fuel consumption. It is well known that additional drag is introduced by steering and that both the rudder motion and the yawing motions contribute to this added drag. A measure of the added drag - given as a cost function-is

$$J = \frac{1}{T} \int_0^T (\lambda \psi_e^2 + \delta^2) dt \quad (1)$$

While this expression is an approximation, it is convenient for shipboard use because ψ_e and δ are readily measurable. There is no general agreement on numerical values for the weighting factor, λ , and in this study we used values given by R.E. Reid (8) for the SL-7.

The basic procedure used in this research was to simulate ship steering controller, couple a function minimization subroutine to this simulation and use the subroutine to adjust controller parameters to minimize the cost function and evaluate the minimum cost.

PRECEDING PAGE CONTAINS
4.87

Since such results are no better than the model, we used the best model available, ie, the equations of motion as defined by series expansion including all terms (both linear and nonlinear) for which hydrodynamic coefficients were available. We used a similar technique to obtain the Nomoto second order and third order transfer functions from the equations of motion, and checked these against analytic results from the linearized equations. Fig 1 shows the scheme used to obtain the Nomoto transfer functions, and Fig 2 shows the scheme used to evaluate the controller parameters. Two different programs were used. In preliminary studies an existing, interactive program using function minimization was used with Nomoto models to study controller characteristics in calm water. During this period simulation of the equations of motion was coupled to the function minimization subroutine and to sea state input, and the second program was used for the comparative studies.

Note that for the Nomoto model studies ψ_e and δ were in degrees; when the equations of motion were simulated ψ_e and δ were in radians. Thus the numerical values of the cost, J , are different.

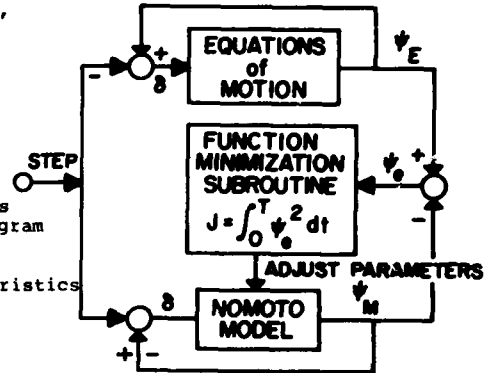


Figure 1. Determination of Nomoto Models

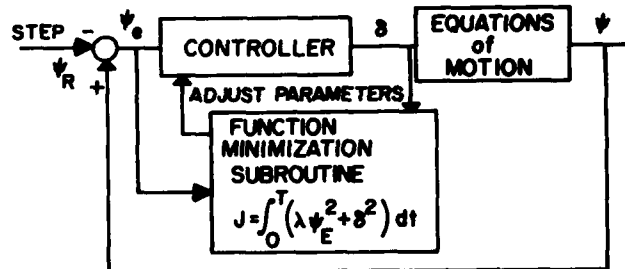


Figure 2. Optimization of Controller

Preliminary Studies

Reid (8,12) uses a second order Nomoto model for the SL-7

$$G = \frac{K}{s(sT+1)} \quad (2)$$

and also uses a controller

$$G_c = \frac{K_1(sT_1+1)}{(sT_2+1)} \quad (3)$$

For structure B, which includes an additional pole, the function minimization subroutine tries to drive the additional pole to infinity, and no doubt would have done so if we had continued the calculations. For structure C which has two poles and two zeros, a zero and pole cancel indicating that they are not needed or wanted. For structure D the integrator gain is driven to zero. The same pattern of results is obtained at 23 knots and 32 knots. Note that in all cases the minimum cost is essentially the same, as would be expected since all controllers are the same. These results seem to indicate that the dynamics of the plant determines the optimum structure for the controller.

Table 2

Simulation Results - Steady State 600 Secs.
Calm Water for Various Controllers
For Fixed Ship Speed (16 Knots)
Nomoto Second Order Model ($K=.1084, T=90.36$)
 $\lambda=16.796$. Optimal Parameter Gains of
Various Controllers, Cost Function

	K1	T1	CONTROLLER GAINS			T4	T1	COST J MIN
			T2	T3				
A	0.4546160	90.43594	10.02143	-	-	-	-	340.854
B	0.4441010	90.29498	9.856573	.01	-	-	-	341.026
C	0.4545110	90.36846	10.02236	23.08439	23.08529	-	-	340.854
D	0.4545813	90.8719	10.02219	-	-	-	1E09	340.854

Table 3

Simulation Results - Steady State 600 Secs.
Calm Water for Various Controllers
For Fixed Ship Speed (23 Knots)
Nomoto Second Order Model ($K=.1556, T=64.67$).
 $\lambda=8.128$. Optimal Parameter Gains of
Various Controllers. Cost Function

	K1	T1	CONTROLLER GAINS			T4	COST J MIN
			T2	T3			
A	0.3731706	64.56579	8.463957	-	-	-	139.9916
B	0.3400238	70.65872	8.889204	.01	-	-	140.9338
C	0.3731387	64.56865	8.463497	25.97382	25.97194	-	139.9991

Table 4

Simulation Results - Steady State 600 Secs.
Calm Water for Various Controllers
For Fixed Ship Speed (32 Knots)
Nomoto Second Order Model ($K=.2167, T=45.45$).
 $\lambda = 4.2$. Optimal Parameter Gains of
Various Controllers. Cost Function

	K1	T1	CONTROLLER GAINS		T4	COST J MIN
			T2	T3		
A	0.3186452	45.4474	7.0661	-	-	60.828
B	0.318	45.45	7.066	.05	-	60.933
C	0.3186779	45.5751	7.0678	50.0483	50.1820	60.828

Nomoto Models of the SL-7

Deriving the second order Nomoto transfer function from the yaw equation* only, the result is

$$\frac{\psi}{\delta} = \frac{.040893}{s(8.539932s+1)} \quad (4A)$$

Using function minimization as in Fig 1

$$\frac{\psi}{\delta} = \frac{.0409221}{s(8.5520782s+1)} \quad (4B)$$

and the agreement is obvious. Using function minimization with both yaw and sway equations, but linear terms only

$$\frac{\psi}{\delta} = \frac{.1072744}{s(31.9199524s+1)} \quad (5A)$$

If the nonlinear terms are included but the perturbation is small

$$\frac{\psi}{\delta} = \frac{.1072082}{s(31.8907013s+1)} \quad (5B)$$

and it is clear that the nonlinear terms contribute little.

* Yaw equation is $\dot{R} = [N_V V + N_R R + N_{\delta} \delta + N_{VV} V^2 + N_{VR} VR^2 + N_{VVV} V^3 + N_{VRR} R^3 + N_{\delta\delta\delta} \delta^3] / (I_z - N_{\dot{r}})$

Sway equation is $\dot{V} = [Y_V V + (Y_R - MASSU) R + Y_{\delta} \delta + Y_{VV} V^2 + Y_{VR} VR^2 + Y_{VVV} V^3 + Y_{VRR} R^3 + Y_{\delta\delta\delta} \delta^3] / (MASS - Y_{\dot{v}})$

Proceeding to the third order Nomoto equation:

$$\frac{\psi}{\delta} = \frac{K(1+T_2s)}{s(1+T_{p1}s)(1+T_{p2}s)} \quad (6)$$

The parameters were calculated and the calculations checked by using function minimization as in Figure 1. The results are given in Table 5. It is clear that the answers obtained by function minimization agree closely with the analytic solutions.

Optimal Controllers (calm water)

Using the computer method of Fig 2 and the third order Nomoto models of Table 5 we optimized the controllers A,B,C of Fig 3. Also using the same method (but a second program) we substituted the nonlinear equations of motion for the Nomoto model and once again optimized. The results are shown in Tables 6,7,8,9,10,11.

Table 5

Third Order Nomoto Model for the SL-7

SPEED KNOTS	K		T4		TP1		TP2	
	CALC	COMP	CALC	COMP	CALC	COMP	CALC	COMP
16	.073812	.072812	22.567	22.945	12.945	12.945	107.583	107.583
23	.1067	.10614	15.675	15.7025	9.014	9.00570	75.130	74.8457
32	.14771	.14771	11.2833	11.2833	6.4699	6.4699	53.7931	53.7931

Table 6

Simulation Results - Steady State 600 Secs.
Calm Water for Various Controllers
For Fixed Ship Speed (16 Knots)
Nomoto Third Order Model
(K=.073812, T2=22.5673, TP1=12.9458, TP2=107.5853)
 $\lambda=16.796$, Optimal Parameter Gains of
Various Controllers, Cost Function

	CONTROLLER GAINS					COST
	K1	T1	T2	T3	T4	J MIN
A	0.6446104	90.09938	15.277127	-	-	370.4023
B	0.6441367	84.826	15.78691	.2459817	-	374.3808
C	0.6151139	107.5782	8.73520	24.96757	12.93679	369.9297

Table 7

Simulation Results - Steady State 600 Secs,
Calm Water For Various Controllers
For Fixed Ship Speed (23 Knots)
Nomoto Third Order Model
($K=.1067, T_Z=15.675, T_{P1}=9.014, T_{P2}=75.13$)
 $\lambda=8.128$, Optimal Parameter Gains of
Various Controllers, Cost Function

	K1	T1	CONTROLLER GAINS			COST J MIN
			T2	T3	T4	
A	0.5224258	63.13609	12.72212	-	-	152.2920
B	0.5216467	64.93709	12.63218	.05051739	-	152.5333
C	0.5001907	75.14852	6.527490	18.26001	9.039928	152.2800

Table 8

Simulation Results - Steady State 600 Secs,
Calm Water For Various Controllers
For Fixed Ship Speed (32 Knots)
Nomoto Third Order Model
($K=.14771, T_Z=11.2833, T_{P1}=6.4699, T_{P2}=53.7931$)
 $\lambda=4.2$, Optimal Parameter Gains of
Various Controllers, Cost Function

	K1	T1	CONTROLLER GAINS			COST J MIN
			T2	T3	T4	
A	0.4276331	48.66048	10.74485	-	-	68.09039
B	0.2987319	89.40696	15.01033	.05977864	-	69.32355
C	0.4179906	53.69961	4.970016	13.85724	6.294354	68.04735

Table 9

Simulation Results - Steady State 600 Secs,
Calm Water for Various Controllers
For Fixed Ship Speed (16 Knots)
Equations of Motion
 $\lambda=16.796$, Optimal Parameter Gains of
Various Controllers. Cost Function

	K1	T1	CONTROLLER GAINS			COST J MIN
			T2	T3	T4	
.64864007	89.817039	15.3816986	-	-	-	1.128189
.6200501	90.6729431	15.5422974	0.9201326	-	-	1.173323
.6173262	107.149399	8.5971985	25.213623	13.3539276	1.126307	

Table 10

Simulation Results - Steady State 600 Secs.
 Calm Water For Various Controllers
 For Fixed Ship Speed (23 Knots)
 Equations of Motion
 $\lambda = 8.128$, Optimal Parameter Gains of
 Various Controllers. Cost Function

	K1	T1	CONTROLLER GAINS				COST J MIN
			T2	T3	T4		
A	.5221061	66.3312225	12.8332672	-	-		0.4640879
B	.4958694	66.1515198	13.0118256	0.9278297	-		0.4857854
C	.5039673	74.7977142	6.6588020	18.4022064	9.2053299		0.4636095

Table 11

Simulation Results - Steady State 600 Secs.
 Calm Water For Various Controllers
 For Fixed Ship Speed (32 Knots)
 Equations of Motion
 $\lambda = 4.2$, Optimal Parameter Gains of
 Various Controllers. Cost Function

	K1	T1	CONTROLLER GAINS				COST J MIN
			T2	T3	T4		
A	.4284037	48.6553955	10.8144264	-	-		0.2072417
B	.2987319	89.40696	15.01033	0.01	-		0.2118334
C	.4173327	53.0965431	5.0965481	14.0205002	6.4748573		0.2071124

Comparison of controller gains and time constants obtained with the third order Nomoto model are essentially the same as those obtained using the equations of motion for a model. One concludes that the third order Nomoto model is a good approximation of ship dynamics in calm water; the second order Nomoto model is not.

Of major interest is the fact that the difference in "cost" between A,B,C is less than 1%. At each speed (16,23,32 knots) controller C is "best", but the difference is slight. Examining the parameter values obtained for controller C, it is seen that at all three speeds both poles of the ship are essentially cancelled by zero of the controller.

State Feedback Controllers

If we assume that the steering dynamics of the ship is adequately modelled as a second order system then we need feedback only two states; for a third order system three states are required. The controller structures are shown on Figure 4. Using the scheme of Fig 2 with no change in cost function or weighting the optimal gains and costs were determined as shown in Tables 12 and 13. Comparing costs,

there is little difference between the two state system and the three state system. If we compare state feedback with controller C, it is seen that at each speed controller C is better, but not much better.

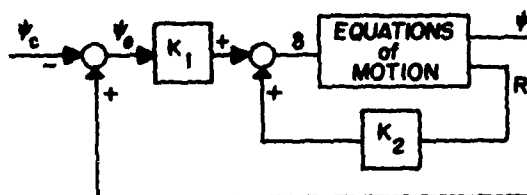


Figure 4A. State Feed Back Controller (2 States)

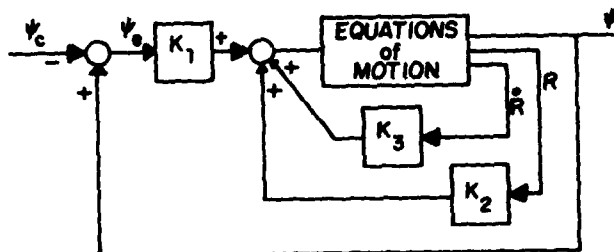


Figure 4B. State Feed Back Controller (3 States)

Table 12

Simulation Results - Steady State 600 Secs,
Calm Water for Various Ship Speeds
Equations of Motion
Optimal Parameter Gains for
Two State System

SPEED KNOTS	GAINS K1	GAINS K2	WEIGHTING FACTOR	COST J MIN
16	4.4033689	77.5041656	16.796	1.128771
23	3.0889006	45.2637787	8.128	.4646050
32	2.2342062	27.6808014	4.2	.2075207

Table 13

Simulation Results - Steady State 600 Secs,
Calm Water for Various Ship Speeds
Equations of Motion
Optimal Parameter Gains for
Three State System

SPEED KNOTS	K1	GAINS K2	K3	WEIGHTING FACTOR	COST J MIN
16	4.8617249	87.7073364	99.9802704	16.796	1.128284
23	3.6630983	56.2784882	88.5913391	8.128	.4643548
32	2.5967150	33.7511444	41.3186025	4.2	.2074225

Figure 5A compares the yaw response to a small course change for the ship with each controller (calm water). Figure 5B compares the rudder activity. Comparison shows little difference, but does indicate that controller C is marginally best in calm water.

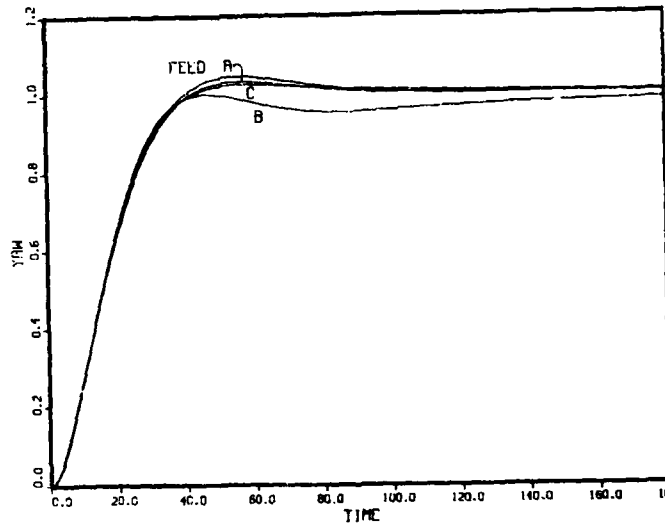


Figure 5A. Yaw Vs. Time (Controller A, B, C and State Feedback)

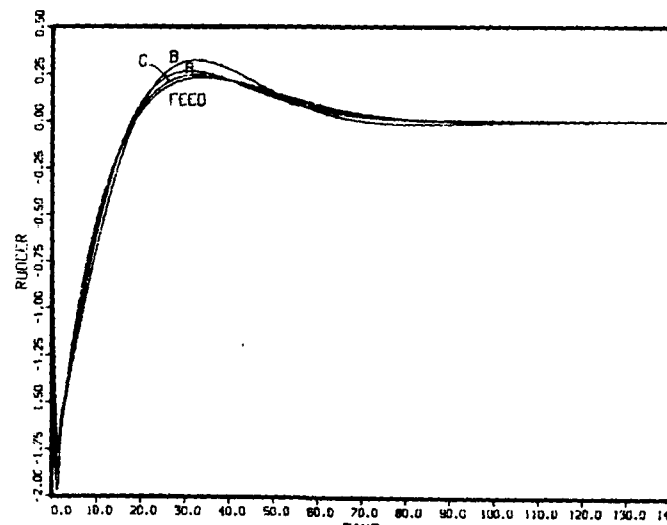


Figure 5B. Rudder Vs. Time (Controller A, B, C and State Feedback)

Optimization of Controllers in Sea State

As seen in Fig 2, provision was made to couple a sea state generator to the ship, so that the function minimization subroutine could be used in the presence of sea state. Our sea state generator was an elaborate program obtained from DTNSRDC. This program generates added mass and inertia values as functions of encounter angle and also forces and moments. We generate the forces and moments, and store in a look up table which is coupled to the equations of motion. Results to date are given in Tables 14 and 15.

Table 14

Simulation Results 600 Second Observation
Fixed Ship Speed (32 Knots) in a Seaway
Ship Model: Equations of Motion
Seaway: Encounter Angle = 30° , Encounter Frequency = .5 rad/sec
 $Y\ddot{v} = -3654800 \text{ lb}_f \text{ sec}^2/\text{ft}$

$$N\ddot{r} = -2.1815E+11 \text{ lb}_f \text{ sec}^2/\text{ft}$$

Parameters, Controller A

State	K_1	T_1	T_2	Jmin
1	.8371677	30.4191437	1.7180796	.04753901
2	.19999962	9.8272924	1.0	.03449364
3	1.7571793	5.0605762	1.0	.002326320
4	4.9999943	23.2652588	8.9242773	.1372949

Table 15

Simulation Results 600 Second Observation
 Fixed Ship Speed (32 Knots) in a Seaway
 Ship Model: Equations of Motion
 Seaway: Encounter Angle = 30° , Encounter Frequency = .5 rad/sec
 $Y\dot{v} = -3654800 \text{ lb}_f \text{ sec}^2/\text{ft}$
 $N\dot{r} = -2.1815\text{E}+11 \text{ lb}_f \text{ sec}^2/\text{ft}$
 Parameters, Controller C

Sea State	K_1	T_1	T_2	T_4	T_3	Jmin
1	.4173327	53.0965424	5.0965481	6.4748573	14.0205002	.0521172
2	.8505149	45.3816376	14.3378038	23.2321675	43.5092773	.03995675
4	1.5054150	99.9998932	75.055237	2.8038440	77.662384	.00270220

Typical transient variations in yaw and rudder angles are shown in Figure 6.

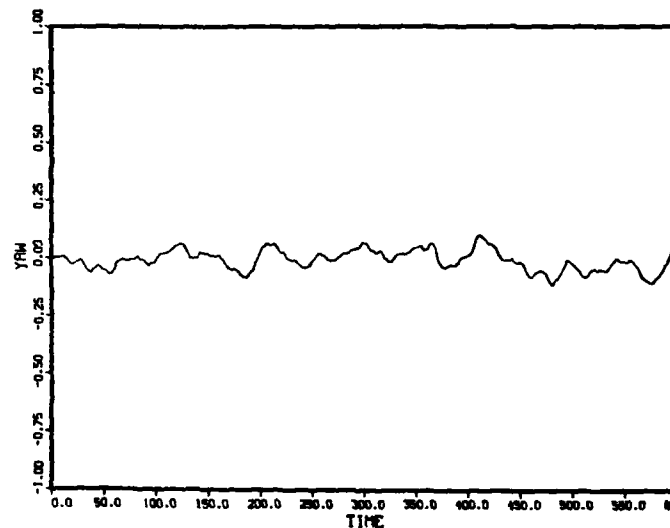


Figure 6A. Yaw Vs. Time-Controller A (32 Knots-Sea State 4)

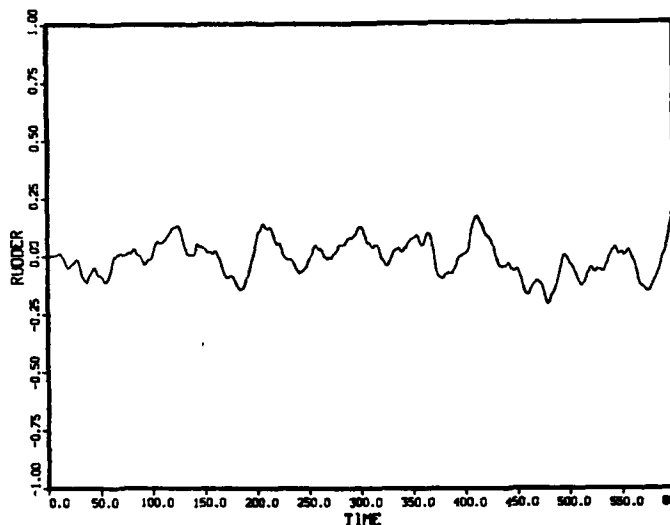


Figure 6B. Rudder Vs. Time-Controller A (32 Knots-Sea State 4)

It is clear from Tables 14 and 15 that parameter values for the optimal controller are functions of both sea state and encounter angle. We do not have enough data to draw firm conclusions, but our studies are continuing.

ACKNOWLEDGEMENTS

We are indebted to Reilly Conrad (NAVSEA), and Young Hong of DTNSRDC for supplying the sea state program, and LCDR Jim Cass for installing and debugging it. Also we thank LT Emmanuel Horianopoulos and LT Pericles Kyritsis Spyromilios for their assistance in generating the data.

References

1. Nomoto K and Motoyama T., Loss of Propulsive Power caused by Yawing with Particular Reference to Automatic Steering. Japan Shipbuilding and Marine Engineering Vol 120, 1966
2. Motora S. and Koyama T., Some Aspects of Automatic Steering of Ships, Japan Shipbuilding and Marine Engineering., Vol 3, No 4 July 1968
3. Norrbin N.H., On the Added Resistance due to Steering on a Straight Course, 13th ITTC Report of Performance Committee 1972
4. Bech M.I, Some Aspects of the Stability of Automatic Course Control of Ships., J. of Mech. Engineering Science., Vol 14 No 7 1972
5. van Amerongen, J. and van Nauta Lemke, H.R., Optimum Steering of Ships with an Adaptive Autopilot Proc 5th Ship Control System Symposium, Annapolis, Md 1978
6. Kallstrom, C.G., et al Adaptive Autopilots for Tankers. Automatica Vol 15, 1979
7. Astom K.J., Design of Fixed Gain and Adaptive Autopilots based on the Nomoto Model Proc of Symposium on Ship Steering Automatic Control Genova, Italy June 1980
8. Reid, R.E. and Moore J.W., Optimal Steering Control of High Speed Containership. Part I System Dynamics, Part II Controller Design Proc 1980 ACC. San Francisco, CA August 1980
9. van Amerongen, V and van Nauta Lemke, H.R., Criteria for Optimum Steering of Ships. Proceedings of Symposium on Ship Steering Automatic Control, Genova, Italy June 1980
10. Kallstrom C.G., and Norrbin, N.H., Performance Criteria for Ship Autopilots: An Analysis of Shipboard Experiments. Proceedings of Symposium on Ship Steering Automatic Control. Genova, Italy 1980
11. Clarke, D., Development of a Cost Function for Autopilot Optimization., Proceedings of Symposium on Ship Steering Automatic Control, Genova, Italy 1980

The main aim of the present paper is to scrutinize many our actual ship's data sets by this time domain approach, which use a multi-dimensional AR model.

The data which we are going to use here are listed up in Tab.1.1, Tab.1.2 and Tab.1.3. Looking at the tables, we notice that the data sets are classified by the ones which were observed on two recent standard ship's and the one which was on a small training ship. Moreover, we notice also that the simultaneous information on a main engine's motion; namely a propeller r.p.m. and a governor's motion are involved in one of the data sets besides ship's motions.

Hence, we shall build not only the AR models about ship's motions, but also more extensive ones including some information about engine's motions.

The following two chapters in this paper are devoted to describe the Akaike's method for fitting a multi-dimensional AR model to the multi-dimensional system with some feedback loops a multi-dimensional AR model, which we are going to use in the later chapters.

In chapter 4, using the above method, we analyse the data set from view point of interaction between ship's motions under various sea conditions. According to the tables, the data set can also be classified by the one which were observed under manual steering, the one which were under conventional autopilot's one and the one which were under optimal one that the authors developed in 1978(Ohtsu et.al. (1978)). Hence, we discuss also different type of interactions of ship's motions which are induced by different steering methods.

In the last chapter, we treat the problem of interaction between ship's motions and main engine's motions under various sea conditions by the data in Tab.1.3.. And basing on this chapter's analysis, we propose an entirely new type of governor for regulating a propeller motion, which takes into account of ship's motions.

Motion	A-1	A-2
Pitching	X	X
Rolling	X	X
Yawing	X	X
Vertical Acceleration	X	X
Lateral Acceleration	X	X
Rudder Angle	X	X
Steering Method	Manual	PID
Sea Condition	Beam Sea W.F. 6	Beam Sea W.F. 6
Figures	4.1 4.3-(A) 4.4-(A) 4.5-(A) 4.6-(A)	4.2 4.3-(B) 4.4-(B) 4.5-(B) 4.6-(B)

Table 1.1 Data Set of Container Ship A (16000 g.ton class)

Motion	AR-1	AR-2	PD-1	PD-2	AR-3
Roll Rate (RR)	X	X	X	X	X
Yawing (YU)	X	X	X	X	X
Rudder Angle (RA)	X	X	X	X	X
Steering Method	Optimal	Optimal	PID	PID	Optimal
Sea Condition	Head Sea W.F. 4	Head Sea W.F. 4	Head Sea W.F. 4	Head Sea W.F. 3	Head Sea W.F. 4
Figures	4.7-(A) 4.8-(A) 4.9-(A)	4.7-(B) 4.8-(B) 4.9-(B)	4.7-(C) 4.8-(C) 4.9-(C)	4.10-(A) 4.11-(A)	4.10-(B) 4.11-(B)

Table 1.2

Data Set of Small Training Ship (330 g.ton class)

Motion	B-1	B-2	B-3	B-4	B-5
Yaw Rate	X	X	X	X	X
Governor	X		X		
Rolling	X	X	X	X	X
Pitching	X	X	X	X	X
Vertical Acc.	X		X	X	X
Propeller r.p.m.	X		X	X	X
Rudder Angle	X	X	X	X	X
Steering Method	PID	PID	PID	PID	PID
Sea Condition	Following Sea U.F.6	Following Sea U.F.6	Head Sea U.F.6	Following Sea U.F.6	Following Sea U.F.6
Figures	4.11	4.12 4.13	5.1	5.2	5.3

Table 1.3 Data Set of Container B (30000 g.ton class)

STATISTICAL ANALYSIS METHOD BY AR MODEL

It is common that the observing data of ship's motions at sea are contaminated with some disturbances as wind and wave. Moreover, it is considered that the yawing data observed under course keeping motion, the variation of the propeller r.p.m at rough sea and so on, has some feedback loops and they may mutually tangle with other ship's motions through them. Thus, considering these situations, it is required that we adopt a stochastic model for analysing such complicated multi-dimensional data.

Now, the exact model for this object may be a statistical ARMA model (Box Jenkins(1970)). But it is wellknown that the identification of this model is formidable. In place of it, we will throughoutly use multi dimensional AR (Auto Regressive) model defined by

$$X(n) = \sum_{m=1}^M A(m)X(n-m) + W(n) \quad (2.1)$$

In this paper, where the $X(n)$ is a k -dimensional vector process of which the i -th element, $x_{ij}(n)$, represents the state of the i -th mode of the ship's motion and $W(n)$, a k -dimensional white noise process of which the i -th element $w_i(n)$ represents a white noise process.

However since the AR model is a statistical model, it does not correspond to an actual physical model. Therefore, we must select an appropriate physical model to be linked with the statistical AR model. In this paper, we adopt a linear impulse response model,

$$X(n) = \sum_{m=1}^M a(m)X(n-m) + U(n) \quad (2.2)$$

The model representing an actual ship's motion at sea, where $U(n)$ is assumed to be generally a colour noise vector process with k dimension of which the i -th element are composed of $u_i(n)$ and $a_{ij}(m)$ is the (i,j) element of the matrix $a(m)$, an impulse response function of the j -th process to the i -th one. But we put a sufficient assumption that $a_{ii}(0) = 0$.

, where $\hat{Q}_M$ denotes the estimate of covariance matrix of the prediction error's term of $W(n)$. According to the MAICE procedure, it is recommended to select the order M at which the AIC takes minimum value. Thus we gained a practical method to identify the stochastic linear process (2.2).

3 ANALYSING TOOLS USING AR MODEL

Once an AR model is fitted to the actual data, we get various important tools for data analysing. Especially, the analysis of feedback system will become easy.

Firstly, an impulse response function $a_{ij}(m)$ is calculated from (2.6). This gives the response of the j -th process to an impulsive change of the i -th process. Moreover, the Fourier transform of the impulse response function

$$a_{ij}(f) = \sum_{m=1}^M a_{ij}(m) \exp(-12\pi f m) \quad (3.1)$$

gives a frequency response function at frequency f .

As a quantity which gives more important informations on interactions between many variables linking with some feedback loops, Akaike proposed a noise contribution function. Before deriving this, let us pay attention that the power spectrum of noise process $u_1(n)$ is easily given by

$$p_{u_1}(f) = \sigma^2 / |1 - \sum_{k=1}^M A_{11}(k) \exp(-12\pi k f)|^2 \quad (3.2)$$

where $\sum^2$ denotes the variance of the prediction error $\epsilon_1(n)$.

On the other hand, we obtain the relation

$$(I - A(f))X(f) = U(f) \quad (3.3)$$

from (2.2), in frequency domain, where

$$X(f) = (X_1(f), X_2(f), \dots, X_k(f))^T$$

$$U(f) = (U_1(f), U_2(f), \dots, U_k(f))^T$$

$$X_1(f) = \sum_{n=1}^N x_1(n) \exp(-12\pi f n)$$

$$U_1(f) = \sum_{n=1}^N u_1(n) \exp(-12\pi f n)$$

and $A(f)$ denotes $k \times k$ matrix of the Fourier transformation of $A_1(n)$. Defining

$$B(f) = (I - A(f))^{-1} \quad (3.4)$$

$B(f)$ denotes the closed loop frequency response function of the noise process $U(n)$ to the process $X(n)$.

Here, calculation $X_1(f) \overline{X_1(f)}$ in order to examine the power spectrum of the process $x_1(n)$, we get the relation

$$\begin{aligned} X_1(f) \overline{X_1(f)} &= \sum_{j=1}^k b_{1j}(f) U_j(f) \overline{U_j(f)} \\ &+ (\text{the cross terms with respect to } U_i(f) \overline{U_j(f)}, i \neq j) \end{aligned} \quad (3.5)$$

But as wellknown in time series theory, we are not given a good result by a direct application of a least square's method for estimation of the parameters of (2.2) except for in the case that $U(n)$ is a white noise process.

Akaike(1968) indicated that the difficulty can be avoided by transforming the model (2.2) to the AR model (2.1). Let us explain his method, using a single input and single output system for simplicity; namely the 2 dimensional case in (2.2), like

$$\begin{cases} x(n) = \sum_{m=1}^M a(m)y(n-m) + u(n) \\ y(n) = \sum_{m=1}^M b(m)x(n-m) + v(n) \end{cases} \quad (2.3)$$

At first, let us write the $u(n)$ and $v(n)$, using the AR models of $u(n)$ and $v(n)$ as

$$\begin{cases} u(n) = \sum_{t=1}^L c(t)u(n-t) + \xi(n) \\ v(n) = \sum_{t=1}^L d(t)v(n-t) + \epsilon(n) \end{cases} \quad (2.4)$$

where $\xi(n)$ and $\epsilon(n)$ are white noise process. Substituting these representations into the basic model (2.3), we gain

$$\begin{aligned} y(n) - \sum_{t=1}^L c(t)y(n-t) &= \sum_{m=1}^M a(m)(x(n-m) - \sum_{t=1}^L c(t)x(n-m-t)) \\ &\quad + u(n) - \sum_{t=1}^L c(t)u(n-t) \end{aligned}$$

For example, with respect to the upper one of (2.3). Completing these operations, we shall have again an AR model

$$y(n) = \sum_{t=1}^L c(t)y(n-t) + \sum_{m=1}^{M+L} A(m)x(n-m) + \xi(n) \quad (2.5)$$

as a result, where the coefficient $A(m)$ and $a(m), c(t)$ have the relations,

$$\begin{cases} A(1) = a(1) \\ A(m) = a(m) - \sum_{t=1}^{m-1} c(t)a(m-t) & (m=2,3,\dots,M+L) \\ c(t)=0 & t > L \end{cases} \quad (2.6)$$

We obtain the same result with (2.5) respectively to the lower one of (2.3). Since the model that we have gained above is an AR model, we can apply an ordinary least square's method to identify the parameters $\{A(m)\}$ without statistical bias. Namely, we can use, for example, the known Levinson-Durbin procedure or the Householder transformation method (Goodwin and Payne(1977)).

The only one problem remained at this stage is how to determine the order M in (2.1). We are going to use the Akaike's Minimum AIC estimate (MAICE) method(Akaike(1974). In the case of a multi-dimensional AR model (2.1), the AIC of the model with the order M is given by

$$AIC(M) = N \log \det(d_M) + 2k^2 M \quad (2.7)$$

Then, if we could assume that the second term of (3.5) is uncorrelative, namely

$$E[u_i(n)u_j(n-k)] = 0 \quad i \neq j \quad (3.6)$$

the power spectrum $p_{11}(f)$ of the process $x_1(n)$ is given by the power spectrum of the noise process $u_j(n)$, $j = 1, 2, \dots, k$.

$$p_{11}(f) = \sum_{j=1}^k |b_{1j}(f)|^2 p_{u_j}(f) \quad (3.7)$$

Since the noise process $u_j(n)$, $j = 1, 2, \dots, k$ drive the whole, this result is reasonable.

In (3.7),

$$q_{1j}(f) = |b_{1j}(f)|^2 p_{u_j}(f) \quad (3.8)$$

yields the contribution of the power spectrum of the noise u_j to the power spectrum $p_{11}(f)$. Therefore, the ratio

$$r_{1j}(f) = q_{1j}(f)/p_{11}(f) \quad (3.9)$$

gives a relative noise contribution. (3.9) is called by a contribution.

$$R_{1j}(f) = \sum_{h=1}^j r_{1h}(f) \quad (3.10)$$

denotes the accumulated contribution. The last representation for a graphical display.

4 INTERACTION BETWEEN SHIP'S MOTIONS

As typical state variables of ship's motions we have rolls, a yaws, a heaves, a sway and a surge motion. We can propeller torques, a propeller r.p.m.s, a thrust and so on variables relating to main engine motions. On the other, a rudder motion, an engine governor and so on as control. In this paper, we will treat the ship's motions in the wide sense as above mentioned.

In the later chapters we consider the time series of variables as the vector process $X(n)$ in chapter 2. And a fitting procedure AR model of the process $X(n)$ in the pt we analyze the ship's motions in a wide sense from vari and confirm the usefulness of the method.

4.1 Ship's Course Keeping Motion and Rudder Steering

4.1.1 Manual steering and PID steering

In this section we analyze the course keeping motion various steering ways, applying the preceding method. we are going to treat here are the ones under a manual conventional and an optimal steering ways.

We begin with the comparison between the data under a manual steering and the one under a conventional PID type of steering, using the data sets as shown in Tab.1.1). These data were observed on a container ship navigating at rough North Pacific Ocean. The sampling rate was 2 sec., and the data length, about 400 points. We might find out the difference between ship's motions under both steerings from these data sets.

Fig.4.1 (A) and Fig.4.1 (B) are the spectra of yawing and rudder motions under manual steering and a PID steering, respectively (Data A-1 and A-2 in Table 1.1). We can see significant low frequency motions in Fig.4.1 (A), while, high frequency motions in Fig.4.1 (B).

Nextly, let us build the multi-dimensional AR models composed of the six variables of Tab.1.1 in both steering methods. Fig.4.2 (A) and Fig.4.2 (B) show the behaviour of the AIC of the each model. From these figures, we notice that the model with order 4 in the manual steering and the one with order 6 in the PID's cases, are gained.

Fig.4.3 (A) and Fig.4.3 (B) demonstrate the noise contributions to yawing (Co.Dev.) in both steering methods. According to Fig.4.3 (A), we note that yawing receives especially a strong effect from the rudder motion at low frequency domain. On the other hand, the effect of the rudder motion is rather expanding into higher frequency domain (Fig.4.3 (B)).

Fig.4.4 (A) and Fig.4.4 (B) denote the noise contributions to rudder motion in both cases. Fig.4.4 (A) indicates existence of a strong feedback loop from yawing to rudder motion at low frequency domain in manual steering. On the other hand, according to Fig.4.4 (B), a feedback loop from yawing to the rudder motion is expanding into higher frequency domain in PID steering.

Fig.4.5 (A) and Fig.4.5 (B) denote the impulse response functions of the rudder motion to impulsive yawing change in both steerings, which are calculated from (3.10). As easily detected, manual steering has the characteristics resembling to a typical response pattern of first order system, while PID steering has a strong beating response.

The last analysis is the frequency response one of these data sets. Fig.4.6 (A) and Fig.4.6 (B) demonstrate the closed loop frequency response functions of the rudder motion to yawing, calculated from (3.4). These figures give us very strong impression about the difference of both steerings. According to Fig.4.6 (A), manual steering is skillful at low frequency, because it responds differentially at low frequency yaw change. Conversely high frequency response is severe for a human being. The conventional PID steering is working out an average feedbacking in whole frequency domain. The figure of the curl at high frequency denotes the effect of some non linear response like weather adjustment of the autopilot.

4.1.2. The Properties of Optimal Steering

The following analysis is to research the properties of optimal steering in comparison with the conventional PID controller.

The authors developed a new type of autopilot system, using an AR model

as the model representing the ship's steering dynamics, where $x(t)$ denotes yawing signal and $u(t)$ the rudder angle (Ohlson (1978)). And this autopilot is controlled by an optimal controller operating under an appropriate evaluation function. We named it OR autopilot. The other autopilot for comparison is an analog type of one which is controlled by the conventional PID control law.

We have repeated many fundamental experiments by an actual ship. Here we extract some characteristic ones from these data (see Table 1-2). Fig.4.7 (A) and (B) demonstrate the noise contribution to yawing in two OR autopilots with different gains. The last gain (OR-2) is imposing more strong weight to yawing than the first one (OR-1). Fig.4.7 (C) demonstrates the noise contribution to yawing in PID steering (PID). Comparing the former two optimal controllers with the last PID ones we notice remarkable reduction of rudder influence to yawing especially at low frequency domain in optimal steering. On the other hands, we can not discerned a strong rudder influence to yawing at low domain.

Let us display the noise contribution to rudder motion in Fig.4.8 (A), (B) and (C) corresponding to Fig.4.7 (A), (B) and (C) respectively, so that we might make clear the difference between both types of steering methods from view point of the pattern of feedback control. We can observe average feedbacking from wider frequency domain than in the PID controller. And the stronger the gain to yawing is strengthened, the higher the optimal controller feedback from yawing. On the hands we can see wasteful rudder motion at important frequency domain in the PID controllers because the noise contribution to the rudder motion to itself is very high in Fig.4.8 (D).

Nextly, let us examine the impulse response functions of rudder motion to yawing. Fig.4.9 (A), (B) and (C) demonstrate them corresponding to the ones in Fig.4.6 or Fig.4.8 respectively. We notice that the characteristic response pattern to the impulsive yaw change are indicated without beating in the OR autopilots after the initial impulsive steering to compensate yawing. In comparison of the former two patterns a salient beating is remained in the PID autopilot as depicted in Fig.4.9 (C).

4.2 Ship motions and their interaction

4.2.1 Rudder effect to Ship Motions

Ordinarily, the rudder motion is moved only in response to a yawing disturbances, but its motion generates a side swaying, a rolling and other motions. Dr. Korvin-Kroukovsky described that a rudder motion is the seventh ship motion (Korvin-Kroukovsky (1961)).

Fig.4.10 (A) and (B) show the rudder effect to rollings in comparison with an optimal controller (OR-3) and a conventional PID in (PD-2). It is clearly noticed that wasteful steering of a PID controller is inducing a strong rolling motion as shown in Fig.4.10 (A), but that of the optimal controller is making it decreased as shown in Fig.4.10 (B).

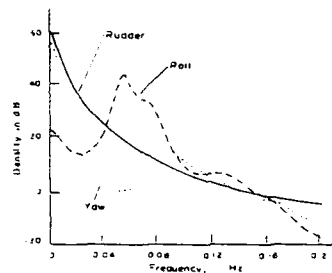


Fig. 4.1-[A]
The Spectra of Yaw, Roll, Rudder
of Container Ship A (Manual Steering)

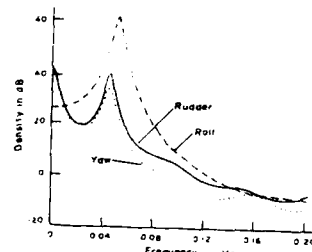


Fig. 4.1-[B]
The Spectra of Yaw, Roll, Rudder
of Container Ship A (PID Steering)

Fitting up to the order $M = 20$ is tried.

ORDER	INNOVATION	VARIANCE				AIC(M)-AIC(MIN) (TRUNCATED AT 40.3)
M	SD(M)		AIC(M)	AIC(M)-AIC(MIN)		
0	0.1341086200-01		566.787	4300.150		
1	0.25547151250-02		-2375.274	1341.101		
2	0.10412702700-03		-1441.843	230.505		
3	0.1216244470-03		-3582.257	130.194		
4	0.7500072000-04		-3713.445	0.000		
5	0.3549423500-04		-3710.117	0.000		
6	0.5587410120-04		-3700.734	11.721		
7	0.4447144760-04		-3713.961	4.400		
8	3.415110240-05		-3600.406	27.550		
9	0.17532172520-04		-3601.427	50.027		
10	0.12150122470-04		-3651.408	34.947		
11	0.2942430040-04		-3617.353	51.100		
12	3.20190041700-04		-3567.251	121.174		
13	0.23500347920-04		-3573.367	144.508		
14	3.2100000000-04		-3548.367	168.569		
15	0.15000000270-04		-3530.374	183.301		
16	3.10001910200-04		-3505.374	200.281		
17	3.14250224500-04		-3502.560	215.550		
18	3.12351000300-04		-3432.344	270.107		
19	3.11304235100-04		-3455.540	292.256		
20	3.10211207300-04		-3420.551	298.304		

*** Minimum AIC = -3420.551/MQ-04 ATTAINED AT M = 20
SD(M) = 0.175321725200-04 *****

[A] Manual Steering

Fitting up to the order $M = 20$ is tried.

ORDER	INNOVATION	VARIANCE				AIC(M)-AIC(MIN) (TRUNCATED AT 40.3)
M	SD(M)		AIC(M)	AIC(M)-AIC(MIN)		
0	0.10417245140-03		-3400.530	2040.527		
1	3.1404930020-05		-3161.130	708.553		
2	3.0140740040-05		-5451.558	115.104		
3	3.40195950200-05		-5485.564	34.477		
4	3.39157800000-05		-5486.529	04.242		
5	0.28570716770-05		-5067.367	2.304		
6	0.24010001800-05		-5050.065	0.300		
7	3.21392770500-05		-5027.777	27.285		
8	0.1857510740-05		-5014.404	13.160		
9	0.15444003140-05		-5023.284	26.470		
10	3.13047701000-05		-5005.270	44.102		
11	3.12710400000-05		-5003.344	60.414		
12	3.11100444400-05		-5041.475	108.700		
13	3.10137044700-05		-5018.303	131.100		
14	3.11037043300-05		-5070.041	171.720		
15	0.2553115710-05		-5045.377	204.280		
16	3.16143090000-05		-5011.043	214.100		
17	3.09412110200-05		-5001.052	250.710		
18	3.3530145530-05		-5140.327	304.115		
19	3.58772581470-05		-5719.314	330.744		
20	3.55251004400-05		-5673.550	370.400		

*** Minimum AIC = -5001.052/MQ-05 ATTAINED AT M = 17
SD(M) = 3.24010001800-05 *****

[B] PID Steering

Fig. 4.2 The Behaviour of AIC of the Container Ship's Model

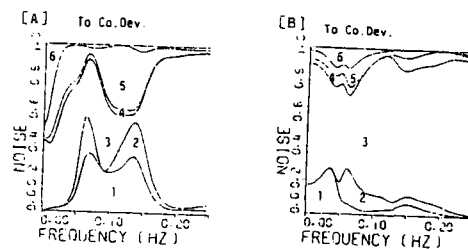


Fig. 4.3
Noise Contribution
to Yaw (Co.Dev.)

[A] Manual
[B] PID

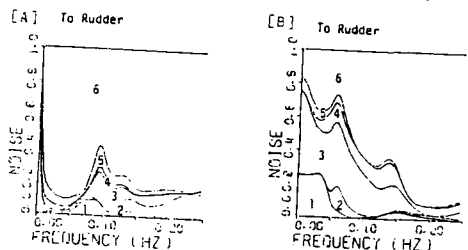
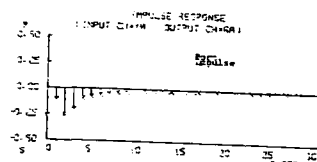


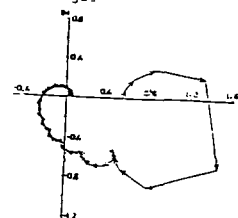
Fig. 4.4
Noise Contribution to Rudder

[A] Manual
[B] PID

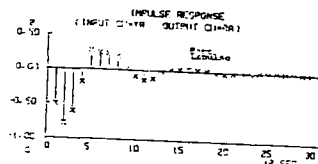
Key : 1. Pitch 2. Roll 3. Yaw
4. Vertical Acc. 5. Lateral Acc. 6. Rudder Angle



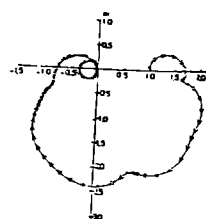
[A] Manual steering



[A] Manual steering



[B] Autopilot steering



[B] Autopilot steering

Fig. 4.5 Impulse Response Function
of Rudder to Yaw

Fig. 4.5 Closed Loop Frequency
Response Function of
Rudder to Yaw

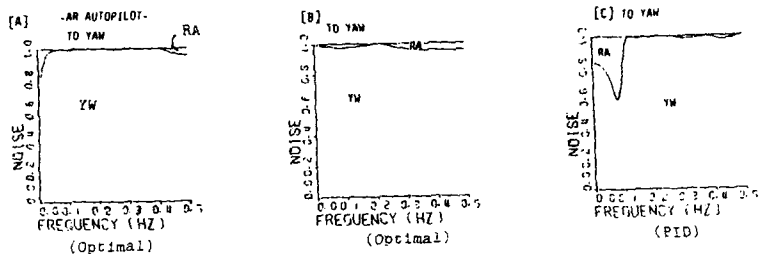


Fig. 4.7 Noise Contribution to Yaw (Optimal and PID Steering)

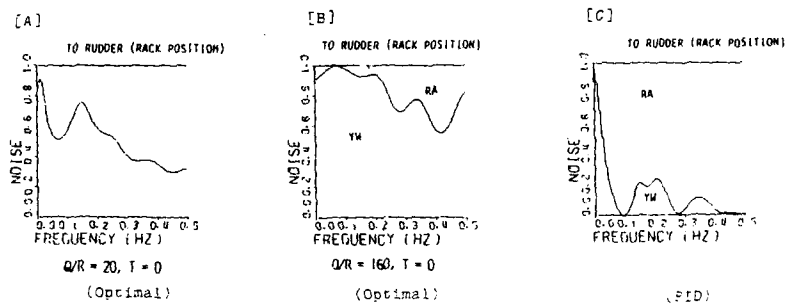


Fig. 4.8 Noise Contribution to Rudder (Optimal and PID Steering)

Key to Fig. 4.7, Fig. 4.9:

YW : Yaw
RA : Rudder Angle

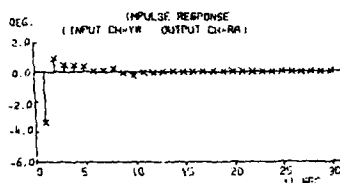
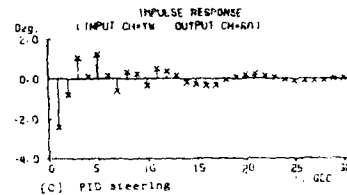
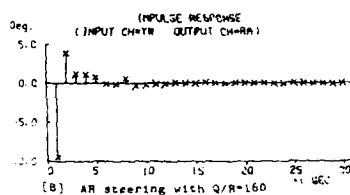
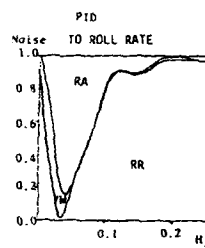


Fig. 4.9 Impulse Response Function of Rudder to Yaw
[A], [B] Optimal
[C] PID



[A]



[B]

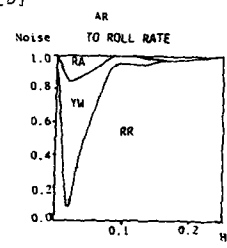


Fig. 4.10 Rudder Effect to Yaw (Noise Contribution to Roll Rate)
(RR : Roll Rate, YW : Yaw Rate, RA : Rudder Angle)
([A] PID, [B] Optimal)

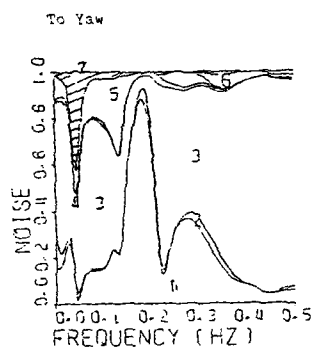


Fig. 4.11 Rudder Effect to Yaw

Key : 1. Yaw Rate 2. Governor
3. Rolling 4. Pitching
5. Vertical Acc. 6. Propeller r.p.m.
7. Rudder Angle

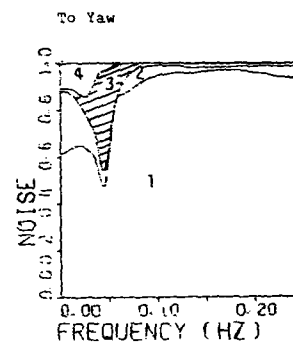


Fig. 4.12 Pitch Yaw Effect
(Noise Contribution to Yaw
(Container Ship B))

Key : 1. Yaw Rate 2. Roll
3. Pitch 4. Rudder Angle

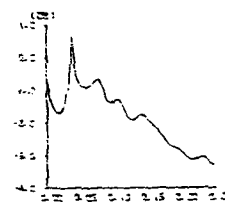


Fig. 4.13 Spectrum of Pitch

the data were analyzed by the MATCH method. The results of the analysis are shown in Table 1. The data show that the MATCH method is capable of detecting the presence of a significant difference between the two groups of data.

The data were also analyzed by the MATCH method. The results of the analysis are shown in Table 1. The data show that the MATCH method is capable of detecting the presence of a significant difference between the two groups of data.

The data were also analyzed by the MATCH method. The results of the analysis are shown in Table 1. The data show that the MATCH method is capable of detecting the presence of a significant difference between the two groups of data.

PROPELLER R.P.M. and SHIP'S MOTIONS

Influence from vertical motions and heaving

A ship's main engine motor receives a strong influence from vertical motions through propeller motion. The authors, however, have only a few reports which treated statistically this problem. In the data shown in Table 1, two data sets of propeller r.p.m. are included. Therefore, the 2-dimensional OR model by the MATCH method included a model to find out the cause of the variation of propeller r.p.m.

Fig. 5.1 demonstrated the noise contribution to the data set in the following sea (B.5), which were induced from the 3rd order model. On the other hand, Fig. 5.2 demonstrated the noise contribution to the data set from the 3rd order model. The data set in the following sea (B.5) were analyzed about 4000 data points. The number of data points and the number of noise data set were about 4000 and 400, respectively.

Indicated from these graphs, we can see that the noise contribution to the data set is very small. The maximum value of the noise contribution is about 0.01. This indicates that the noise contribution is very small.

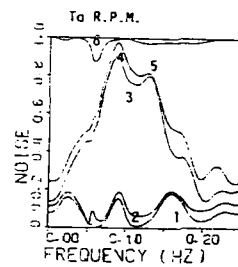


Fig. 5.1 Noise Contribution to Propeller r.p.m. at Head Sea

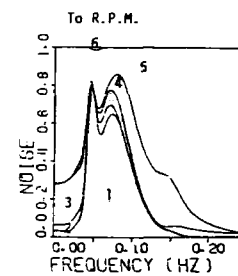


Fig. 5.2 Noise Contribution to Propeller r.p.m. at Follow Sea

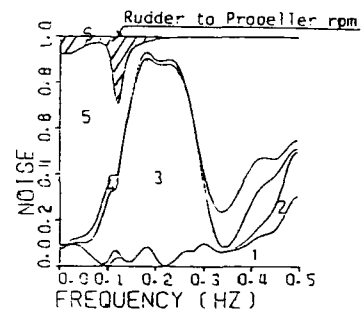


Fig. 5.3 Rudder Effect to Propeller r.p.m. in Noise Contribution to Propeller r.p.m.

Key to Fig. 5.1 - Fig. 5.3 ;

- | | |
|---------------------|------------------|
| 1. Yaw Rate | 2. Roll |
| 3. Pitch | 4. Vertical Acc. |
| 5. Propeller r.p.m. | 6. Rudder Angle |

If a flow field near the propeller is changed for some reasons, propeller load receives the direct influence of it. Since a pitching motion is normally significant at head sea and at follow sea, a larger yawing motion developed at follow sea, it is considered that each motion gives a stronger influence to a flow field near the propeller.

5.2 Influence from rudder motion

It is expected that propeller motion receives an influence from the motion of rudder which is installed near it. According to experiences in the record of a ship's turning motions, we can clearly observe a large influence from a rudder motion to propeller r.p.m.

In Fig. 5.3, we can observe a significant domain of rudder motion from which propeller r.p.m. receives a large effect (B.5). This indicates that propeller r.p.m. receives an important influence from a rudder motion even in comparatively a small motion as a course keeping one.

From these results described above, if we plan to design a new governor regulating a propeller rotation; perhaps which will be a digital type with a multi-variate controllers, we must make great account of the informations of a pitching, a yawing and a rudder motion in particular.

6. CONCLUSION

By the above data dependent analysis, we get the important suggestions for designing optimal controllers of ship's motion and main engine motion as follows.

(1) We must take into account of the Rudder-Yaw effect at low frequency domains, the Rudder-Roll one and the Rudder-Propeller r.p.m. one for a new autopilot system.

(2) Moreover, we can not disregard the Pitch-Propeller r.p.m. effect at head sea and the Yaw-Propeller r.p.m. one at follow sea for new engine governor system.

We could also corroborate the Pitch-Yaw interaction in a container ship's data.

As a whole, we could confirm that the multi dimensional AR modeling method by the MAIC procedure to the ship's data worked very effectively and especially the usefulness of the noise contribution function was proved.

ACKNOWLEDGEMENT

The author devotes great thanks to Dr. G.Kitagawa for completing his study.

-

THE MACHINERY CONTROL ROOM FOR ONBOARD TRAINING

by G.M.Primrose and K.M.Glen
YARD LTD., Glasgow

ABSTRACT

Recent studies and experience have highlighted the importance of Continuation Training in maintaining high standards of control room operator efficiency. Presently, the training needs of these personnel are met by shore-based training establishments using sophisticated training simulators. However, the high cost of running and maintaining these facilities, coupled with the processing power and space savings now being afforded by microprocessor based control and surveillance systems, have led many navies to the opinion that this training should, and could be, given onboard.

This paper identifies the benefits to be gained by the use of an on-board trainer and reviews the functional requirements of such a system. Implementation options are also discussed, in particular the integration of a trainer within a distributed, digitally based, machinery control and surveillance system.

INTRODUCTION

Machinery Control Room (MCR) watchkeepers play a key role in the operation of a modern warship, in ensuring the safe and efficient running of the ship's propulsion and auxiliary machinery. If high standards of watchkeeping are to be maintained, it is essential that whatever initial training is received, regular re-training periods are established whereby operators can refresh their skills and practice procedures they may not have had the occasion to perform during the ship's mission.

The current practice in most navies is to provide this Continuation Training (CT) at shore-based training establishments using sophisticated high-fidelity training simulators. These simulators offer watchkeepers the facility to practice drills and procedures on what is, to all extents and purposes, the real thing, with the knowledge that any mistakes made do not incur damage to actual machinery.

In the UK, the Royal Navy makes extensive use of these simulators both for Pre-Joining and Continuation Training and currently there are some seven simulators meeting the needs of surface ship and submarine personnel.

There is no doubt that these simulators have proven themselves as valuable training aids. However for CT, certain limitations have been identified which are causing many navies to re-examine their current policy and look for more effective means of meeting the requirement.

These limitations are usually due to one, or a combination of, the following factors:-

- incompatibility between the ship's programme and the training programmes of the shore-base
- difficulties associated with the movement of ship's staff between ship and shore-base, particularly when the ship and shore-bases have different geographical locations.
- gradual degradation of simulator fidelity due to differing ship fits and console modifications
- insufficient resources at the shore-base to meet the demands for CT.

As a result of these factors, and with an aim to improving the ship to shore manpower ratio, attention has been directed at improving facilities on board for CT. Currently, some level of CT is given onboard, in the form of "touch drills" but these lack realism and do not exercise the trainees under simulated stress conditions.

A concept that is receiving increased attention, is that of interfacing the actual ship's Machinery Control Console (MCC) to a simulator having dynamic response capabilities and using the console for training while controlling the machinery from some other location. To implement such a radical concept with the previous generation of hard-wired analogue control and surveillance systems, would have proved technically difficult; however the advent of digital microprocessor based systems, using data highways for signal transmission has moved the concept much nearer to realisation. Such systems have the inherent advantages of ease of interface, flexibility and minimal space and power requirements; all of which are directly compatible with the implementation of an onboard trainer concept.

The benefits to be gained by the inclusion of such a facility are obvious.

the limitations of environmental and console fidelity of the shore-base are immediately overcome and there would be no logistic problems with the movement of personnel since, in effect, the trainees would represent a 'captive' audience. Both the operator and the watchkeeping team would be being trained in the real environment both within and without the MCR adding considerable credibility to the training exercise.

Although, at first viewing, the concept looks extremely attractive; its adoption will have a considerable impact both directly and indirectly on the operational philosophy of the ship's Marine Engineering (ME) department and equal consideration must be given to these aspects as well as the technicalities, during the project specification and design phases.

Before discussing the console trainer in depth, it is worthwhile to consider what effect the widespread use of such a facility would have on shore-based establishments. In our view, onboard trainers would relieve the current burden being experienced by the shore-bases, enabling them to concentrate more on Pre-Joining Training and providing operators with the comprehensive systems knowledge required for good watchkeeping. The shore-base would remain the focal point for feedback from the fleet and direct the training programmes given at sea. The onboard trainer would thus supplement rather than supplant the shore-based training establishments.

ONBOARD TRAINER SCOPE

Assuming a requirement has been identified for the provision of an onboard trainer using the MCC, what should be the scope of such a facility and what forms of training should it be used for?

The scope of the trainer will be governed by the nature and level of skills to be acquired e.g.

- What functions of the overall watchkeeping task should be included?
- What operational procedures are important?
- What degree and accuracy of simulation is required?

There are four main facets to MCR watchkeeping

Normal Console Operation. Essentially this is a skill based behaviour associated with low-risk tasks that are frequently performed. The skills involved require a thorough familiarity with the console layout and controls and an awareness of normal plant responses.

Breakdown Diagnosis and Drills. When a breakdown occurs an operator must recognise the symptom, identify the cause and take the necessary action. This necessitates a good understanding of plant and plant interactions coupled with an ability to make the correct decision under what may be stressful conditions.

Team Working. Watchkeeping is essentially a team operation. Members of the team will change during the ship's life and hence continual re-moulding of the team is required to maintain their effectiveness and to refresh their confidence in each other.

- Communications. This aspect could be considered as an inherent part of team working but has been separately identified to highlight the importance of good communications and the practice of standard voice procedures, particularly under damage conditions.

In recent years MCRs have taken on another dimension, namely that of a Damage Control and Surveillance Centre. There is now increased emphasis on designing consoles of an integrated Machinery Controls/Damage Control nature. Thus if the actual MCR console is to be used for training, a decision is required as to whether the Damage Control and Surveillance section should be encompassed within the scope of the trainer. Recent experience in the Falklands, would seem to indicate that this aspect of training should be included.

A further consideration is that of training in local control. In present shore-based simulators this training is given realism by extensive use of visual aids to represent the local machinery spaces and operator actions in these spaces are simulated by the input of signals by the instructor. Extension of the onboard trainer to encompass this aspect, would obviate the need for such animation as the machinery spaces actually exist. However, providing a facility for operators to perform control actions in these spaces, while in the training mode, remains a problem unless some method could be found for injecting pseudo signals at the local positions. The means whereby this could be achieved would depend, to a great extent, on the control system configuration being proposed and how the trainer would interface to it.

At first glance, it might seem that training in Normal Console Operation can be discounted as being unnecessary since the operators will already have the basic knowledge and familiarity required. However such a decision is dependent on several factors eg

- What level of operator is envisaged for this operation? If a Junior Rating is to be used, as a Propulsion-Operator, has he received any Pre-joining training on the shore-based simulator?
- It may be intended to use the trainer as part of career training in which case it may be worthwhile to retain this aspect within the trainers scope.

The final scope of the trainer should be established during consultation between the ships operating staff and the training department. Cost will also be a major consideration since a trainer not limited to specific drills or procedures would require extensive dynamic simulation to represent all plant states and responses under both normal and abnormal (e.g. fault) conditions.

Once the scope of training has been established, the functional requirements of, and the facilities to be provided by, the trainer itself can be considered. For shore-based simulators one of the main requirements is for a high level of console and environmental fidelity such that a realistic physical environment is attained. The use of the actual console for onboard training meets this requirement immediately and the prime task becomes one of assessing what plant simulation is required and what level of instructor facilities should be provided.

In broad terms the simulation must have the functional capability to:

- Accept control signals from the MCC and drive the MCC instrumentation and annunciators.

- Model plant responses under both normal and abnormal conditions.
- Accept signals, simulating the on-plant actions of remote watchkeepers in the machinery spaces.
- Drive VDU pages of a surveillance system if a computerised plant monitoring system is included in the control system design.

Dependent on whether the implementation of the trainer utilises actual controls system hardware, it may also be necessary for the simulation to model certain functions of the controls system e.g. changeover sequences, interlocks, alarm and warning annunciation etc.

The machinery models must produce a realistic response on the panel indicators. The watch under training will be accustomed to the panel responses during their normal watchkeeping activities, any significant departure or over-simplification will result in a loss of credibility in the eyes of the trainees. Therefore, in terms of response times, the simulation must react in real-time or as close as possible to it.

The scope and complexity of the drills and procedures to be practiced on the trainer must also be considered. Obviously abnormal conditions and emergency procedures should be the priority as operators will not have the opportunity to exercise these procedures on a frequent basis. However even in this area there is a choice between simple faults and exercises and those where, for example, during damage conditions, a procedure may be compounded by the introduction of other fault conditions while operators are dealing with a previous fault. The latter is obviously preferred as being the more prevalent situation and the one in which a measure of operator performance under stressful conditions can be most easily assessed, although cost constraints may limit the extent to which it can be achieved.

In terms of scope, the final major factor is the scope of the facilities required for the Training Instructor. Since the instructor performs both a training and assessment role on shore-based simulators, the facilities provided for him must provide a high level of inter-action between him and the trainee. Whether the facilities on an onboard trainer require the same level of sophistication, is a question that will be answered by the scope of training itself. For junior operators such facilities as freeze, playback etc would be useful, but if the trainer has been designed solely for the practice of procedures for experienced personnel, it is believed these facilities could be limited to those necessary for exercise repeatability and ease of initiation.

DESIGN CONSIDERATIONS

General

Any decision to include an onboard training facility must consider the impact such a facility will have on the normal ships operation. A design prerequisite must be a set of constraints within which the design can proceed. Fundamental constraints may be that the inclusion of an onboard trainer must:

- not seriously affect the integrity of the actual ships control system
- not risk any damage to ships machinery

In addition to these fundamental constraints, early consideration must be given to the operational and technical design problems posed by its inclusion eg

- How will the ship's machinery be controlled during training and from where?
- How will the concurrent tasks of training and normal watchkeeping be manned?
- How will communications be maintained during training?
- What additional space will be required to house the additional hardware?

Alternative Control Position

If the MCC is to be isolated from the control system for training purposes, normal control of ships machinery must be achieved from an alternative position.

If Bridge control is envisaged for the ship, then this would seem the obvious choice. Alternatively, machinery could revert to local control during training but penalties, in terms of increased manning, may be incurred to achieve the desired level of co-ordinated control. These manning problems could be overcome to a certain degree, if a limitation on the ships speed during training, can be accepted. e.g. by shutting down the main engines and operating with the cruise engines only.

If neither of these options is available, then an alternative control position (ACP) must be designed for that specific purpose. Should this be the case, then the preferred location for this facility would seem to be the MCC itself, such that the Engineer Officer of the Watch (EOOW) is always aware of the training scenario and, as he has ultimate responsibility for the ships machinery, can make a rapid decision whether to abandon training and transfer back into normal console operation.

Whatever means is adopted for normal control during training, the position must provide adequate surveillance information for the EOOW to continue to perform his normal task. Full duplication of all console alarms and warnings can be ruled out on the basis of space and cost constraints but a critical examination of what surveillance information is required should be made on the basis of what machinery will be operational during training. One option may be to group alarms and warnings by geographical location, rather than by system, thus enabling the EOOW to direct a roving watchkeeper to the particular compartment where the fault exists for identification and diagnosis of the fault.

Obviously training would not be given when the ship was in confined waters or when manoeuvring but would be restricted to open waters when there is little risk of an immediate requirement to revert back to normal control. Under such circumstances, and assuming that all propulsion machinery and auxiliaries are running, the watch could be reduced to for example a roving watchkeeper and the EOOW.

A further alternative not yet addressed is whether to restrict the trained use to harbour only. Such a decision would obviate any manning problems encountered by its operation at sea, but it is felt that this negates the objective to such a considerable extent and introduces further restrictions, that it should not be considered as a viable option.

Communication Facilities

For a training session to be realistic, voice communications between the MCR/Command/Machinery spaces must be exercised. If the normal communication routes and facilities are used during training then some alternative communication routes must be established for normal ship control during training.

On shore-based simulators the instructor's facility incorporates facilities to simulate command orders and it is felt that this approach should be maintained for an onboard trainer also, leaving the Bridge free to communicate directly with the EOW in the MCR or watchkeepers in the machinery spaces. This communication route must not have any disruptive effect on the trainees in the MCR, therefore direct communication links would be preferable to communications of a broadcast form.

Physical Aspects

The physical effects of a decision to include an Onboard Trainer facility must be reviewed early in the ship design. The MCR must accommodate the additional hardware associated with the instructor's facility and the ACP if these are to be stand alone items. The current trend is to design the machinery control consoles on a two tier basis with supervisory control functions being implemented at a supervisor's desk. With such an arrangement one or both of the above facilities could be integrated into the desk thereby reducing inter-console cabling and making for a more functional arrangement.

ONBOARD TRAINER IMPLEMENTATION

General

While the concept of providing training onboard may not be new the practicalities of implementation have tended until now to prevent the incorporation of such facilities. As mentioned previously most of the current machinery control and surveillance systems are hard-wired and as such would present problems when trying to switch the large number of signals involved while ensuring that integrity of the control system was maintained on switchover.

With the advent of microprocessor based control and surveillance systems transferring data between system elements via a data bus, the onboard trainer has moved that much closer to being realised. Before considering the implementation of the trainer in detail it may be useful to review the type of system that the onboard trainer will require to operate with as a result of developments in machinery control systems.

As other papers in this symposium, and in past symposia have described, there is a move in ship's machinery control to software based distributed digital processing with data being transferred between processors via point-to-point highways or bussed systems (either shipwide as, for example, the SDMS system or dedicated to machinery control and surveillance as in the SHINMACS system). Figure 1 shows the conceptual arrangement of such a system and it is to this system that we have addressed implementation of the onboard trainer.

On the basis of the requirements and considerations previously identified, we can identify the main facilities required for the trainer, namely:

An MCS/TRAINER interface

- Simulation models of the ship's machinery
- An instructor facility
- Training Scenarios .

From an implementation viewpoint, without consideration to factors such as integrity, the obvious means of coupling the onboard trainer to the control system under consideration is via a standard interface to the data bus as shown in Figure 1. This appears at a superficial level to be a totally practical solution, the onboard trainer becoming a source of data for driving the machinery control console during training. However, in this situation both real and simulated (or pseudo) data will be present on the bus. Real data must be present if control of machinery is to be effected from an alternative control position (be it on the bridge or at a console located in the MCR) and also to maintain surveillance of the actual propulsion machinery.

Since retention of integrity of the ship's controls system must be fundamental to the implementation of any onboard trainer the main task in interfacing will be to ensure that control signals or data signal generated by the training operation are not misinterpreted, by the plant or control system, as real data, and acted upon accordingly.

Removing this possibility should be one of the main design objectives in implementing an onboard trainer system. For this reason it is considered essential that the design of the onboard trainer and the machinery control system **MUST** be approached as one integrated system.

One possible solution to the problem is to arrange for all machinery to be in local control during training, although as assessed previously, this would have an impact on manning in that during training, local control positions would have to be manned. In addition reversion back to normal control could be protracted. For these reasons, this is a solution which is particularly unattractive.

Alternatively, a software solution could be adopted whereby differentiation between real and pseudo data is achieved by means of parameterisation (or identification) of such signals. This can be done in two ways, by:

- (a) using the same parameter numbers for live and training data signals. However, this will require an extra level of intelligence in the system elements to select between live and pseudo data, plus initialisation sequences to obtain the current values of parameters when switching between training and normal control or vice versa.
- (b) allocating a unique parameter number for each plant data and control signal in the system. This method gets round the shortfalls of using the same parameter numbers and satisfies more easily the possible requirement to maintain and display both live and pseudo data in parallel during training.

If there is a requirement to display real and pseudo data simultaneously, method (b) requires that the processor(s) associated with the machinery control console have access to both live and pseudo data. The application level code in the MCC processors software therefore has to transparently access either the live or pseudo data depending on what mode the machinery control console is in. This can be achieved by maintaining a dual database (of current values) of both live and training data in parallel.

With such a solution an additional application level process must be added to each MCC processor to map on to the correct and appropriate database

switchover. Further, synchronisation of these processors would require to be achieved following switchover to ensure that the MCC instrumentation displays the correct data dependent on the mode.

While not offering a definitive solution we have indicated the importance that must be attached to differentiating between real and pseudo data and we consider it paramount that such considerations are made at an early stage in machinery control system design for implementation of an onboard trainer to be practical.

Machinery Simulation Models

Simulation models of the ship's machinery are required to generate the signals and responses normally sourced from the actual machinery. Development of these models could be a significant task depending on the level of realism being proposed.

It should be noted that with any newly developed machinery control system there is normally a requirement for shore testing of the system to evaluate and test the new design prior to its delivery to the ship.

The shore test may take the form of connecting the system to actual propulsion machinery. However, from a safety and cost viewpoint this may be unacceptable and in the UK the tendency is to utilise machinery simulation models for testing and evaluating control systems. Therefore, for certain machinery control system projects there is the likelihood that software packages are being produced or are already developed which could be utilised or adapted for onboard trainer applications and the possibility therefore exists for sharing development costs between the two facilities. To reap this benefit, it is essential that the requirements are written into the Machinery Control Specification at an early stage such that cognisance is taken of them when developing the models, e.g. machinery behaviour during abnormal as well as normal operating conditions.

An alternative approach would be to develop the onboard trainer simulation software independently thus ensuring that the software is compatible with the particular requirements of the trainer and more specifically the ship's control system design. Generally, a sensible objective for the trainer would be to be able to implement the simulation software on a single processor.

Notwithstanding the chosen method of development we consider that shore test machinery models on their own would probably be inadequate to simulate 'knock-on' effects and the various interactions between shaft sets required for the trainer, therefore some additional modelling would be necessary to simulate the effects of fault conditions either through plant failure or incorrect operator action.

For these reasons it may be advantageous to develop separately an additional model for the trainer to meet the deficiencies identified above. We have identified this function as the Immediate Model. This model would effectively 'fall-in' the machinery simulation models and modify dynamic responses according to the exercise in progress and/or the appropriate operator responses.

Instructor Facility

This facility should be designed such that it allows the instructor to perform the following functions with a minimum degree of interaction:-

- (a) Select the training exercise and set up initial conditions.
- (b) Control the training exercise.
- (c) Program faults into the exercise.

On the basis of cost, space and flexibility, a VDU and keyboard offers the best solution.

The instructor could inter-act with the trainer using either:

- (a) a standard keyboard and a simple command language
- or
- (b) a functional keyboard with each key having a unique pre-determined function.

A command language is slow to use and would require more training to operate but it affords the advantages of being more flexible in terms of change and extension in scope. Functional keys are quicker to use and easier to learn but more difficult to change and limited to the number one can have. In general, the more complex the trainer and the wider its scope, the more likely it is that a command language would meet the requirement.

Training Scenarios

We would suggest that the various training exercises e.g. operating procedures, breakdown drills etc., take the form of training scenarios. These would be developed in consultation with the appropriate training establishments and ship's staff. Each of the training exercises would be considered as a separate scenario, implemented in software in the onboard trainer and which the instructor would, using interactive commands, select as appropriate.

Consideration has been given as to the method of storing the scenarios within the onboard trainer. From a system management viewpoint it would be better to store all scenarios on one medium, e.g. Winchester type disc, but this would have inherent problems when it came to the issue of updates on the scenarios through training feedback or change in procedures or drills. An alternative would be to hold each scenario on a floppy disc or cartridge medium. Scenario selection could then be solely by the loading of the appropriate disk or cartridge. All this would enable the shore (training) establishment to reprogram individual drills/operational procedures and update or issue new scenarios to all class vessels on the basis of feedback received from the fleet.

The requirement for, and use of an immediate modelling facility previously mentioned, should be transparent to the instructor. The immediate model could be loaded automatically into memory as part of the scenario selection sequence. The scenario selected would dictate which immediate model or models are required. The number of immediate models, or commonality between scenarios, may require selection procedure. However, we envisage that the modelling files could be held along with the other specific scenario files.

The requirement for immediate modelling is dependent on the scenario selected. The scenario start-up sequences will detect which scenarios require immediate modelling and which models are to be initiated. The scenario start-up would inform the OT executive which scenarios are to be initiated. The executive functions used to control the modelling are comparable to the scheduling facilities normally provided by MASCOT based systems.

A functional block diagram, depicting the main system elements outlined above, is shown on Figure 2.

Trainer Hardware

Since what we have been discussing is, for the most part, still at a conceptual stage we obviously are not in a position to recommend hardware nor would it be project practice to do so until the particular requirements were clearly defined. It appears however, that in terms of the software structure and the number of software packages requiring development, the best method of implementing the onboard trainer will be via multiprocessing.

Since the software content of an onboard trainer is likely to be relatively large and subject to change or update during its life (as a result of development improvements, machinery changes or class variations) it would seem appropriate that the main program be stored on a mass storage device such as Winchester disk. The main program could be downline loaded from disk to RAM on start up of the onboard trainer.

Since an onboard trainer would likely be classed as 'non-essential' in ship operational terms, there may be significant cost advantages in implementing the trainer in commercial/industrial hardware. It is considered that such hardware, suitably packaged and installed in a shock mounted console/cabinet could provide a system of adequate reliability for its role.

Preliminary considerations suggest that the onboard trainer could be configured into a single console with the electronic assemblies located in the base and the instructor's VDU/keyboard and disc drives at desk level. Every attempt should be made to locate the onboard trainer console in the MCR since its operation by the instructor will be integral with the training operation.

If space is an insurmountable problem in the MCR, consideration may be given to integrating the OT instructor's interface with the MCR supervisor's or EOOW desk. As stated previously, such an arrangement could have operational advantages since the supervisor's desk will be ideally located for allowing the training instructor to carry out his tasks alongside the EOOW.

CONCLUSIONS

For reasons of brevity, it has not been possible to present a detailed analysis of the concept but this outline has identified the major factors to be considered when implementing such a system. The paper has attempted to show that the best chance of success will fall to those who identify the requirement early enough to permit an integrated design approach both with the ship and the machinery control system.

Some problems are foreseen, particularly on the operational side, but we believe the concept to be technically feasible and the benefits to be achieved by its inclusion will act as a stimulus to solving any problems which may arise. We are aware that several navies are currently considering the implementation of such a system and would like to wish them every success in their venture and hope that this paper will be of some value to them.

As a footnote, it should be noted that while the onboard trainer concept described provides a most effective way of providing CT facilities onboard, especially in team training aspects, there is scope for improving individual CT training by utilising the now widely available desk-top microcomputer.

There is a growing awareness, particularly in the civil airline, public

utility and merchant marine fields, of the opportunities afforded by such systems for training aids. Used onboard, they could enable individuals to improve both their systems knowledge and diagnostic skill in an inter-active manner on a readily accessible facility. Provided such training is properly supervised and directed, we believe their use could provide a useful complement to the console trainer itself.

ACKNOWLEDGEMENTS

The encouragement given by the directors of YARD LTD is gratefully acknowledged, as is the generous assistance of the authors' colleagues. Opinions expressed are those of the authors.

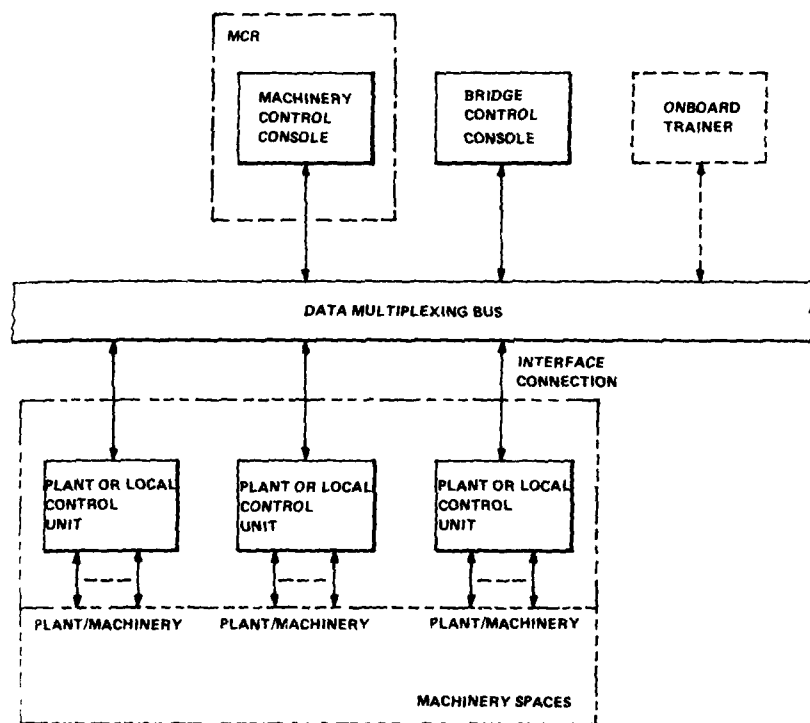


FIGURE 1. CONCEPTUAL MACHINERY CONTROL SYSTEM

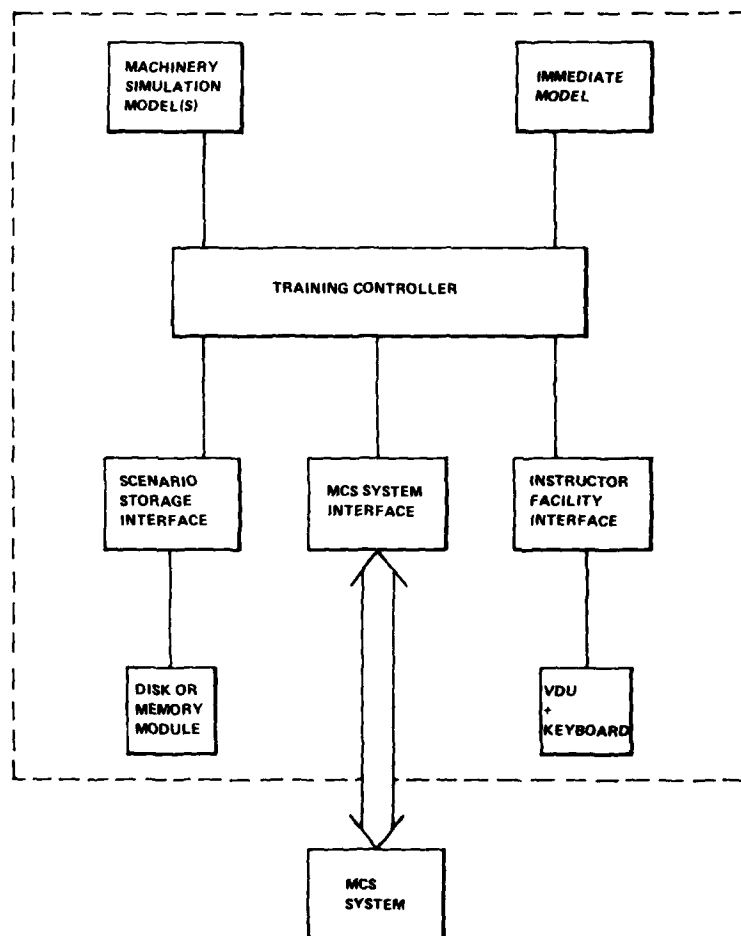


FIGURE 2. ONBOARD TRAINER – FUNCTIONAL BLOCK DIAGRAM

FLEXIBLE CONTROLS FOR A FLEXIBLE SHIP

By C.T.Marwood
Hawker Siddeley Dynamics Engineering Ltd., U.K.
and
T.Munk
Naval Materiel Command, Denmark

ABSTRACT

A multi-role vessel has been planned for the Danish Navy to replace several existing types of ship.

The control and monitoring system for propulsion and all other platform machinery has unusual requirements, arising from the operating flexibility and manning policy. A study of State-of-the-art techniques has been performed to recommend the optimum solution. Parts of this study together with its conclusions form the basis of this paper.

1. INTRODUCTION

Approximately 22 of the smaller ships of the Royal Danish Navy will have to be replaced within the next 10 years.

This includes surveillance units, fast patrol boats, minelayers and coastal minesweepers. They could be replaced one by one with new ships of a similar type, in which case the structure of the Navy would remain unchanged.

It is obvious, however, that the need for ships of a certain type dependant on the actual situation. Sometimes it will be necessary to have a large number of surveillance ships and no minesweepers, while at other times it may be the opposite. This leads to the concept of having a multi-purpose standard vessel, which may be easily converted to the type of ship which is required at a particular time.

This ship will have to cope with a broad range of different tasks demanding special attention to manoeuvrability, draught etc..

The more important are :

- manoeuvring in small harbours

- positioning

- track keeping

- free running, full speed (approx. 30kn) even in shallow waters

- free running, cruising speed (approx. 19kn), low cost

- free running, patrol speed (approx. 8kn), low cost

In addition the ship must be able to operate in bad weather remain at sea in the operation area under all conditions.

Change of Role.

The main problem is to be able to quickly exchange weapon systems and certain role-dependant equipment and to have them fully operational immediately after installation. This will be achieved in part by having a main data bus for integration of nearly all weapons and electronic systems. The different systems, which are to be interchangeable, will be in containers at fixed positions on the ship connected to the ships power system and the main data bus.

The ³ requirements will be accomplished with a combined command bridge and operations room, and here too, a certain standardisation must take place. It will be impossible to have a large number of consoles or a specific console for each system. Therefore, a flexible and universal operator's console will be used. This console will be connected to the main data bus and may be used for any of the connected systems. Normally three standard consoles will be sufficient for the total command and control functions, but up to seven may be required in certain roles.

Manning Policy

All containerised equipment, which is taken onboard, will be followed by the crew necessary for its operation. For practical and economic reasons the size of the crew will be kept very small. To ensure safe operation under these conditions it has been decided to use the following design principles with regards to the main control system :

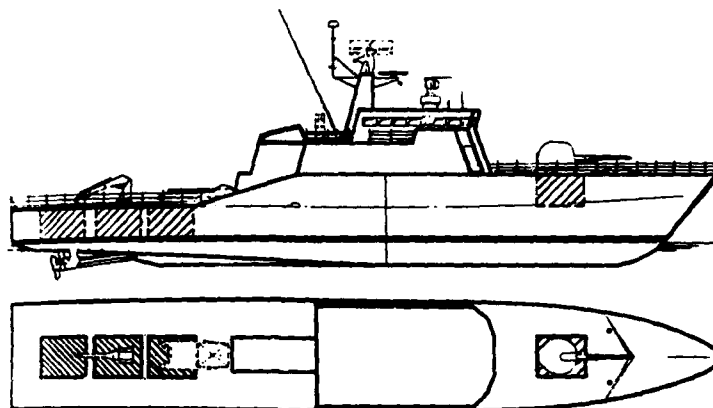
- unmanned machinery room
- unmanned machinery control room during normal conditions
- total command and control from the ship's bridge/operations room
- highly automated machinery operation
- as far as possible, use of standard console for machinery control system
- totally integrated weapons and electronic systems

These demands require a complex command and control system which is normally required only in much larger ships. This system will be expensive in relation to the total cost of the ship. It is expected, however, that the efficiency and the running economy of the ship will justify the increased cost and complexity.

Basic Configuration

The Danish Naval Materiel Command has worked on a project for the design of a ship to meet the multi-purpose concept for the next 10 years. The resulting ship, a flexible standard vessel of approx. 3000 displacement and thus called STANDARD FLEX 300, is now fully designed and ready to be built.

STANDARD FLEX
300
BASIC CONFIGURATION



DISPLACEMENT	300 TONS (STANDARD)
LENGTH O A	54 METERS (177 FEET)
BEAM MAX	8.6 METERS (28.2 FEET)
MAIN ENGINES	CODAG with 2 DIESELS = 2 (2000-2500) HP, 2 SHAFTS = APPROX 19 KNOTS 1 GAS TURBINE = 6-7000 HP 1 SHAFT = APPROX 30 KNOTS 1 AUXILIARY ENGINE 500HP FOR HYDRAULIC DRIVE TO 3 SHAFTS = APPROX 8 KNOTS
ELECTRICAL POWER	3 GENERATORS = 3 x 150 KW
COMPLEMENT	APPROX 15 - 28
ROLL DAMPING	RUDDER - ROLL - STABILIZER U-TANK
HULL MATERIAL	GRP SANDWICH
BOWTHRUSTER	HYDRAULIC, VARIABLE SPEED

Fig. 1. Basic Configuration

The main particulars of the ship are :-

- Overall length 54.00 m approx.
- Beam 8.60 m approx.
- Depth to main deck 4.40 m approx.
- Draft 2.20 m approx.
- Displacement 360 tons

The propulsion equipment will consist of :-

- 1 Gas turbine 6000-7000 hp approx.
- 2 diesel engines 2000-2500 hp each approx.
- 3 diesel generators 150 kw each approx.
- 1 aux. engine 500 hp approx.
 (hydraulic power)
- 1 fixed-pitch propeller in the centre line
- 2 c p wing propellers

The gas turbine or a hydraulic motor may be coupled to the centre shaft while the diesel or a hydraulic motor may be connected to each of the wing shafts.

Manoeuvring

The manoeuvring system will consist of :-

- 1 bow thruster
- 1 rudder in the centre line
- 2 rudder stabilizers aft of the wing propellers

A sketch of the ship in the basic configuration (surveillance) is shown in Figure 1.

All together the propulsion and manoeuvring systems of the ship include a large number of components.

It is considered difficult to manoeuvre safely in narrow water and small harbours, if each component is to be controlled separately. Under these conditions it is desirable to have only one joystick lever, through which the ship handler may control the movement of the ship, and to have an automated system which responds to the lever and generates the necessary command signals to the different propulsion and steering components.

However, this is a break with present naval tradition, in which ship handler maintains a direct control of the propulsion and steering devices by means of wheel and throttles.

The design of an appropriate manoeuvring panel on the bridge in addition to the standard console and its associated computer system is a very important part of the total command and control system.

Study

The success of an unconventional ship such as the STANDARD FLEX 300 depends to a large extent on design and development of the command and control system. In order to ensure the best possible result, a study of state-of-the-art equipments and methodology has been performed. The study tasks included an analysis of the control and monitoring requirements for all the platform machinery, followed by consideration of possible solutions in terms of structure, man-machine interfaces and components. The main aspects considered in this paper are :-

- The implications of one-man operation from the bridge
- The structure and fall back requirements to ensure dependable and reliable operation.
- The needs for flexibility in modes of control and scope for future enhancement.

1. ANALYSIS OF REQUIREMENTS

The detailed requirements for the Ship Control and Surveillance System (SCSS) were evaluated by assessing the control and monitoring needs for each item of plant. It was important to define not only whether open or closed loop control is needed, but also the degree of operator involvement at each control position, ranging from fully automatic to manual.

Control types

Four grades of operator involvement were identified :-

Auto initiated, auto sequenced; no operator involvement.

Example: Start and synchronise diesel generator when electrical load increases.

Operator initiated, auto sequenced; automatic protection and warnings of sequence failure are required, as for A.

Example: Starting main engine.

Setpoint input by operator, auto control; no automatic sequencing. Example : Rudder position.

Operator initiated, operator sequenced; the operator is totally responsible, and the system is used only to transmit the commands. Example : Lining up a fuel transfer path.

FUNCTION	BRIDGE	EQUIPMENT ROOM	LOCAL	ASSUMPTIONS/REMARK
<u>6. FUEL OIL SERVICE SYSTEM</u>				
a. GT Boost Pump Start/Stop	A I	A I	0	
b. GT Boost Pump Auto Cut In	A	A	-	
c. Fuel Filter and Change Over	A	A	-	
<u>7. STRIPPING AND BILGE SYSTEM</u>				
a. Sludge and Bilge Pump			0	
b. Bilge Suction Valves			0	
c. Oil Water Separator			0	
<u>8. SEAWATER COOLING</u>				
a. Sea Water Circulating Pump	A	A	0	
b. Remote Valves	0	0	0	
<u>9. SEAWATER SERVICES/FIREMAIN</u>				
a. Firepump Start/Stop (3)	A 0	A 0	0	Operator selects duty priority
b. Zone Isolating Valves (2)	0	0	0	
<u>10. EMERGENCY BILGE AND BALLAST</u>				
a. Eductor Valves (8 sets)	0	0	0	
<u>11. COMPRESSED AIR</u>				
a. Compressor Start/Stop (2)	A	A 0	0	Operator selects duty priority
<u>12. DOMESTIC FRESH WATER</u>				
a. Cold FW Pump Start/Stop (2)	A	A 0	0	Operator selects duty priority On-plant sequencer
b. Fresh Water Generator Start/Stop	I	I	0	
c. FW Contr. High Salinity Dump	A	A	0	

Table 1 . Control Types

Examples are shown in Table 1.

	<u>Ship Totals</u>					
	A	I	S	O	Not applicable	Total
Bridge	15	17	6	16	7	61
Local	0	3	5	41	12	61

The 16 type O control tasks at the Bridge are simple on-off controls, mainly infrequent e.g. fire extinguishers, hence produce little operator workload, while more frequent or demanding tasks should be automated.

Control and Monitoring Actions

Plant signals were classified as analogue or digital (discrete), with requirements for level detection for alarms and warnings.

The use of each signal was identified in one or more of the following categories :-

1. Analogue monitoring
2. On-Off monitoring
3. On-Off control open loop (e.g. ventilation fan)
4. On-Off control closed loop (Analogue feedback with level detection)
5. On-Off control, local auto (Remote switching of on-plant controller)
6. On-Off control, sequenced (Including interlocks or time delays)
7. Setpoint control (Continuous output signal, either analogue or digitally encoded)

Examples are shown in Table 2.

The total of 1500 signals may seem large for a ship of this size, but the types of plant are almost as varied as those fitted on a frigate. Apart from propulsion and ancillaries, the system is to control hydraulics, electrical generation and distribution, fuel transfer, compressed air, cooling and sea water services including the firemain, ventilation, damage control and domestic equipment.

Since a large number of signals are monitored, the procedure for adjusting the scale and offset and other channel attributes must be simple.

Man - Machine Interfaces (MMI)

Bridge. This is the key position in the SCSS, where ship handling will be carried out, as well as dealing with all other continuous and occasional activities associated with the auxiliary plant. To separate these two major tasks, and to provide the most suitable MMI for manoeuvring, a Ship Handling panel is to be fitted in addition to the

4.138

			CHANNEL TYPES*					ACTION*							
Machinery Description	Qty	Points	AI	DI	AO	DO	A/W	1	2	3	4	5	6	7	Remarks
12. <u>Domestic Fresh Water System</u>															
Fresh Water Generator	1	3	2	0	0	0	1			X					Reverse Osmosis, 3m ³ /day.
Sea Water Supply Pump	1	6	1	2	0	2	1				X				For Reverse Osmosis plant
Fresh Water Tank	1	3	1	0	0	0	2	X			X				
Cold Fresh Water Pump	2	12	2	4	0	4	2			X					Auto scheduled
Hot Water Circulating Pump	1	6	1	2	0	2	1	X							
Hot Water Tank	1	3	1	0	0	0	2	X							C/W electric immersion heater
Remote/Auto controlled valve	1	4	0	2	0	2	0			X					Reverse Osmosis brine overboard discharge
SUB TOTALS	/	10	1	4	0	4	1								
	/	21	4	6	0	6	5								
	/	3	1	0	0	0	2								
	/	3	2	0	0	0	1								
TOTALS	/	37	8	10	0	10	9								

* Channel Types

AI = Analogue Inputs
 DI = Digital Inputs
 AO = Analogue Outputs
 DO = Digital Outputs
 A/W = Alarm/Warnings

* Action Types

1. Analogue Monitoring
2. On/Off Monitoring
3. On/Off Control Open Loop
4. On/Off Control Closed Loop
5. On/Off Control Local Auto
6. On/Off Control Sequenced
7. Continuous Control Set Point

Table 2. Control and Monitoring Actions

console used for machinery control. This panel will include separate controls for power and pitch, as well as a joystick lever for manoeuvring and continuous displays of essential propulsion parameters.

The console used for machinery control must allow safe and effective one-man control of the remaining plant. It is essential to present the operator with manageable information rather than swamp him with raw data, so he can make decisions quickly and correctly, e.g. for manoeuvring or special tasks.

An extra requirement is for joystick control outside the Bridge, and a portable panel is to be provided. Both the portable and fixed Ship Handling panels will be subordinate to the Bridge console.

Equipment Room. This position will normally be unmanned, but will contain a console MMI dedicated to machinery control and surveillance. This position will be used for reversionary control in the event of bridge communications failure, and also for harbour machinery tests and for damage control. It will be manned during action stations. The Equipment Room will be the prime position for adjustment of alarm levels etc., and may be considered as the site for machinery dynamic data recording. Printout facilities will also be provided, as will facilities to transfer recorded data ashore for analysis.

Local control. Vital plant such as steering, engines and pitch will have local mechanical control for emergency fallback.

3. SOLUTIONS

A number of options for the key features of the system were considered in the Study. Some of the reasons for the final choices and their implications are discussed.

System Structure

The number of signals, combined with space restrictions, ruled out a centralised system. The choice of structure for the distributed system depended mainly on three interdependent decisions :-

- Combined or separate systems for control and monitoring
- Number and size of nodes (outstations or MMI processors)
- Distribution of functions between nodes

Control and monitoring have different needs in terms of reliability and response time, and if the number of control signals is small there is a good argument for separate systems, since faults in the monitoring system do not affect the controls. However, in this case the number of outputs is high, and separate systems would greatly increase the number of nodes. Since all control signals must be monitored, separate systems would need to communicate, generating extra data bus traffic. Hence a combined system was chosen.

Number and size of Nodes were constrained by the distribution of signals in the ships, the maximum physical size of cabinet and the effect on cost. Since each node has a fixed overhead cost plus a variable cost depending on the number of channels, large nodes would appear to be more economical. However, the capacity cannot be fully

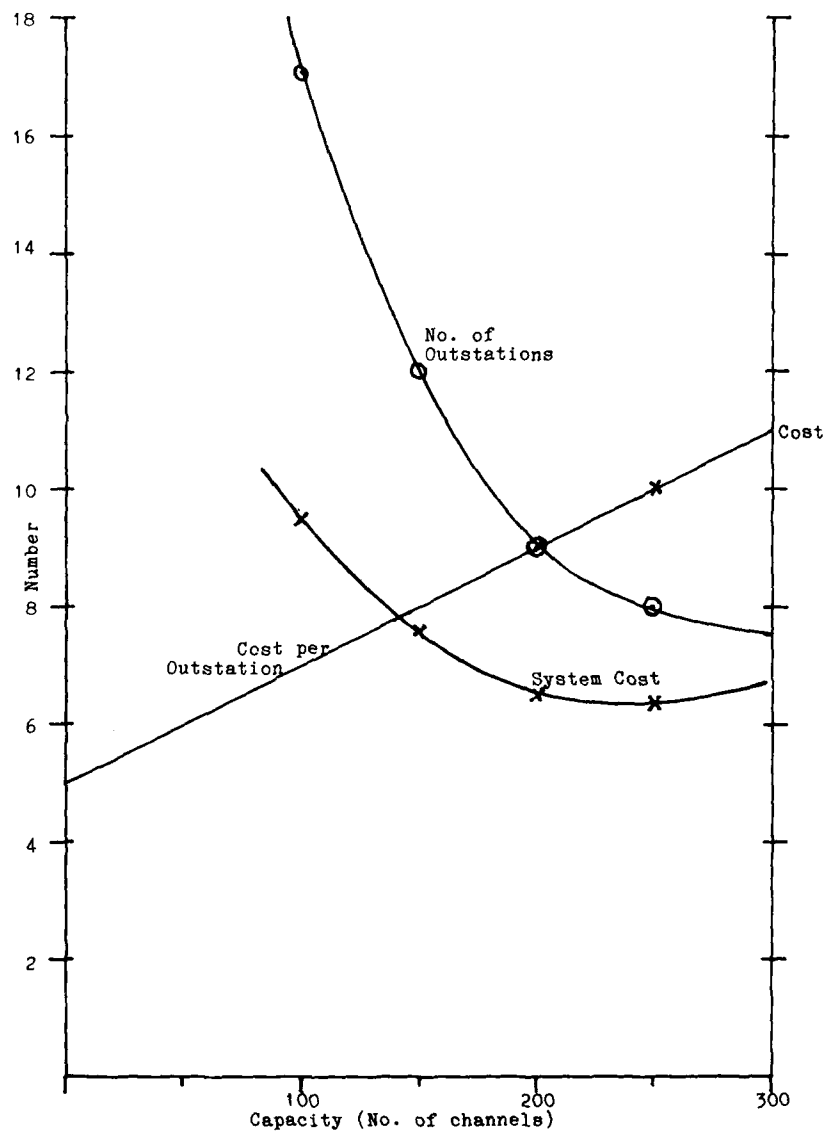


Fig. 2. Cost and Number of Outstations vs Capacity.

utilised because of the distribution of signals in the ship, hence there is an optimum size which gives lowest total cost, as shown in Figure 2.

Distribution of Functions depended on the number of major functions, the number of outstations and ground rules for minimising the incidence of multiple faults.

Major control functions were :-

- Port shaft
- Starboard Shaft
- Centre Shaft
- Bow Thruster
- Port, Starboard and Centre Steering
- Aft Switchboard and Diesel Generators
- Forward Switchboard and Diesel Generator

Ground rules for the optimum distribution of functions between outstations were :-

- a) For integrity, each major function should be controlled entirely by one outstation, and that outstation should have no control of any other major function.
- b) Where plant necessary for more than one shaft is duplicated, e.g. air compressors, the control of each item is allocated to an independent outstation which is not concerned with shaft set control.
- c) All signals other than those necessary for the major control functions will be connected to spare capacity in the nearest outstation.

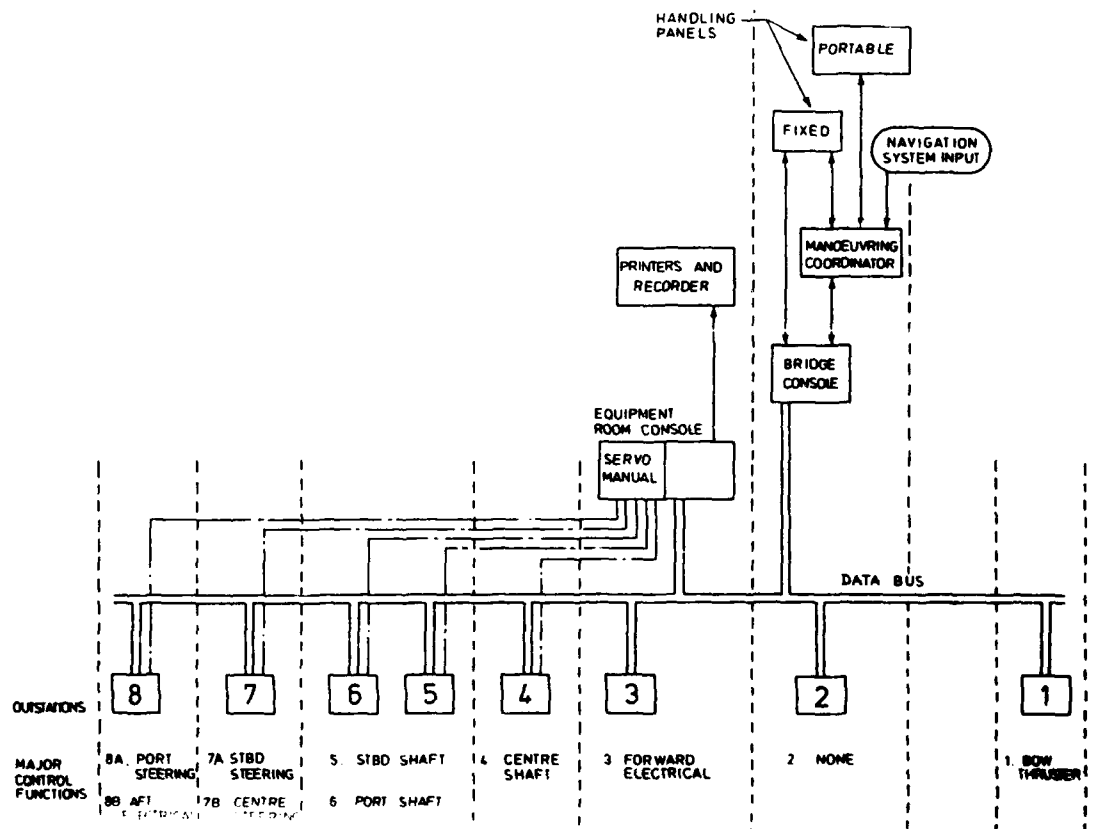
Eight layouts were studied, with varying numbers of outstations and distribution of functions. The final choice was for eight outstations of 250 channel capacity, with two of the outstations each performing two major functions. (Figure 3.) This compromise was permitted since the loss of both these functions did not cause total loss of control. Damage control including fire detection and extinguishers was made a separate system, partly because this significantly reduced the number of signals in the SCSS, and also because the specialised sensors and wiring could be provided at lower overall cost.

Displays and Controls

One-man control at the Bridge console required a careful analysis of the human engineering considerations to decide on the form of presentation. This has been discussed by Gorrell (Ref.1) who shows a Main Machinery Workstation with three colour VDUs with some software-assigned function keys and dedicated controls for propulsion.

Since propulsion and manoeuvring are normally to be controlled at the fixed Ship Handling panel, two VDUs and a function keyboard were assigned for control of ancillaries and auxiliaries. (Figure 4.) Overview mimics will be required for status information, plus one or more pages for each subsystem such as fuel transfer or hydraulics.

Fig. 3. Block Diagram.



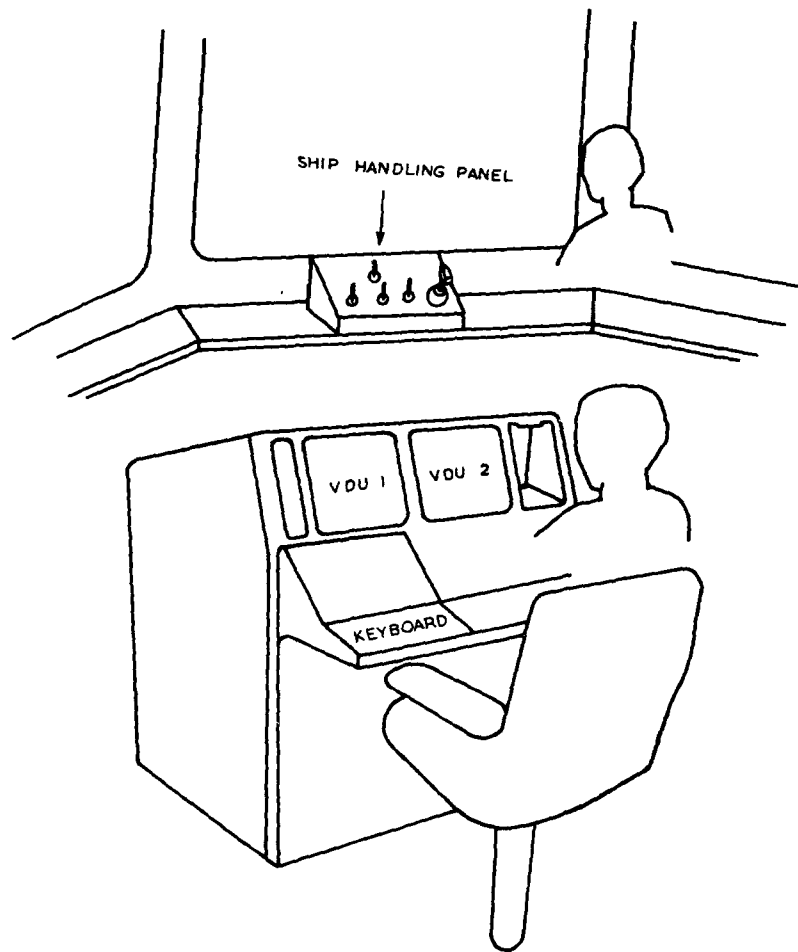


Fig. 4. Bridge Control Positions

Communications Network

This must link the eight outstations with the Bridge and Equipment room consoles, and handle traffic of 100 to 500K bits/sec. The target mean time between failures of the link to the bridge is 100,000 hours, requiring duplicated channels. A bus configuration was preferred rather than star or ring, because of its expansion capability, fault tolerance and cabling cost. IEEE 802.3 ("Ethernet") and MIL-STD-1553B emerged as the best of the standard links, since the chip sets to implement the protocols offer significant savings in size and cost.

Software

There are few, if any, systems of this complexity in use for ship platform control, though weapons systems are often much larger. Constraints on the structure, language and documentation are necessary in a real-time distributed system of this type to ensure thorough testing, freedom from 'side effects' resulting from changes, and long term support after the design team has dispersed.

Structure. The design of the system should be a "top-downwards" approach, starting with the definition of overall system functions before breaking these down into smaller tasks. These should then be subdivided into general-purpose library routines or specific application routines, all with clearly defined interfaces. The structure of data and tables in the system should be specified, and each module described before coding begins.

For the communications software, control of the information exchange should be structured in accordance with the International Standards Organisation reference model known as the Open Standards Interconnection Environment (ISO-OSI model). This defines 7 layers of interconnections, from the Physical Layer 1 up to the Applications Layer 7, and thus allows a clearer understanding of the functions provided by items of hardware, software or a complete network.

Language. High level languages are generally accepted to be more cost effective than Assembler during development and easier for the user to understand and maintain. However, they are less efficient in memory usage and speed of operation, particularly at the hardware/software interfaces. In areas of microprocessor systems where performance is critical there may be significant advantages for using some modules in Assembler, particularly where the functions are unlikely to be changed after delivery, e.g. bus communications protocols.

Commonly used high level languages for small real time systems of this kind are FORTRAN, PASCAL, CORAL and PL/M. These languages are preferred because the software tools for development are widely available, and future support staff are likely to be familiar with them. ADA was considered, but it was felt that the availability of well-proven support software for microprocessor target systems was unlikely within the project timescales, and that the high cost of hardware processors plus additional training would offset much of the benefit.

Documentation. Standards equivalent to JSP 188 (Ref. 2) are necessary to ensure the system can be supported.

Configuration control of software and hardware is vital especially when several ships are in service at different levels of modification.

Diagnostics

A rapid repair time is required, but expertise for onboard fault finding will be limited by low manning. A combination of better than average diagnostics aids will be needed to achieve quick identification of faults, and repair by replacement will be essential for equipment performing vital ship functions.

Self check. Each node must check its own operability as a regular background task, including memory contents and general purpose elements such as analogue/digital converters. Plant transducers and wiring will be monitored for correct operation, including open and short-circuit detection. The diagnostics system should identify the faulty channel and, where possible, the item to be replaced.

Built in test. An overall pre-sea test will be carried out before engines are started to ensure that the SCSS and plant actuators are operational. Plant signals such as shaft speed, needed to close the control loops, will be simulated so that the response to command inputs can be verified using the console displays.

Maintainer Access

A simple MMI will be needed at each node to display diagnostic data from the self check programmes. Signal values in engineering units are required to assist the maintainer in tracing faults outside the system, and for calibrating channels where necessary.

The flexibility of the system is very dependant on the quality of maintainer facilities, and with 1500 signals, the need to configure new channels or altering existing ones will occur frequently. This procedure, which involves entering scaling, offset, datums and other attributes can be made "user friendly" by good ergonomic design. The attributes must be protected against loss of power supplies or module failure.

1. FUTURE AIMS

A flexible system should be readily adaptable to different sizes of ships and numbers of functions.

The system planned for the STANDARD FLEX 300 will include the facility to add 15 % extra channels or even extra outstations and MMI positions if required. The data bus can accept considerably more nodes and data traffic, and the software will be structured to allow reconfiguration without redesign.

Exchange of data between the SCSS and other shipboard networks makes it possible to transmit health monitor information to a shore base for analysis, and receive diagnostic advice for plant and SCSS faults. Alternatively, trend analysis can be performed on-board by extending the processing power at one of the nodes, or by installing a separate processor. (Ref.3). In the latter case, cost can be minimised by using ruggedised commercial hardware and off-the-shelf software packages.

Training facilities, both on-board and ashore, can be provided by extending the simulation software described for Built-in-Test. On-board software can generate realistic displays on existing consoles when the machinery is stopped, simulating normal plant operation or

typical faults.

Dynamic positioning is a possible requirement for many mine countermeasures tasks, and the manoeuvring controls have been specified with the necessary interfaces to allow this to be added with minimal change to the existing system.

5. SUMMARY

The unusual requirements of STANDARD FLEX 300 required a fresh look at methods for controlling the large number of items that make up the ship platform systems. The solution proposed takes advantage of methodology from industrial controls and other advanced naval projects.

The most notable feature is one-man operation from the bridge, resulting in unconventional man-machine interfaces. Particular care has been taken to ensure that the new system will be as dependable as traditional controls, preventing any single fault from causing multiple loss of facilities. This involved distributing vital functions among the processing nodes, and providing three levels of fallback below the normal ship handling position.

It is hoped that the first STANDARD FLEX 300 ships will be ready for trials late in 1986. If, as expected, the container-system concept proves to be successful, it will be applied to larger ships in the future.

This is also true for the command and control system. Experience from the smaller vessels may thus result in a STANDARD FLEX 2000, a 2000 tons displacement combined frigate and fishery inspection ship.

Acknowledgement

The authors would like to thank their colleagues for their contributions to this paper. Opinions expressed are those of the authors.

References

1. E.L. Gorrell, "SHINMACS Machinery Control Console Design", Sixth Ship Control Systems Symposium, Ottawa 1981.
2. JSP188 "Requirements for the Documentation of Software in Military Real-time Computer Systems". Ministry of Defence, U.K.
3. M.J. Curran, "Machinery Monitoring Systems", Sixth Ship Control Systems Symposium, Ottawa 1981.

SURFACE EFFECT SHIP RIDE CONTROL SYSTEM

John D. Adams
Maritime Dynamics, Inc
Tacoma, Washington

Alan W. Ernest
Ernest Engineering
Gig Harbor, Washington

William N. White
Naval Sea Systems Command
Washington, D.C.

ABSTRACT

Surface Effect Ships (SES) operate hullborne like displacement ships or they can operate in a cushionborne mode by pressurizing the region between the catamaran hulls with air. Ride Control Systems (RCS) are installed on these vessels to minimize pressure changes in the air cushion during operation in waves. This smoothes out the vertical motion and improves crew comfort.

This paper summarizes the basic approach that is followed in formulating an SES dynamics model and deriving ride control algorithms using optimal control theory.

A prototype microprocessor-based Ride Control System which contains control logic for performing control algorithms, self-checks, fault monitoring, warnings, and smooth shutdowns is described. Data from tests on the 200-ton U.S. Navy SES-200 surface effect ship are presented to illustrate this system's performance. The system's digital implementation is general purpose in nature. This permits it to be used on different size air cushion-supported vehicles merely by changing coefficients stored in a data table.

INTRODUCTION

Surface Effect Ships (SES) lower their resistance for high speed operation by pressurizing the region between the catamaran hulls and the fore and aft flexible seals with air. Waves induce ship motion by affecting the air cushion that supports the SES and by creating hydrodynamic and hydrostatic forces which act on the catamaran hull.

Ride Control Systems (RCS) have been successfully tested on both small and large U.S. Navy SES testcraft and these systems are now recognized as an integral part of the SES concept. The RCS is used full-time during cushionborne operation as it substantially improves the ride at moderate-to-high speed and it is particularly effective at smoothing out heave motion in the vicinity of the heave resonant frequency.

During a research and development program conducted with the 26-ton XR-1D SES testcraft (Figure 1), modeling of the SES air cushion dynamics was investigated. Methods of applying non-linear

least squares parameter estimation and LQG control theory were developed which provide reasonable assurance that control laws derived for the math model will be effective on the ship.

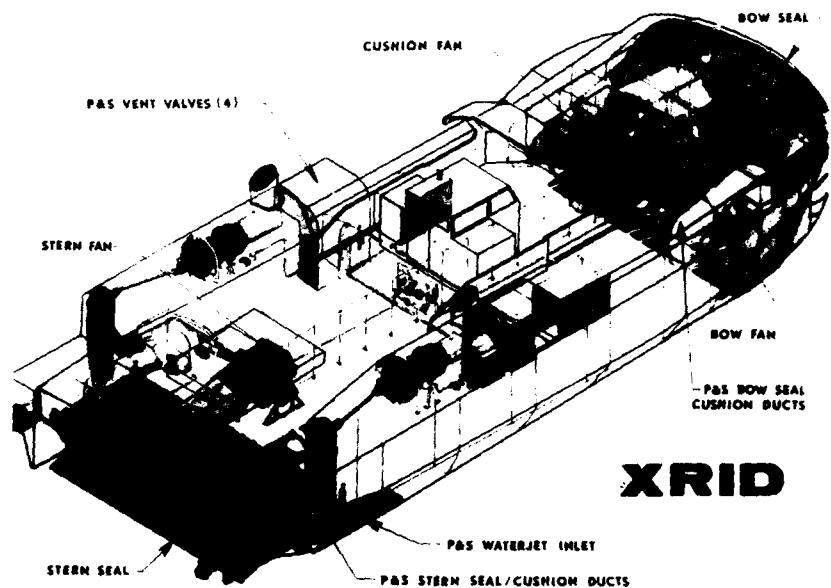


Figure 1. U.S. Navy XR-1D Surface Effect Ship

Upon completion of the XR-1D program, a prototype microprocessor-based control unit and a set of flow control vent valves were developed and subjected to rigorous shipboard testing on the U.S. Navy's latest acquisition, the 200-ton SES-200 (Figure 2). The SES-200 tests have included extensive operation in the Atlantic Ocean in sea states up through high Sea State V.

This paper provides a succinct summary of these research and development programs. The paper is divided into five sections. Mechanisms causing SES heave motion are discussed in the first section. In the next section, the overall approach used to derive ride control algorithms is outlined. Specific details of the SES math model formulation and control algorithm derivation are presented in the third and fourth sections, respectively. The fifth section describes the prototype hardware developed for the SES-200 and presents test results illustrating RCS effectiveness.

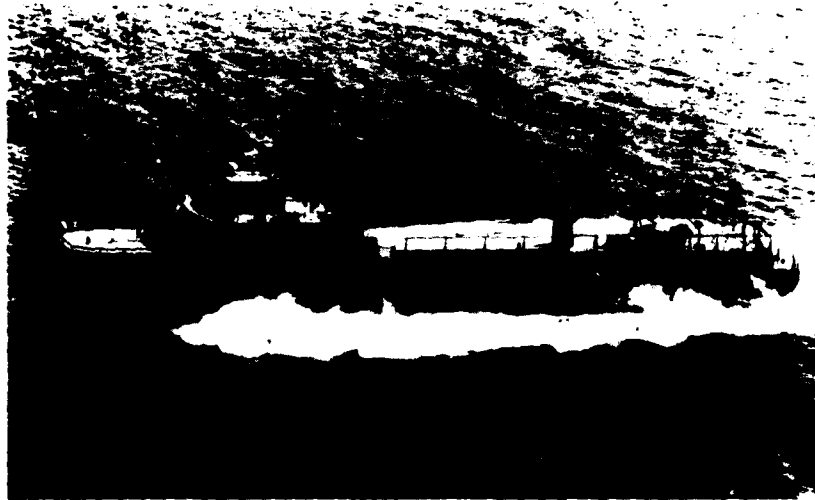


Figure 2. U.S. Navy SES-200 Surface Effect Ship

SES HEAVE MOTION

During cushionborne operation, air pressure supplied by lift fans supports most of the surface effect ship's weight. Pumping is the term used to describe the passage of waves through the SES's air cushion. In this compressible process, the rate-of-change of cushion volume is converted to a rate-of-change of cushion pressure in accordance with the gas laws. In moderate seas, these pressure changes do not exhibit significant variation along the length of the cushion, and therefore they primarily produce heave motion.

The bow and stern seals are flexible structures which ideally should track the rough water surface to maintain a uniform rate of air leakage. In practice, the seals may be deformed by the waves or they may be unable to compensate for relative motion between the craft and the water surface as they are restrained from dropping below the sidehull keel line. Therefore, the cushion leakage is not always uniform and the wave-induced variation in leakage also causes heave motion.

The SES's pneumatic suspension system produces a heave period that is much shorter than that of a displacement ship of comparable size. Heave damping decreases with increases in the slope of the lift fan pressure/flow curve at the nominal operating point (Figure 3). Flatter fan curves increase damping and steeper fan curves reduce damping.

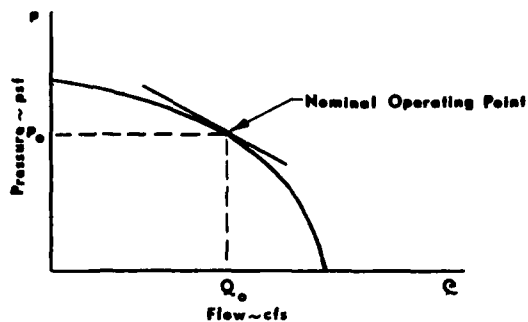


Figure 3. SES Lift Fan Pressure/Flow Curve

Owing to the SES's short natural heave period, synchronism (or tuning) between the seaway period of maximum energy and the SES heave period occurs during high speed operation. In the case of both the 40-knot XR-1D which has a 0.3 sec heave period and the 30-knot SES-200 which has a 0.5 sec heave period, tuning occurs during head sea operation in low and moderate sea conditions.

As the heave damping is fairly low (typical values range from 0.1 - 0.2), tuning results in a relatively bouncy ride which has been referred to as the "heave bounce" or "cobblestone effect". A Ride Control System (Figure 4) regulates the cushion pressure by using cushion vent valves and/or fan inlet guide vanes (IGVs) to modulate the mass of air in the cushion. IGVs modulate the cushion inflow which increases the damping by flattening the fan curve (Figure 5).

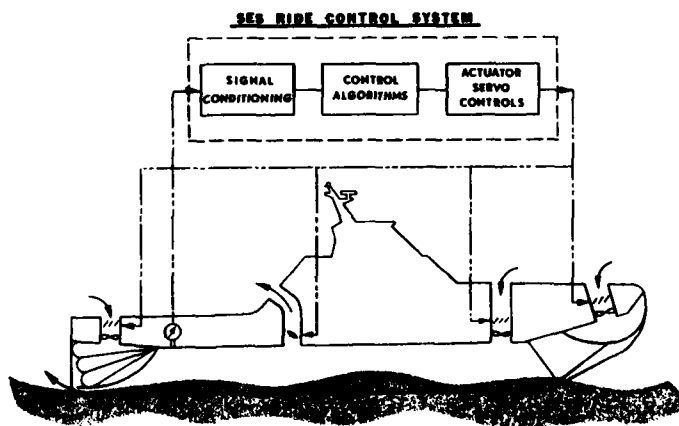


Figure 4. SES Ride Control System

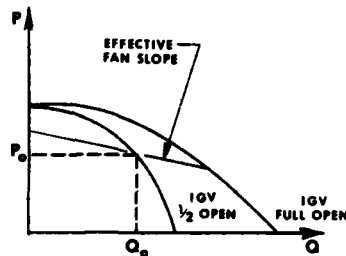


Figure 5. Effective Fan Slope During IGV Operation

Vent valves produce the same effect as IGVs by modulating the cushion outflow instead of the inflow.

OVERALL APPROACH TO CONTROL LAW DEVELOPMENT

The control law for the RCS is synthesized using the LQG (Linear Quadratic Gaussian) method of modern control theory. This approach assures us of arriving at a control law which is, within reason, optimal within the class of linear regulator laws.

The synthesis procedure requires that we first develop an accurate, linear, math model of the craft dynamics. Due to the necessity of including the acoustic modes of the air supply system in the model, the math model for SES is of rather high order. As a result, the LQG method yields a control law which is also of high order.

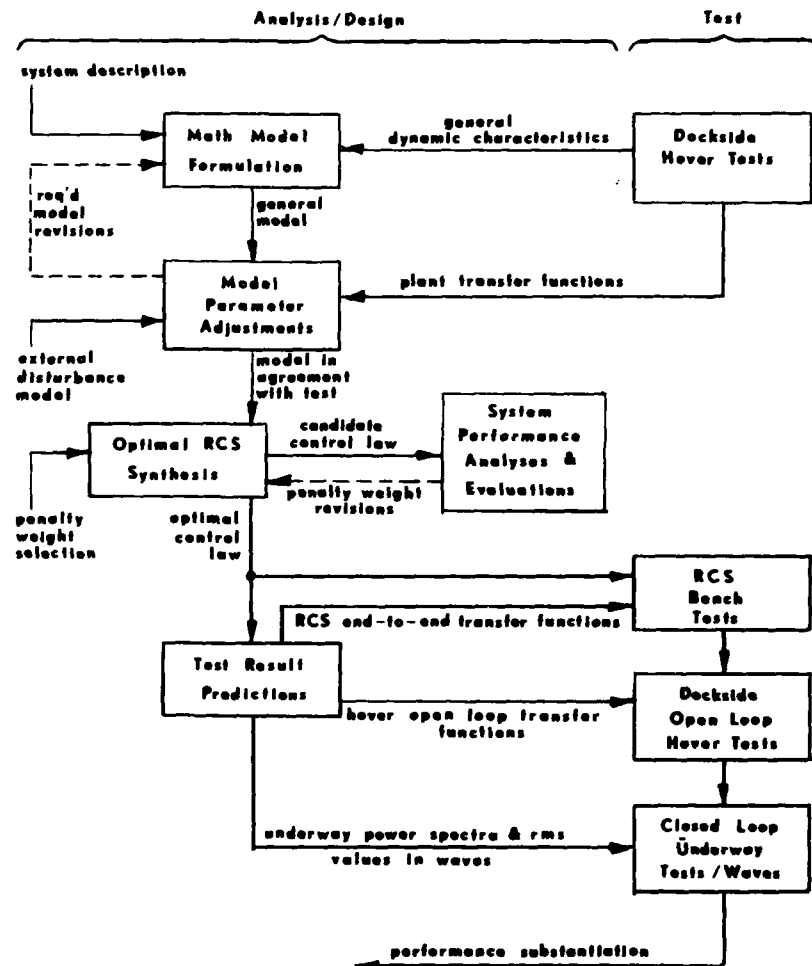
The overall procedure employed is indicated schematically on Figure 6. The first step is to select a general form for the math model and establish the requisite equations. To assure sufficient accuracy for use in the synthesis procedure, the model is expressed in terms of a number of free parameters which are adjusted to bring the model into agreement with test data. The test data employed are generally craft transfer functions with actuator commands as inputs. They are obtained by dockside tests with the craft in a hovering condition.

The model parameters are adjusted to match the test data by means of a non-linear, least-squares procedure. The parameter values minimizing the sum of squared errors are estimated iteratively. The errors are deviations of the model transfer function real and imaginary parts from the measured values at various frequencies.

The math model is then corrected to underway operating conditions and extended to include the responses to external disturbances (waves).

The objective function (penalty) weights are then selected for the LQG procedure and the optimal regulator law corresponding to this math model and weights is derived. As will be discussed in the next section, it is necessary to iterate the LQG procedure to obtain a set of weights yielding a control law which satisfies the system constraints on actuator motions, etc., and produces satisfactory

FIGURE 6 OVERALL CONTROL LAW DESIGN & EVALUATION PROCEDURE



system response characteristics. After each iteration, the system performance is evaluated for the given law and, as necessary, the penalty weights are adjusted to improve the performance or to satisfy the constraints for the next iteration.

Detailed performance analyses of the final law are made, including standard deviations, power spectra, etc., for final evaluation. Predictions of test results for bench tests, open-loop hover tests, and underway tests in waves are also made.

The control law is then incorporated into the RCS. Usually, laws are derived for several sea conditions and these are operator selectable underway from the console.

Verification of proper implementation of the control laws is obtained by bench tests. The predicted open-loop characteristics are verified with dockside hover tests including the RCS. Then underway tests in waves with and without RCS are conducted to substantiate the overall system performance.

FORMULATION OF THE SES DYNAMICS MODEL

The overall procedure used to establish the math model is illustrated on Figure 7. The steps required will be described briefly using the XR-1D math model as an example.

Various types of modelling can and have been used for this purpose. We shall restrict attention to the so-called topological model. This is a direct physical representation of the craft such as one might employ in an analytical determination of dynamic properties. This approach offers advantages in terms of model interpretation.

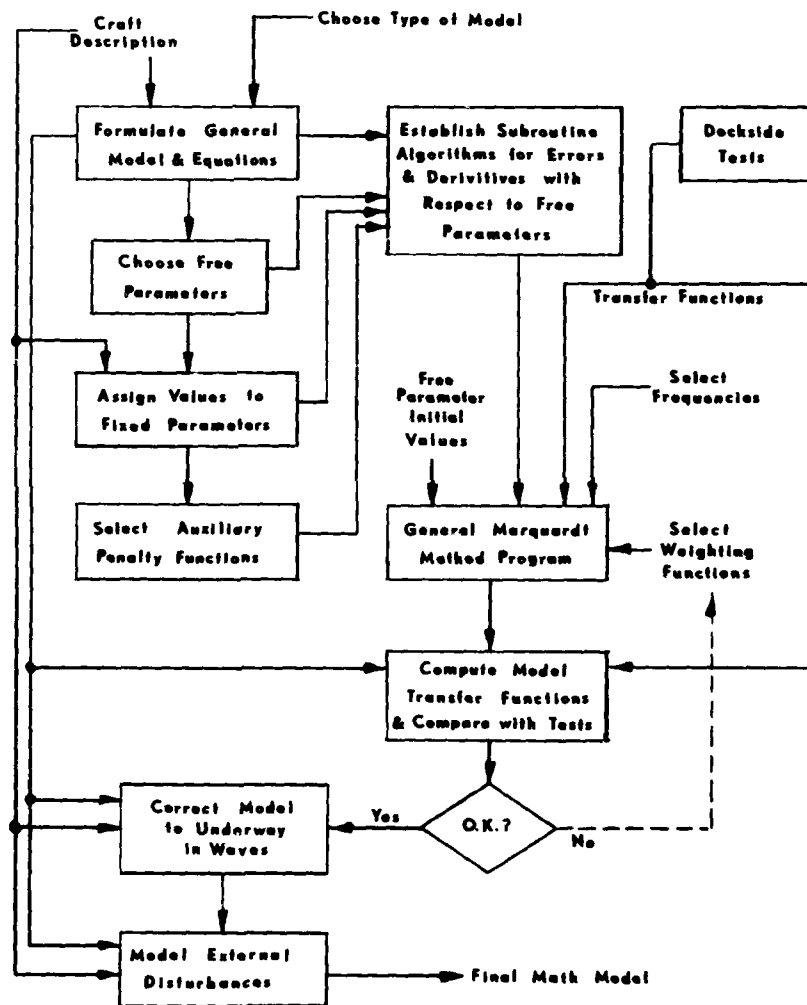
Topological modelling is preferred if the resultant model order is not too high. The model order can, however, get quite large when acoustic effects are taken into account for large SES such as the SES-200; this is due to the extensive partitioning of the lift air system needed for an adequate lumped-parameter model. In such cases, it is more practical to use a canonical state variable model.

The general form of the XR-1D math model and the state variable assignments are shown on Figure 8. Ride control is effected by use of the cushion vent valves.

It has been found that the model for this size SES should be valid out to around 20 Hz. Otherwise, the control law synthesis procedure cannot deal adequately with all the closed-loop stability problems arising, primarily, from the lift air system acoustic modes. In order to establish a lumped-parameter model of the acoustics, the cushion in the XR-1D example was partitioned into fore and aft sections to be modelled separately. This is necessary to maintain the maximum dimension of component segments to be no more than about a quarter of the acoustic wavelength at 20 Hz. Larger craft (e.g., the SES-200) would require additional transverse cushion partitions and also longitudinal partitions in the cushion and seals.

The model required is a linear, state variable representation. The linearization is performed about the mean conditions, i.e., all state variables are measured from the mean (and therefore have zero mean values).

FIGURE 7 MATH MODEL DEVELOPMENT PROCEDURE



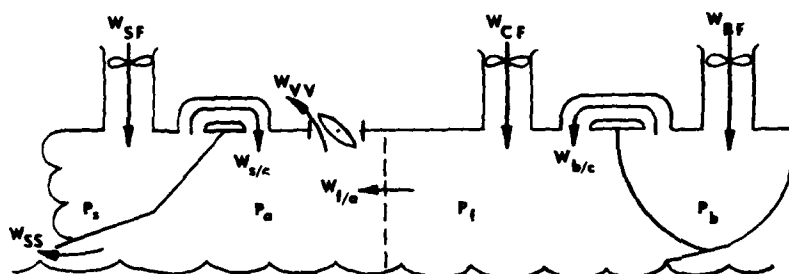


FIGURE 8

OPTIMAL VENT VALVE CONTROLLER
STATE VARIABLE ASSIGNMENTS

RIGID BODY MOTIONS	$\begin{cases} x_1 = \Delta \dot{z} & (\text{HEAVE VELOCITY}) \\ x_2 = \Delta z & (\text{HEAVE POSITION}) \\ x_3 = \Delta \dot{\theta} & (\text{PITCH RATE}) \\ x_4 = \Delta \theta & (\text{PITCH ANGLE}) \end{cases}$
SEAL AND CUSHION PRESSURES	$\begin{cases} x_5 = \Delta P_a & (\text{AFT CUSHION PRESSURE}) \\ x_6 = \Delta P_i & (\text{FWD CUSHION PRESSURE}) \\ x_7 = \Delta P_b & (\text{BOW SEAL PRESSURE}) \\ x_8 = \Delta P_s & (\text{STERN SEAL PRESSURE}) \end{cases}$
FLOW INERTIA TERMS	$\begin{cases} x_9 = \Delta W_{I/a} & (\text{MASS FLOW, FWD TO AFT CUSHION}) \\ x_{10} = \Delta W_{s/c} & (\text{MASS FLOW, S.S. TO AFT CUSHION}) \\ x_{11} = \Delta W_{b/c} & (\text{MASS FLOW, B.S. TO FWD CUSHION}) \\ x_{12} = \Delta W_{CF} & (\text{MASS FLOW, CUSHION FAN}) \\ x_{13} = \Delta W_{BF} & (\text{MASS FLOW, BOW FAN}) \\ x_{14} = \Delta W_{SF} & (\text{MASS FLOW, STERN FAN}) \\ x_{15} = \Delta W_{SS} & (\text{MASS FLOW, UNDER STERN SEAL}) \\ x_{16} = \Delta W_{VV} & (\text{MASS FLOW, THROUGH VENT VALVES}) \end{cases}$
ACTUATOR DYNAMICS	$\begin{cases} x_{17} = \Delta VV & (\text{VENT VALVE POSITION}) \\ x_{18} = \Delta \dot{VV} & (\text{ACTUATOR HYDRAULIC FLOW RATE}) \end{cases}$

As indicated on Figure 8, the state equations are divided into four types. First, there are the usual rigid body motion equations (for RCS work, only the pitch and heave modes are needed). Then, there are equations representing the compression of air within the various volumes of the air system; the mass of air in the volume is found by integrating mass fluxes of air and adiabatic compression is assumed. There are also mass flow rate equations incorporating the effects of head losses and inertias of the flows between various volumes of the air system and flows to and from the surrounding atmosphere (it is the presence of the inertia terms which characterizes the model as acoustic); the fan characteristics are also contained in these equations. Finally, there are the equations representing the effector (vent valve actuator) dynamics. The resultant XR-1D model is of 18th order.

Note that this model does not include the leakage flows under the sidehulls and bow seal as state variables. The time constants associated with these flows are so small that it is permissible to assume that the flow rates go instantaneously to their steady-state values. Separate state equations are not required for these terms.

The primary parameters left free for adjustment to fit the test data are the acoustic parameters representing the ratios of cross-sectional area to effective length of flow passages. Most of the remaining parameters are considered to be well known from geometrical and mass data or from air system tests.

The test data used to adjust the math model parameters are usually obtained from dockside tests with the craft tethered on-cushion in a hover condition. The vent valves are biased open and the cushion pressure adjusted to slightly below normal operating conditions so that the sidehulls are well immersed over their entire length and the seals are depressed into the water. In this condition, there is no appreciable leakage from the cushion other than through the vent valves and no change in leakage with small changes in heave position. This avoids uncertainties with respect to distribution of the air flows between various leakage paths. It also avoids the non-linearities associated with broaching of the seals and sidehulls as the craft heaves. Corrections to the model to adjust to underway operating conditions are made after fitting the model for the hover condition.

The tests are conducted by driving the vent valves sinusoidally at modest amplitudes with a signal generator. The driving frequency is varied over the desired range (0.1 to 20 Hz for the XR-1D) and the data are reduced by Fourier analysis to produce transfer functions. The primary transfer functions obtained, i.e., those used in fitting the math model, are generally cushion and seal pressures.

The adjustment of the math model is accomplished by a non-linear least-squares method. Such methods are all iterative and seek to minimize the sum of the squares of the errors by successive adjustment of the free parameters. The particular method employed here is a modified Marquardt method (c.f. Reference 1 or 2).

A set of frequencies is selected. Then, for each frequency, the differences between the model predictions and the measured values of the real and imaginary parts of various transfer functions are taken as basic errors. To permit greater control of the matching process,

arbitrarily assignable weights are included as factors on the individual error terms. The errors are also divided by the measured absolute value of the transfer function at that frequency (this corresponds to fitting in terms of db rather than amplitudes).

The general model equations are specialized to the dockside test conditions and a subroutine is written for the general Marquardt program. This routine accepts the current values of the free parameters as input together with the test data to be matched. It returns the component error values for the model with those free parameter values together with the derivatives of the component errors with respect to the free parameters. The general Marquardt program uses these data to produce the next estimate of the free parameter values to minimize the objective function.

The procedure is started with some initial guesses for values of the free parameters. If the procedure is allowed to change these parameter values arbitrarily in the early stages of the iterations, it can produce an unreasonable set of parameter values and may find some local minimum error condition which is very far from the desired result. To prevent this, error terms of the deviations from the initial parameter values are also formed and these are included in the objective function. The procedure is allowed to obtain a rough result with these penalty constraints. Then the estimates are recentered (i.e., the errors are now measured from the new values) and the procedure restarted. This is repeated until a satisfactory solution is obtained.

For the XR-1D, the data chosen for the model matching were the transfer functions of the bow and stern seal pressures and the aft cushion pressure. 32 frequencies were selected for use in the fit. With fits made to the real and imaginary parts of three transfer functions, this gave 192 points to be fit by adjustment of 12 free parameters.

Comparisons of the test data and the predictions from the adjusted math model are shown on Figure 9. The agreement is generally quite good, particularly at the higher frequencies. The superiority of the fit at high frequency as compared to low is the intentional result of choosing large weighting factors for the higher frequencies. Experience has shown that the closed-loop system tends to exhibit less stability than predicted by the model due to modelling errors near the natural frequencies of the higher acoustic modes. We therefore elected to sacrifice some accuracy at lower frequencies to improve the model match near these acoustic modes.

The greatest discrepancies between the test data and the model predictions are in the phase data for the seal pressures at frequencies well below the heave natural frequency; this frequency range is not of great interest to the RCS control law design problem. There is also a disagreement in phase for the bow seal pressure at frequencies beyond 13 Hz or so, but this did not have any deleterious effect on the control law design process.

The most important transfer function to match is, of course, that for cushion pressure since it is basically this pressure which must be regulated to achieve a suitable ride. As can be seen on Figure 9, the fit to this transfer function is quite good throughout the frequency range in both phase and amplitude.

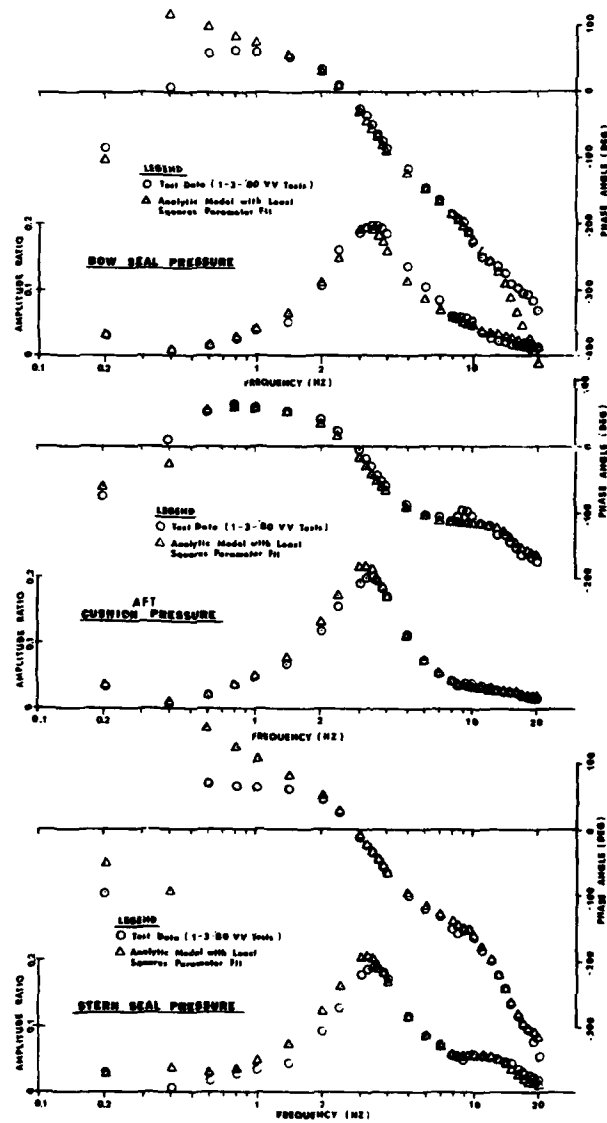


Figure 9. Comparison of Measured and Computed XR-1D Cushion and Seal Pressure Responses to Vent Valve Command

It is important to note that the adequacy of the math model for the purpose at hand is properly judged by the fit to the open-loop transfer function(s) appropriate to the feedback measurement(s) to be employed (in this case cushion and seal pressures). That is all that really affects the control law synthesis procedure. This model is clearly quite satisfactory for the regulator design synthesis problem.

We must now correct the model to represent realistic, underway conditions. The condition of interest is the design point condition for the control law. The condition of relatively high speed in moderate waves is usually a critical one for SES heave motion. The design point is merely the condition at which the controller is optimized. The same basic law will work satisfactorily over an appreciable range of wave heights encompassing the design conditions.

The primary differences between the hover condition and a normal operating condition in waves are as follows:

1. The vent valve bias opening is normally less than used in the hover condition. The larger opening for the hover tests is to allow the fan to operate at normal flow without inducing leakage under the seals and sidehulls.
2. Fan speeds may be different.
3. Sidehull hydrodynamic effects become more important due to craft forward speed.
4. Leakage flows under the seals and sidehulls in waves must be accounted for.

Corrections for the first three items are quite straightforward and affect clearly identified derivatives in the model formulation. The last effect is more difficult to handle; the leakages due to local broaching effects are non-linear and are dependant on the correlation functions between instantaneous wave displacements and craft motions.

The effects of additional leakages in waves are estimated by a method which formulates the cross-correlations of the leakage areas with the wave displacements at some reference point and with the craft motions. Wave propagation effects are accounted for in the formulation. These correlations are converted to cross-spectra by Fourier transformation. The best linear estimates of the requisite transfer functions are determined from the power spectra and cross-spectra in exactly the same manner used to extract transfer functions from test data. This yields a linear model which is the best fit, in the least-squares sense, to the non-linear dynamical characteristics. These transfer functions for the additional leakage terms are then incorporated into the math model.

It remains to incorporate the effects of external (wave) input disturbances into the model. The primary effect of the waves is to drive the cushion and seal volumes. Compression of the air in the cushion provides the primary heave forces arising from waves. The effects of sidehull buoyancy and hydrodynamics are also present, but are of considerably less significance. Therefore, we must modify the math model to accept external disturbance inputs which drive the cushion and the seal pressure state variables in an appropriate

manner. The formulation of the LQG method requires that these disturbances be generated by filtering white noise and inserting the filtered signals at the appropriate points in the model.

There is a problem in deciding just what constitutes an appropriate disturbance model. It is possible, for instance, to provide a very detailed simulation of the actual disturbance spectra for a well defined operating condition (the control law design condition). This tends to produce a very cumbersome math model. Also, we are not really interested in designing a control law highly optimized to a precisely defined condition. We would prefer a law which, in some sense, was optimized for best average performance over a representative range of likely operating conditions for the craft, including variations in speed, heading, wave height, etc. This suggests that the power spectra of the forced, RCS-off motions represent an average of many specific spectra or an envelope of such spectra. The most important point is to assure that the model disturbance spectra have appreciable power at frequencies where the control law must be effective.

For the XR-1D, it was found that a model applying independent, uncorrelated, white noise disturbances passed through first-order, low-pass, shaping filters to each of the two cushion pressures and to the bow seal pressure yields acceptable results. The amplitudes of the white noise sources are varied to seek to attain a suitable disturbance condition in terms of the standard deviations and shape of the vertical acceleration and cushion pressure power spectra.

With the addition of the three first-order shaping filters, the complete XR-1D math model is of 21st order. This is as high an order as we care to handle using the LQG approach as the matrices involved tend to become too complex and the iteration times become excessive. Even here, the LQG method requires the solution of sets of 231 simultaneous, non-linear, first-order, time-domain, differential equations to arrive at a control law.

DERIVATION OF THE RIDE CONTROL ALGORITHM

The application of control law synthesis techniques, such as the LQG method, to the design of realistic laws for complex dynamical systems is considerably less straightforward than most papers on the subject would seem to indicate. The principal problem is simply that the penalty weights to be used in the objective function (the function which the control law is designed to minimize) cannot be specified in advance, but must be found iteratively, usually by trial-and-error methods. The overall procedure is summarized on Figure 10.

The difficulty in specifying the penalty weights does not arise from formulation of the primary quantity to be minimized. In cases of interest here this is merely rms vertical accelerations or cushion pressure standard deviations. Such quantities are easily incorporated into the objective function. Instead, the problems arise in specifying the constraints on the design. These constraints include maintaining vent valve actuator motions and rates within design limits. Experience has shown that laws designed without such constraints subject the vent valve/actuator system to such broad-band excitation that fairly rapid deterioration of the hardware results.

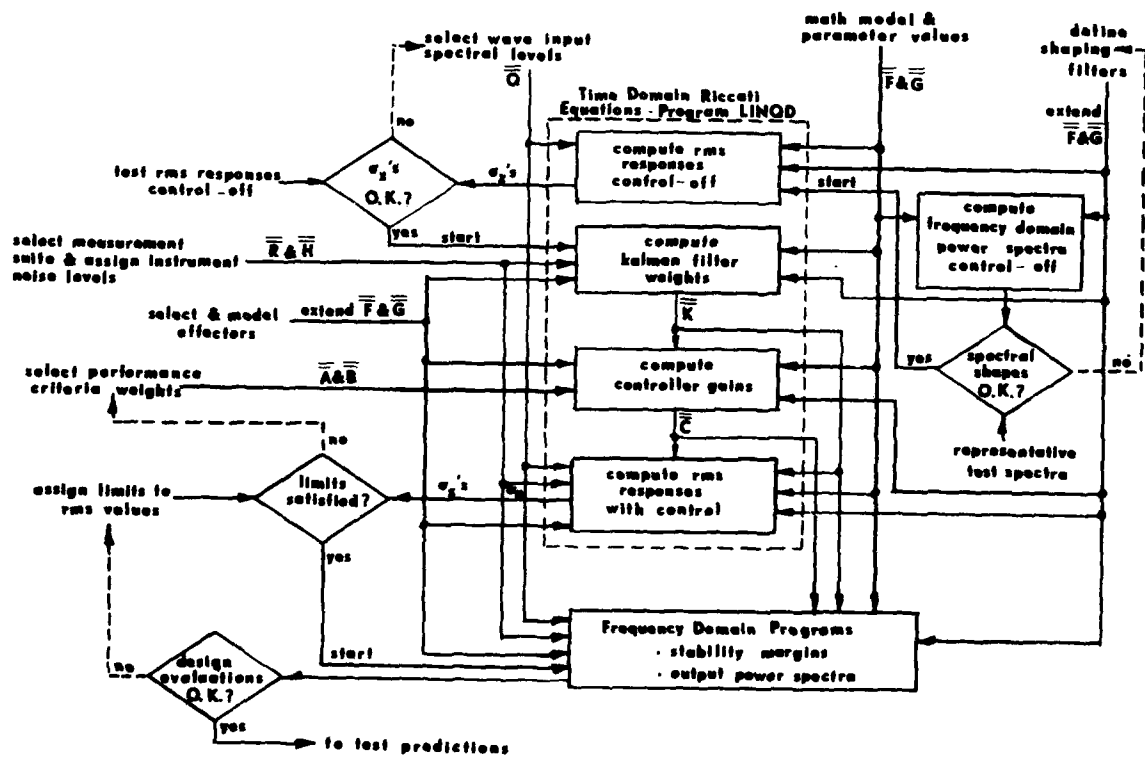


FIGURE 10 CONTROL LAW SYNTHESIS PROCEDURE

As indicated on Figure 10, the LINGD Program, which implements the LQG method, is used to compute the standard deviations of state variables and auxiliary functions such as the rms vertical acceleration level. Vertical acceleration is not a state variable, but it is expressible as a linear combination of state variables. The program also produces Kalman filter weights (the Kalman filter provides optimal estimates of the complete state vector from incomplete, noise contaminated measurements) and the controller gain matrix for a state variable feedback controller. The optimal controller for the system with incomplete and noisy feedback measurements is then found as a cascade of the Kalman filter and the state variable feedback law. The implementation used employs direct time-domain integrations of the various matrix Riccati equations. The steady state regulator control law (which is all that is required for our application) is found by extending the integrations in time until steady conditions are reached.

The mathematical model gives the state equations in the form

$$\dot{\hat{x}} = \hat{F}\hat{x} + \hat{G}\hat{u} + \hat{w}$$

where $\hat{x}$ is the state vector, $\hat{u}$ is the vector of control signals and $\hat{w}$ is a vector of white noise disturbance inputs with covariance matrix $\hat{Q}$.

The Kalman filter has the form

$$\hat{\hat{x}} = \hat{F}\hat{\hat{x}} + \hat{G}\hat{u} + \hat{K}(\hat{z} - \hat{H}\hat{\hat{x}})$$

where $\hat{\hat{x}}$ is the filter estimate of the state vector and $\hat{z}$ is the vector of feedback measurements given in terms of the measurement matrix $\hat{H}$ as

$$\hat{z} = \hat{H}\hat{\hat{x}} + \hat{v}$$

with $\hat{v}$ a white measurement error noise vector with covariance matrix $\hat{R}$. The Kalman filter weighting matrix $\hat{K}$ is found by the LINGD Program.

The control signal is given by

$$\hat{u} = -\hat{C}\hat{\hat{x}}$$

where the control weight matrix $\hat{C}$ is also given by the LINGD Program. The matrix $\hat{C}$ is chosen to minimize the expected value of the objective function

$$J = \frac{1}{2} \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T \begin{bmatrix} \hat{x}^T & \hat{u}^T \end{bmatrix} \begin{bmatrix} \hat{A} & \hat{B} \\ \hat{B}^T & \hat{E} \end{bmatrix} \begin{bmatrix} \hat{x} \\ \hat{u} \end{bmatrix} dt$$

The matrices $\hat{A}$, $\hat{B}$ and $\hat{E}$ are the penalty weight matrices.

In addition to the LINGD Program, various frequency domain programs are needed to produce power spectra and conduct stability studies (Bode plots, Nyquist diagrams, etc.) of the closed loop system once control laws are obtained. These programs employ the same state variable math model as the LINGD Program and use the Kalman filter and state variable feedback laws passed directly to disk files from that program.

laws for which the open-loop gain approaches zero sufficiently rapidly at high frequencies. Using higher levels of measurement noise spectra tends to induce the desired effect by making the measurements appear unuseable at high frequencies (since the noise is assumed white).

An example of the results obtainable with the LQG method are shown in Figure 11 where the predicted heave acceleration power spectra with and without ride control are plotted for the design condition. As shown, the RCS achieves considerable attenuation of the energy in the vicinity of the spectral peak which corresponds to the vessel's 3.2 Hz heave natural frequency. At frequencies well removed from the heave mode the system is ineffective. This is the desired result as we do not require that the system control these low amplitude responses.

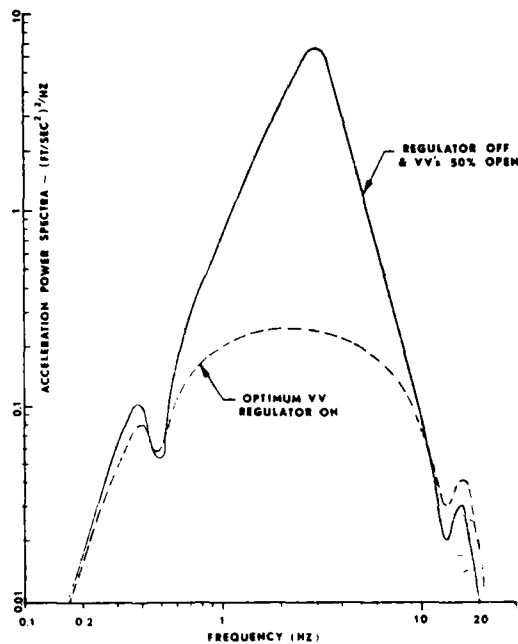


Figure 11. Predicted XR-1D Heave Acceleration Power Spectra

Figure 12 is the predicted Nyquist diagram for the XR-1D RCS. As can be seen the system stability is quite good and adequate stability margins are maintained at all frequencies. Also the rapid decrease of the open-loop gain to zero at high frequencies is apparent and tends strongly to offset any possible deleterious effects of high frequency modelling errors.

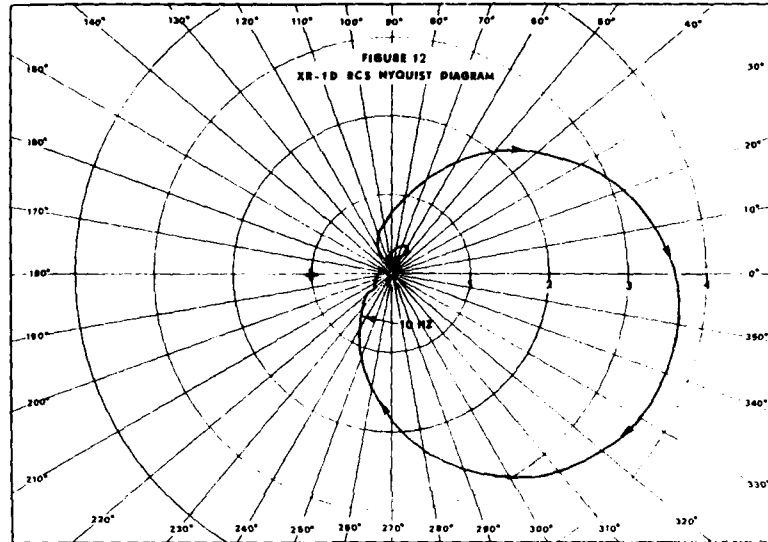


Figure 12. XR-1D RCS Nyquist Diagram

The predicted regulator effectiveness is further illustrated by the data shown on Table 1. Here, the standard deviations of the state variables with and without RCS are shown together with the desired upper limits imposed on some variables. The rms heave acceleration and the control signal standard deviation are also shown. The RMS heave acceleration is more than halved by the use of the RCS; the effect would be even greater if the comparison were made with the vent valves-closed, RCS-Off condition instead of the RCS-Off vent valves-50% open condition shown in Figure 11.

Heave acceleration test data showing the measured performance of the XR-1D RCS during 25-knot head sea operation in Sea State II are shown in Figure 13. The data are plotted in 1/3rd octave band format for three test points along with the acceleration limits given in ISO 2631-1978, Guide for the Evaluation of Human Exposure to Whole Body Vibration. With the RCS-Off and the vent valves closed (Test Point 6), the accelerations exceed the 4-hr criteria at the vessel's 3.2 Hz heave resonance. Opening the vent valves with the RCS-Off (Test Point 5), reduces the acceleration levels below the 4-hr criteria. Turning the RCS-On (Test Point 4) reduces the accelerations to below the 24-hr criteria at all frequencies.

Table 1
Standard Deviations With and Without Ride Control

State Variable	Standard Deviation RCS-Off	Standard Deviation RCS-On	Limit Value	Units
z	0.3766	0.1618		fps
$\dot{z}$	0.0873	0.0766		ft
θ	0.0063	0.0053		rad/sec
$\dot{\theta}$	0.0029	0.0025		rad
Pa	15.69	6.29		psf
Pf	16.07	7.84		psf
Pb	17.50	11.27		psf
Ps	16.36	8.12		psf
Wf/a	0.4009	0.5162		slugs/sec
Ws/c	0.1121	0.0875		slugs/sec
Wb/c	0.2972	0.3203		slugs/sec
Wcf	0.0754	0.0351		slugs/sec
Wbf	0.0718	0.0450		slugs/sec
Wsf	0.0349	0.0166		slugs/sec
Wss	0.1339	0.1073		slugs/sec
Wvv	0.1442	0.5002		slugs/sec
vv	--	0.9973	1.415	sq. ft
$\ddot{z}$	--	4.876	4.87	c/s
z	0.1821	0.0661		g's
U	--	2.453	2.44	v

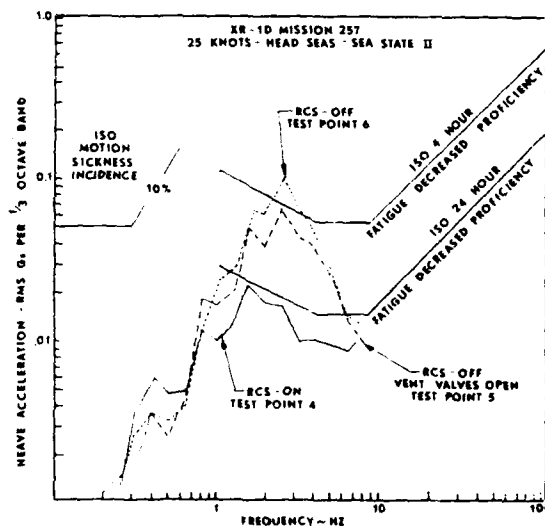


Figure 13. XR-1D Heave Acceleration 1/3rd Octave Band Data, Sea State II

PROTOTYPE RIDE CONTROL SYSTEM

In the middle of 1981, the U.S. Navy initiated the development of a prototype microprocessor-based RCS using vent valves for flow control. This system was developed over a one (1) year period and was installed on the Navy's SES-200 surface effect ship (Figure 2) in late 1982.

The SES-200 is a testcraft that is being used to demonstrate the performance, seakeeping and stability characteristics of platforms with higher length-to-beam ratios than the previous generation of Navy SES (Reference 3). These "high length-to-beam SES" operate below the primary wave making drag hump and hence do not have to be propelled through a high drag speed regime in order to reach efficient cruise speeds.

System Design

A functional block diagram of the SES-200 RCS is shown in Figure 14. In brief, the system accepts up to four pressure signals for control law feedback, performs a sampled data control algorithm, and outputs a single control signal which drives four vent valves. The system is designed for unattended operation and contains logic for performing self-checks, fault monitoring, and smooth shutdowns.

The broad aims of the design were to develop a no-frills system that would serve as a prototype for future Navy SES. This meant that the software had to be structured such that the system could be used on both larger and smaller SES without rewriting the resident controller program.

Electronic Control Unit

Hardware

The system electronics, operator controls, and display are all housed in a single compact control unit (Figure 15). Electronic components are mounted on modules and housed in a card cage and backplane which conform to MIL-STD-28787 (USN Standard Electronic Module - SEM).

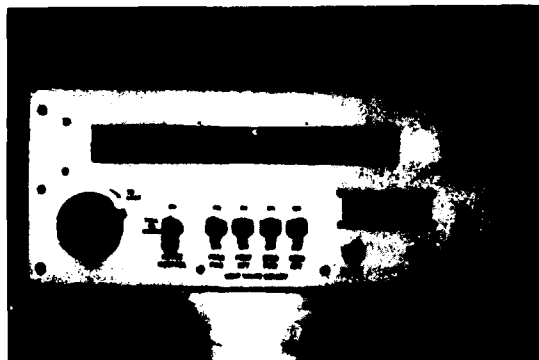


Figure 15. Electronic Control Unit

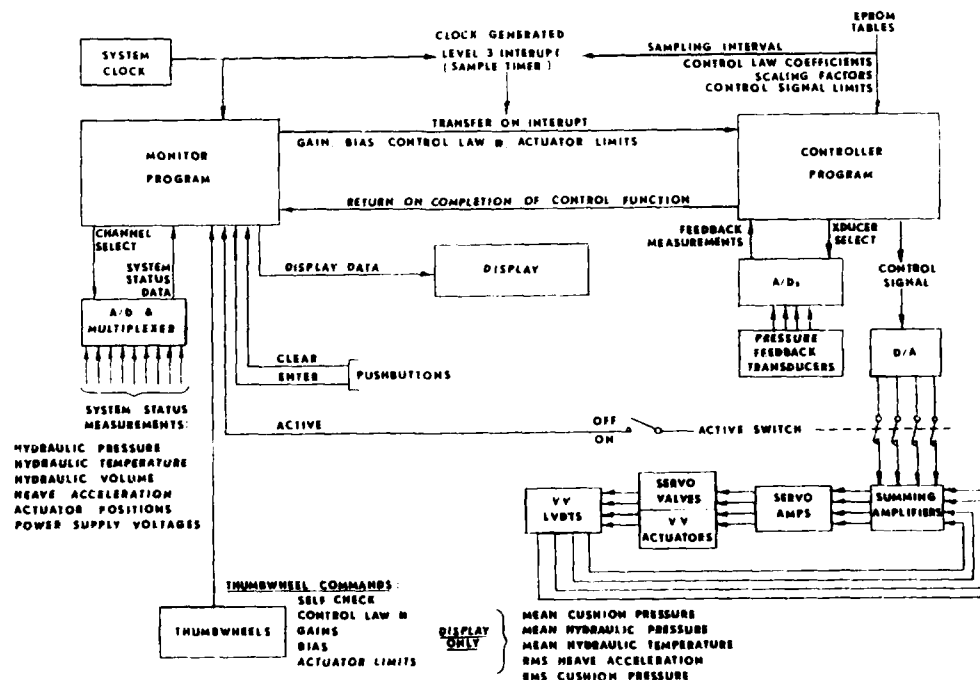


FIGURE 14
RCS FUNCTIONAL BLOCK DIAGRAM

This arrangement permits the components which perform various functions (A/D, D/A, CPU, RAM, ROM, etc.) to be packaged on replaceable units (Figure 16). There are a total of forty-four (44) separate modules in the system of which eleven (11) are analog and thirty (33) are digital. Only twenty three (23) different types of modules are used.

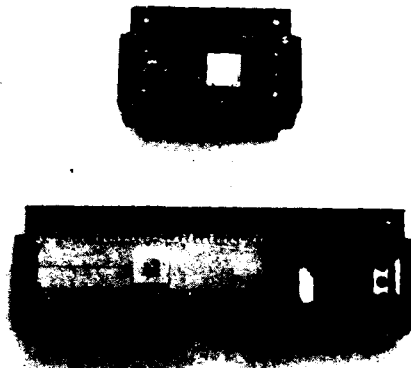


Figure 16. Electronic Modules

Commercial components were used in the prototype to meet schedule constraints. However, the various devices used can be purchased in militarized equivalents if this becomes a future requirement.

A 16-bit TI 9900 microprocessor is used in conjunction with a 16 x 16 multiplier/accumulator to provide the speed and accuracy required for control algorithm computations.

The multiplier/accumulator, A/Ds, D/A and other support chips which communicate with the CPU via the data bus are memory-mapped. Switches, pushbuttons and other single bit devices that communicate with the CPU are input via the 9900's communications register unit (CRU).

System memory consists of 6K (x16) words of read only storage and 2K (x16) words of random access storage. There are eight memory modules each of which contains one 2K x 8 bit RAM or ROM chip. Four ROM modules are devoted to program storage and two modules are dedicated to coefficient storage. As will be discussed subsequently, the system software is general purpose in nature and only the ROM modules used for coefficient storage need be changed to implement different control laws or to use the system on other SES.

The four cushion pressure feedback signals are input via separate 12-bit A/Ds while measurements which are monitored to verify proper system operation are input via a multiplexed 12-bit A/D. These monitor measurements include hydraulic pressure, temperature and volume, heave acceleration, actuator positions, and the power supply voltages. Spaces are provided in the control unit

for analog pre-sampling filters, but these are not presently implemented.

Software

The system software was developed in 9900 assembly language using structured programming techniques. Symbols were used to define all CRU and memory mapped addresses, all interrupt and transfer vectors and all fixed and variable data locations. This arrangement permitted remapping ROM and RAM and changing hardware addresses during software and hardware development.

All fixed data values such as transducer scale factors, control law coefficients, gain limits, alarm values, etc. are contained in a separate data block. Since symbols are used to define these locations, this data block can be assembled separate from the main program. The data block resides in the pair of 2K x 8 ROM modules that are used for coefficient storage. This makes the system general purpose in nature, i.e., it can be used on various size cushion-supported craft merely by changing coefficients stored in the ROM modules.

The system operates under the direction of a short main program which takes care of initialization, and then sets the control law cycle timer. Next, a monitor routine is called which is composed of a number of subroutines. These routines acquire inputs from the control panel and from auxiliary system measurements (e.g., hydraulic pressure). The inputs are checked against limits stored in the data tables to see if a default value should be used or if a fault condition exists. Other monitor functions include updating the console display and computing means and standard deviations of all feedback and auxiliary measurements. The monitor subroutines also perform smoothing operations on system gain and vent valve bias to ensure smooth changes in response to operator entries from the control panel. All monitor functions are performed in the processor's "spare time" (i.e., when it is not performing the control algorithm).

The controller routines implement the control algorithm when the sample timer generates an interrupt request. The software implementation selected is designed to minimize undesirable phase lags due to delays between acquisition of the latest feedback signal(s) and the outputting of the corresponding control signal. This is accomplished, in part, by precomputing all portions of the control algorithm which depend only on past data. When the next cycle is generated only an updating with the latest measurement is needed to obtain the control signal which is expressed in the form:

$$u_0 = b_0 y_0 + b_1 y_1 + \dots + b_q y_q + a_1 u_1 + a_2 u_2 + \dots + a_p u_p$$

where u is the control signal, y is the feedback measurement, and the subscripts refer to the sample time, i.e., (y is the feedback measurement acquired rT seconds earlier). Any linear control law can be placed in this form.

On receiving the interrupt signal, the following sequence of operations is initiated. First the necessary data and workspace pointers are stored to enable proper re-entry to the monitor loop. Then control transfers to the main program which resets the sample timer and interrupt flag and calls the controller routines. These

routines acquire and use the latest data samples from the feedback transducers to update the control signal. Then limits are applied to the control signal to prevent mechanical contact between the vent valve louver or within the hydraulic actuator. The signal is properly scaled and output to the vent valve servo amplifiers via an 8-bit D/A. All control law computations are carried out in double precision (32-bit arithmetic).

The control tables (tables of previous feedback measurements and previous control signals) are then updated and those portions of the control algorithm depending only on past values are precomputed and stored to await updating at the next sampling time. Control then returns to the monitor loop which resumes its functions where it was interrupted.

The controller routines are not dependent on the number or type of control laws implemented. For each control law, the data tables include: the coefficients, the limiting values of gain and bias, and the default values of gain and bias. Any linear control law of 15th order relating no more than 4 inputs (feedback pressures) to one output (vent valve command) can be accommodated. Extension to multiple outputs and more inputs can be implemented. On the SES-200, the control laws presently in use are being computed at 200 pps (5 msec).

It is emphasized again that only these data tables need be changed to implement different control laws; neither the hardware nor software routines need be altered in any way.

Vent Valves

The vent valves were designed and built as modular units which are connected to the cushion via flow trunks. In the SES-200, the modules are bolted to short trunk stacks that extend approximately 2 ft above the weather deck. This deck mounted configuration was selected so that technical personnel can easily witness system operation. In a production SES, the vent valves would exhaust out the sides to maximize use of deck space for other purposes.

Each module (Figure 17) contains its own hydraulic actuator to drive the vanes, a servovalve to direct flow to the actuator, and a linear variable differential transformer (LVDT) for actuator position feedback. Self-aligning spherical roller and needle bearing are used throughout the mechanism to minimize radial internal clearances. These bearings eliminate noise and any chance of binding due to the combination of forces acting on the vanes.

The vanes are aerodynamically balanced so that cushion pressure will hold them closed when hydraulic and electrical power has been secured. Aerodynamic closing torque in this condition is sufficient to keep air from leaking between the seals. This is a fail-safe feature.

The Vent Valve Modules have been subjected to extensive analysis and testing to ensure there are no non-linear elements, e.g., flow and load limiting which will compromise RCS performance. Additionally, tests have been run to insure there is no dynamic vane overshoot when vent valve travel is limited by the RCS Electronics. In high sea states, system operation can approach a bang-bang mode without fear of mechanical contact between the vanes or within the

hydraulic actuator. The mechanism has been tested at frequencies in excess of 50 Hz to ensure structural integrity in the event of a high frequency RCS instability.



Figure 17. SES-200 Vent Valve Module Installation

Shipboard Operation

The system is activated by selecting a control law using thumbwheel switches located on the front panel (Figure 15). Control laws are provided for Sea State I/II conditions and Sea State III/IV conditions. Once the control law has been selected, the operator turns the "Active Control" switch to the "on" position.

The system gain and the mean Vent Valve opening or bias are automatically set at optimum values for the selected control law. If the operator desires, he can change the gain and bias values using the thumbwheels, but he cannot exceed limits which are stored in memory for each law. The system increases the gain and bias gradually; it takes about one (1) minute for the full effectiveness to be felt.

The system can be left unattended at all times. If a malfunction occurs, a warning message will be displayed and a programmed shutdown may be issued depending on the severity of the malfunction. The shutdown routine gradually reduces the system gain to zero, and closes the vent valves.

Using the front panel thumbwheel switches, the operator can display the mean values of hydraulic pressure and cushion pressure and the standard deviations of heave acceleration and cushion pressure. These standard deviations give the operator's quantitative data on the RCS's effectiveness.

Routine maintenance that the system requires involves greasing the bearings in the vent valve modules and inspecting the linkages for looseness. Additionally, the hydraulic system is inspected for leaks and the reservoir fluid level is checked.

The electronics do not require maintenance and there are no trim pots in the system to adjust. Self-check and fault monitoring routines can pin point failures in approximately half of the modules.

Sea Trials

The prototype RCS has been rigorously tested during sea trials in the Chesapeake Bay and in the Atlantic Ocean. During the 79-day span of the SES-200 Technical Evaluation Program (Reference 3), the vessel was underway conducting tests or transiting to and from test areas a total of 41 days. The total underway time was 530 hours during which the RCS was used for a total of 117 hours. The longest non-stop period of operation covered 17-hours during a 600 nautical mile transit from the Surface Effect Ship Test Facility at Patuxent River, MD to NOAA Buoy 'RY41001', which is located 200 miles due east of Cape Hatteras, N.C.

The system has been tested in Sea State I through high Sea State V and has been found to provide significant reductions in the heave accelerations in all cases.

Figure 18 illustrates the system's effectiveness during head sea operation at 25 knots in low Sea State II. Under these conditions, most of the wave energy is concentrated near the vessel's natural heave period. With the RCS-Off and the vent valves closed, the heave accelerations exceed the 4-hr habitability criteria at the vessel's 2 Hz natural heave frequency. With the RCS-On, the overall vertical acceleration standard deviation is reduced by 68% from 0.11 g's (the average value for the two RCS-Off points) to 0.045 g's and the accelerations are below the 24-hour criteria at all frequencies.

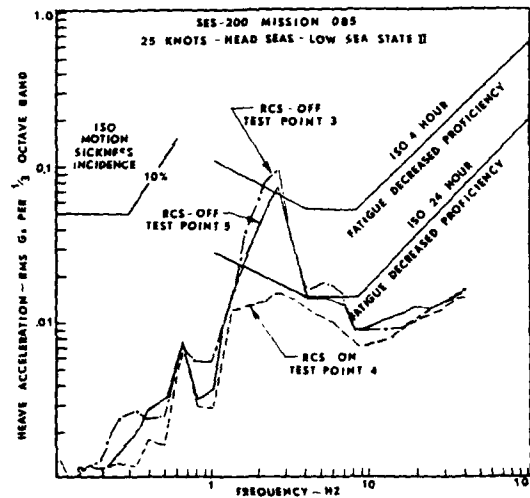


Figure 18. SES-200 Heave Acceleration per 1/3 Octave Band, Low Sea State II

Figure 19 illustrates the system's effectiveness during head sea operation at 21 knots in high Sea State III. These seas represent typical heavy weather conditions for this size vessel as the 1/10 highest waves exceed the distance from the keel line to the cross deck structure.

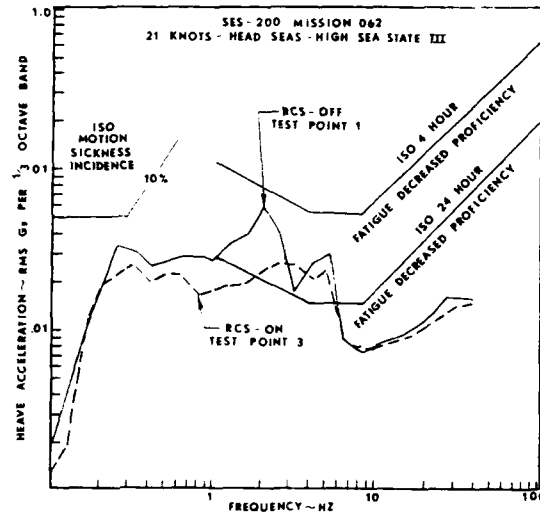


Figure 19. SES-200 Heave Acceleration per 1/3 Octave Band, High Sea State III

In Sea State III, the encounter frequency of maximum wave energy is much lower than the natural heave frequency. There is still appreciable motion at the heave resonance. However, the wave energy is concentrated at lower frequencies where wave components causing large changes in the cushion volume are present.

The RCS reduces the heave accelerations at all frequencies except 3 Hz with the net effect that the overall RMS is reduced by 33% from 0.12 g's to 0.08 g's. The system was not as effective in Sea State III as it was in Sea State II since there was insufficient air flow available to counter the low frequency wave pumping effects.

Since completion of the SES-200 tests reported in Reference 3, two additional lift fans have been installed on the SES-200. During the fall of 1984, tests will be conducted to assess the ride quality improvement and power requirements associated with using this additional fan flow for ride control. Future plans call for retrofitting the SES-200's lift fans with inlet guide vanes as throttling the fan inlet flow is more efficient than dumping pressurized air overboard through vent valves. The vent valves will be retained and used in conjunction with the inlet vanes in the higher sea states.

ACKNOWLEDGEMENT

The authors wish to acknowledge the efforts of Mr. Mark Lindler (formerly of Maritime Dynamics, Inc) who designed and built the electronics for both the XR-1D and the SES-200 Ride Control Systems. The authors are indebted to the civilian and military personnel who worked at the U. S. Navy Surface Effect Ship Project Office (PMS-304) and the Surface Effect Ship Test Facility (SESTF) during the XR-1D and SES-200 test programs. Successful completion of these test programs would not have been possible without the enthusiastic support shown by the following individuals: Mr. Don Herr (Head of Testcraft Maintenance), Mr. Harold Ketcham (Data Systems Engineer), Ms. Barbara Schaub (Head of Data Reduction), and ETI Steven Goss (XR-1D Data System Technician).

REFERENCES

- (1) P. Eykoff, System Identification - Parameter and State Estimation, John Wiley and Sons, New York, NY 1974
- (2) J. C. Nash, Compact Numerical Methods for Computers: Linear Algebra and Function Minimization, John Wiley and Sons, New York, NY 1979
- (3) J. D. Adams and W. F. Beverly III, "Technical Evaluation of the SES-200 High Length-to-Beam Surface Effect Ship", Naval Engineers Journal, Volume 96, Number 3, May 1984

INTEGRATED DAMAGE CONTROL SYSTEM

by David W. Geer
United Technologies
Norden Systems (USA)

ABSTRACT

The practical problems associated with control of battle damage in combatant ships have not been addressed in the design of shipboard control systems, resulting in significant under-utilization of existing damage control resources aboard ship.

Two divergent design approaches have not lead to the most effective implementation:

Automation of propulsion, electrical, and auxiliary control with attendant high performance and reduced manning.

The labor intensive functions of damage control, with heavy reliance on information exchange and situational decision making to prioritize, coordinate, and direct work crew effort and equipment allocation.

Control system designs, which include damage control as a subsystem, provide limited equipment control and sensor monitoring. This is only a part of the capability required to control damage. The most essential ingredient is rapid and accurate organizational direction and control of repair personnel. This can be accomplished via a total ship data communications network, combined with rapid data injection, distribution, and the ability to process, model, and display information directly relating to the damage sustained in a format immediately usable to the operator for decision making.

The basic architecture utilized to illustrate the advantages of an Integrated Damage Control System is that of the Data Transfer Network such as Shipboard Data Multiplex System (SDMS) or Ships Integrated Processing and Display System (SHINPADS) currently included in modern ship designs.

INTRODUCTION

The sources of potential damage to today's ships, military and commercial, are many. The weapons threat includes high-density anti-ship missile attacks, guided projectiles and bombs, torpedos and mines. Their payloads include not only conventional high explosives, but also nuclear, chemical, and biological agents used in combination or separately. Equally threatening sources of potential damage include collision, hazardous cargo, hazardous operational activity, sabotage, terrorism, negligent acts and extremes of weather. The resultant damage can readily exceed a ship's capabilities to detect, contain, and control that damage. The effect can range from reduced mission effectiveness to loss of life or total loss of the ship.

Ships are seldom lost as a result of direct damage effects (primary damage) but rather as a result of the spread of primary damage such as fire, flooding, and equipment disabling, into surrounding areas (secondary damage). The effectiveness of the damage control organization, led by the Damage Control Officer, largely determines ship survival and the ability to restore mission effectiveness.

Many avenues are being addressed to enhance the control of shipboard damage, such as provision of expanded sensor capability to detect smoke, temperature or pressure changes to warn of damage, development of new firefighting chemicals, and the significantly expanded use of electrically operated valves to allow system reconfiguration subsequent to damage. These developments may not keep pace with the many sources of damage, potentially leaving a significant margin between shipboard damage engagement capability and the damage threat.

Ship designs provide the best available capability to engage and control damage. However, the shipboard resources are limited by the constraints applicable to a particular ship class design and the damage engagement equipments available for inclusion in that design. The potentially significant disparity between the threat and capability to respond provides an urgent incentive to maximize the effectiveness of damage control resources to detect, contain and control damage.

This paper presents the damage control system in terms of detection, evaluation and decision, and engagement subsystems, focusing on the evaluation and decision process and the synergism that can result from application of present day command control (C2) designs and equipments to problems of damage control.

Command control integration of the damage control subsystems for centralized decision making would unite them into a damage control system. New construction ships are only beginning to reflect this concept. Optimizing the performance of the damage control system as a whole is not yet being performed systematically. This is considered attributable to the limited view of the "damage control system", which varies from ship class to ship class, and is usually restricted to mean the fire fighting capability of a ship.

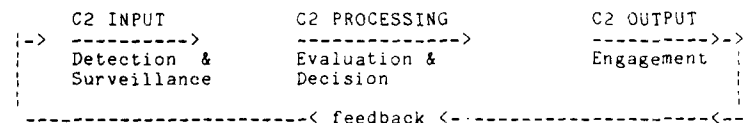
Given that the ship design and outfitting includes the best resources available to detect and engage damage, then it is proposed that further increases in capability will result from command control improvement that will maximize the effectiveness of residual ship resources subsequent to the occurrence of damage. The effective application of resources is predicated on an accurate and timely description of damage from which the development of an "optimal" strategy will evolve to rapidly engage damage within the available resource constraints. The key to successful engagement of damage is an accurate understanding of the damage and the speed with which adequate resources are brought to bear at the proper location(s) to contain the spread of damage and control the source of damage.

THE PROBLEM

The increasing severity of battle damage threats and the associated rapid spread of secondary damage mandates the most rapid containment and control of damage possible. Containment and control must be established rapidly while still within the limited resources available to the ship to minimize the further reduction in ship mission effectiveness.

In general, the principal shipboard decision makers do not have an accurate or timely description of damage. Difficulty is experienced continuously in prioritizing and coordinating efforts at the scenes of damage. Damage may spread out of control for tens of minutes while investigators observe and report damage, damage plots are made, and repair teams are properly equipped and positioned to begin engagement.

The following conceptual approach is used as a method of presentation:



where:

Detection and Surveillance includes:

Damage or malfunction sensors such as temperature, liquid level; equipment status, vibration, load conditions, pressure indicators, and; visual descriptive data from investigators at scenes of damage.

Evaluation and Decision includes:

Receipt, recording and display of detection and surveillance data; situation evaluation, receipt and dissemination of requests and orders, evaluation of equipment malfunction and operator reconfiguration; determination of strategy and response actions.

Engagement includes:

Remote control or directed application of resources at the scene(s) of damage and reconfiguration of residual capability. Control includes operation of electrically operated valves, pumps, high capacity fire fighting systems, and damage control equipments configured for remote operation. Directed application of the resources to be applied at the scene of damage including: firefighting teams, stations, and equipments; stability, buoyancy and dewatering teams and equipments; decontamination teams, stations, and equipments; and coordination of combat system and engineering reconfiguration activities such as emergency power, chill water, and air conditioning.

Deficiencies in current damage control systems may be illustrated by a descriptive scenario:

The typical damage engagement sequence proceeds from detection and description of damage, to evaluation and allocation (or reconfiguration) of resources (repair personnel, systems, and equipments), to engagement of damage, to evaluation and reengagement as required. The damage is engaged with prioritized resources (e.g., 1st fire, 2nd flooding) as are multiple damage locations (e.g., fire near magazines vs. fire in berthing space). Certain types of damage such as fire, are assigned resources regardless of location of occurrence, whereas other types of damage, such as structural debris, may be deferred for engagement until further resources are available.

Damage Detection and Description

The sources of information, or "inputs", on which decision makers act are provided in three broad areas: (1) environmental sensors, (2) equipment status, and (3) human observation.

(1) Sensor inputs such as temperature and pressure are being installed aboard ships in significantly increasing numbers to provide decision makers more detailed status of systems and spaces. However, it is becoming increasingly difficult to monitor and evaluate the meaning of "stand alone" sensor displays and to apply the information rapidly to a total-ship corrective action strategy.

(2) Equipments such as electrical pumps and large numbers of electrically operated valves are also being installed on ships in increasing numbers. The decision makers are being provided an ever increasing capability to remotely monitor status and control essential equipments. Here too, the operator faces data saturation impacting operational decision making subsequent to damage due to difficulty in determining proper response action. Sensor data and equipment status are not "integrated" for display and decision, forcing the decision maker to perform that function.

(3) Human observation of major damage is the principal source of descriptive damage data on which decision makers act. The descriptive data is relayed several times and plotted for decision maker use. The damage plots are separate from sensor

data display and equipment status display, forcing the decision maker to scan several locations and plots to achieve an integrated picture. The damage investigation is performed under conditions of fire, smoke, flooding, and jammed hatches precluding access. It is a highly subjective process based on the experience of the investigator. Damage investigation is also a time consuming process requiring physical coverage of assigned routes, notation of the damage observed, conveying or presenting the observations to organizational superiors, and the relay of information to the required stations. Investigators are generally precluded by the damage itself from gaining direct access to the central source of damage and are thus limited to observations of peripheral symptoms of that damage or the spread of secondary damage. Decision makers must await the fragmentary reports of investigators as they complete their routes and the reports are relayed and analyzed.

The significance of current methods of damage detection and description are that: (1) the sensor and equipment status inputs are saturating the operator with data in the event of major battle damage, (2) the interrelationship between sensor data and equipment status is not clearly presented to operators, (3) proper operator response actions are difficult to determine, (4) investigator reports are "manually" relayed and plotted, (5) the sensor data and investigator reports do not adequately describe the primary damage, and provide a limited and probably incomplete description of secondary damage, (6) the investigator reports are fragmentary in occurrence and highly susceptible to reporting inaccuracies such as to location and symptoms, (7) damage investigation and reporting is a time consuming process relative to the speed with which damage spreads, (8) there is high probability that by the time all sensor and investigator reports are received the actual damage is significantly different than that reported, and (9) the cumulative effect of reporting inaccuracies may preclude the development of an optimal strategy to engage damage.

Evaluation and Decision

The principal decision maker in the evaluation and decision process is the Damage Control Officer. His decisions are subject to the approval of Command, either affirmatively or by negation. Supporting decision makers of the Damage Control organization include, the Repair Officers of each repair party and scenes of damage leaders, as well as the department heads and key battle stations affected by the damage. Each requires an accurate and timely description of the damage, individually tailored to meet their specific need, in order to perform assigned duties related to restoring ship capability.

The baseline for command evaluation and decision is a total-ship descriptive damage "model" developed by the Damage Control Officer. This "model" is developed from the damage detection and description inputs previously discussed and is the basis for determining the engagement actions that will be discussed in the next section.

The current method for developing this "model" involves several separate displays and operator actions. Typically, these

include fire alarm panels that may include some fire equipment controls, equipment status panels or consoles where changes of status may be displayed, summary status boards to depict significant ship capabilities or casualties and the several ship damage control diagrams on which the damage plots are maintained. These facilities vary in availability within the damage control organization. Each station is required to maintain a "plot" appropriate to assigned responsibilities.

The quantity and rate of data coming into key stations can easily exceed the ability to manually record, sort, retrieve, and display for decision. This is equally true of the use of status panels when numerous alarms are activated nearly simultaneously. Data must be recorded in a manner suitable for rapid retrieval, reference, display, and decision making.

"Integration" of the various displays and information must be accomplished by the decision maker. All of the detection and description data (inputs) are not available to all of the key decision makers as they occur. This is further impacted by the "input" delays and potential inaccuracies previously discussed.

The current procedures for recording and display of damage information do not provide real-time support to the decision maker in accomplishing assigned duties. Rather it makes him a slave to the function of recording and monitoring of status, data integration, and relaying information. To enable the key members of the battle organization to perform their principal task, DECISION MAKING, they must first be freed from the morass of details.

Currently, by default, critical decisions tend to be made in real-time, with the information at hand, at the lowest levels of the organization and have the effect of committing the ship to a less than optimal procedural course of action (strategy) to engage damage.

Difficulty in real-time, informed, decision making is also impacted by the lack of comprehensive decision aids that will assist key decision makers in the development of overall strategies to engage damage and the reallocation or reconfiguration of resources subsequent to the occurrence of major damage. A modest decision aid capability is available in the sense of prediction nomograms for nuclear fallout and color coding on liquid load diagrams to portray flooding effect, and in procedural response to alarms and equipment malfunctions. However, major battle damage will affect, in some measure, nearly every shipboard system as well as potentially 1/2 (or more) of the shipboard spaces on a destroyer size ship. The sheer volume of data, interactive effect of actions or decisions in one system or section of the ship with others, compounded by specialized nuclear, biological, and chemical event actions, potentially exceeds the capacity of the human mind to assimilate and determine an optimal overall strategy in a reasonable time, especially under conditions of exceptional stress.

In summary, the evaluation and decision process is difficult at best, because: (1) the damage model may be an inaccurate basis for evaluation and decision, (2) the damage model for each

decision maker may be different because of time delays coupled with reporting and plotting inaccuracies, (3) evaluation and decision may significantly lag the actual events, (4) decision makers can easily be relegated to recorders of events due to their inability to maintain real-time cognizance and control, (5) real-time coordination among the repair parties and other battle stations is nominal at best, (6) no decision aids are available to prioritize the containment and control of damage, the restoration of systems and equipments, or the allocation of remaining resources, (7) no decision aids are available to develop an "optimal" strategy to engage total damage, and (8) no decision aids are available to assist in prioritizing the reallocation of resources to engage subsequent hits.

Engagement

Damage control involves the actual containment and control of damage by both men and equipment. This includes the actions required for nuclear, chemical, and biological attack, as well as high explosives. The specific response actions required for these events, either individually or in combination, are complex and will not be addressed as part of this paper. The focus of this section of the paper is on the "output" of the evaluation and decision process previously described. This output consists of (1) equipment control actions for remotely controlled equipment, and (2) organizational coordination and direction of not only the repair organization, but coordination with and decision making by command and the various ship departments and key battle stations.

Equipment control actions, such as firemain reconfiguration, by means of electrically operated valves and pumps, are a total system decision that have a logical and unique solution based on (1) the damage sustained and (2) the functional demands on the system. The sensors and equipment status reports can generally provide the information required for control actions if the results are recorded so that the decision maker may comprehend the damage and the decision maker has the time to carefully think through the response actions. However, the occurrence of major damage will typically result in a solid "red" status board which provides no insight into response action required. This lack of meaningful information coupled with the stress of "doing something", is not conducive to carefully thought out responses.

Organizational coordination and direction is presently a voice process with several relays between the originator and the destination of the information. Of necessity, this leads to brevity, lack of amplifying information, and the significant probability of error in transmission. Further, the effect of severe stress in voice communications with men, working in a life threatening environment, under severe stress, poor visibility and communications, using equipment and systems whose effectiveness at the time of application is not "known", against damage whose precise nature and extent is unknown, requires the utmost clarity, timeliness, and conciseness of information specifically tailored for the user.

The coordinated efforts of the total repair organization are seldom brought to bear at a scene of damage. Reallocation of repair resources to engage spreading damage or new damage

involves the redeployment of men dressed in protective clothing and masks and cumbersome equipment with reduced communications capability and reduced mobility once committed. Reallocation of committed resources is time consuming relative to the spread of damage, and places a premium on the correct initial allocation and placement.

Engagement deficiencies may be summarized as: (1) the precise nature and extent of damage is not known, (2) damage is likely to be engaged with inaccurate and incomplete information, (3) specific decisions at the scene tend to be made with inaccurate or late information, (4) the engagement of damage may be largely a trial and error process based on the best judgement of decision makers at the scene, (5) the effectiveness of the equipment, techniques, and tactics used to engage damage is uncertain with respect to effect and outcome, and (6) coordination and information exchange among the several groups engaging the same damage at separate locations is not maintained.

Problem Summary

The cumulative effect of the detection, evaluation and engagement deficiencies as currently implemented is that: (1) initial damage effects may be allowed substantial time to spread uncontrolled, thereby significantly increasing the resources eventually required to contain and control the damage, (2) resources may be allocated inefficiently, resulting in the spread of damage and potentially in loss of life or loss of the ship.

PROPOSED DAMAGE CONTROL SYSTEM IMPROVEMENTS

The implementation of a Damage Control System is proposed as a solution to the problems previously discussed. The system discussed here is essentially a distributed microprocessor control system utilizing the installed ship's data bus (Data Transfer Network), with appropriate consoles to:

(1) Provide accurate, real time knowledge of the damage status of the total ship and the capability to immediately apply the combined and total damage control resources of the ship to the prioritized containment, control, and restoration of damage.

(2) Maximize the effectiveness of residual damage control resources subsequent to damage by providing the facilities to develop and implement an informed and reasoned strategy on a "hit-by-hit" basis that: (1) maximizes the probability of ship survival, (2) maximizes the expected gain in the prioritized control of damage and the restoration of mission essential systems and equipments in minimum time, and (3) minimizes secondary damage.

Specifically, the goal of the damage control system is to enable the Damage Control Officer to reduce Damage control Reaction Time which is defined as:

The elapsed time from the known failure of the combat system to preclude successful attack resulting in the occurrence of damage, including chemical, biological, and radiological attack, until damage is engaged with sufficient resources to

contain the effects of primary damage and control the spread of secondary damage.

Reducing reaction time will result in reduced total ship damage as portrayed in figure 1:

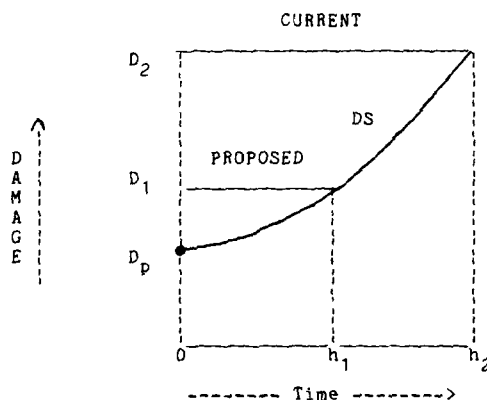


FIGURE 1: DAMAGE AS A FUNCTION OF TIME

where: D_p is the primary damage or direct effects, caused by the source of damage

D_s is the time dependent spread of damage, or secondary damage plus primary damage

h_1 is the time required to contain and control a given level of initial damage, D_p , and its secondary damage, using the proposed damage control system

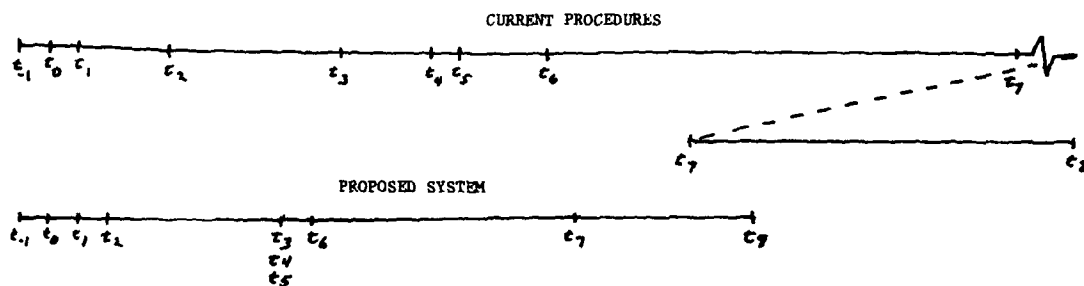
D_1 is the resultant total ship damage at h_1

h_2 is the time required using current procedures, to contain and control a given level of initial damage, D_p and its secondary damage

D_2 is the resultant total ship damage at h_2

Figure 2 illustrates selected elements of the damage control reaction time and the potential reductions identified with the proposed system.

Reaction time begins with the known (or high probability) failure of the combat system to preclude damage. This is identified as t_{-1} . The time between t_{-1} and the occurrence of primary damage, t_0 , is currently not utilized by the damage control organization.



- 4186
- t_{-1} Damage Imminent
 - t_0 Damage Occurs
 - t_1 Sensor and Equipment Status
 - t_2 System Reconfiguration
 - t_3 Investigators Complete Routes
 - t_4 Investigator Damage Reports
 - t_5 Damage Model Completed
 - t_6 Orders to Repair Organisation
 - t_7 Secondary Damage Controlled
 - t_8 Primary Damage Contained

FIGURE 2: ELEMENTS OF DAMAGE CONTROL REACTION TIME

Inbound threats are generally identified with relatively high confidence as to weapon type, payload, trajectory speed, terminal attack, angle, and bearing. This information and the time between t_{-1} and t_0 could be used by the damage control organization to take significant survival action if it were made available. For example, the electrical load could be split, load shed initiated, ventilation systems in the expected area of impact isolated, the fire main further segregated, and repair personnel relocated away from the expected area of impact. These pre-damage actions could significantly enhance overall ship survivability and mission effectiveness.

Equipment malfunction indications and sensor data inputs are considered to be available to the Damage Control Officer at t_1 . Equipment and system remote reconfiguration is considered accomplished at t_2 based on data available at t_1 . The proposed system would input all sensor and equipment status into the Data Transfer Network allowing digital data processing techniques to provide recommended actions to decision makers enabling more rapid and informed response actions.

The damage investigators are considered to have completed their routes at t_3 . Their complete reports are considered available to decision makers at t_4 . The proposed system would provide portable data entry devices (to be described later) for investigators to enter descriptive damage data as it is observed into the Data Transfer Network for immediate display at key stations. This would eliminate the need for note taking, numerous voice relays, and plotting of data. All stations would be using the same data in real time.

The damage model is considered to be completed and displayed at key damage control stations at t_5 . The proposed system would record and display damage as rapidly as it was entered into the Data Transfer Network, whether done automatically by installed sensors, manually via Data Entry Devices, or a combination of both.

The Damage Control Officer is considered to have evaluated the overall damage and determined the courses of action at t_6 . The proposed system would display computer generated recommended courses of action as a decision aid.

Secondary damage is considered to be under control at t_7 and primary damage contained at t_8 . The proposed system would contain and control damage more rapidly in part due to a faster and more accurate description of damage, in part due to system reconfiguration prior to the occurrence of damage, and finally due to faster initial engagement, and more efficient direction of repair resources to contain and control damage.

A further reduction in reaction time could be obtained by a significant expansion of shipboard sensors to describe and display damage more rapidly than the investigators can provide it. This would have the effect of substantially shortening the interval $t_2 - t_3$ as well as the intervals $t_6 - t_7$, and $t_7 - t_8$. However, it was considered that the highest priority for performance improvement was in information processing. With the requisite processing capability, the further employment of

sensors could then be effectively utilized.

Proposed improvements to the Damage Control System are initially concentrated in the areas of; detection and surveillance, and evaluation and decision. These include the following:

Detection and Surveillance

(1) Input combat system threat data directly to the Damage Control Officer when the probability of ship damage passes a currently unspecified predetermined threshold.

(2) Digitize all electrical damage control sensor and equipment status data, and input it to the ship's Data Transfer Network.

(3) Develop and provide a hand held Data Entry Device (DED) to be used by investigators and personnel in charge of scenes of damage to input damage status and repair progress as and where occurring. The DED would be ruggedized, battery powered, and capable of transmitting the same type of information currently prescribed for standardized damage control message formats and use standard damage control symbology. The DED would be used to make both internal and external surveys of the ship. Preferably, the DED would employ low power radio frequency transmission, similar to Man On Move Communications (MOMCOM) to an embedded antenna system allowing the user full freedom of movement. Alternatively, and less effectively, the DED could utilize a plug-in jack system. The DED would "input" data via the ship's Data Transfer Network.

Evaluation and Decision

Recording and Display of Information. Digitized sensor data and equipment status are easily amenable to computerized recording. The use of a DED would also make observer data suitable for computerized recording, thus providing a coherent, total ship data base.

With all input data suitably recorded, retrieval for display and decision will be possible. The decision makers can be presented an integrated "picture" of the damage, together with its rate of spread, and (as will be discussed later) a corrective action strategy, together with expected outcome and risks associated with the selected strategy.

Generation of Status Information. The generation of status information in a format suitable for each decision maker is easily accomplished in real time with computerized information handling. Complex matters requiring both vertical and horizontal organizational consideration can be rapidly presented together with relevant decision parameters. Alternatives can be rapidly explored by use of "what if" routines and the outcomes evaluated. Status information can be specifically tailored to support each decision maker at his console and amplifying data called up as required.

Generation of Output Data. Output consists of two broad yet interrelated categories of information; (1) equipment control orders such as valve or pump operations, and (2) information to other stations such as requests, orders, and status. Neither category can be accomplished in isolation. For example: the reconfiguration of the firemain subsequent to damage involves not only knowledge of sequence of low pressure alarm activation; but pump status, current demands, imminence of emergency demands such as flooding magazines and activation of water-washdown systems, and the effect of reconfiguration such as bypassing AFFF units or loss of equipment cooling water. Information directed to other stations will include the strategy for containing damage, specific coordinative instructions to all repair parties, and locations to initiate action.

The computer generated output data would address all major considerations and provide recommended operator action for acceptance or modification as appropriate.

Expected Damage Model. The use of a computer generated Expected Damage Model will provide the real time decision aids necessary to support key decision makers.

The Expected Damage Model consists of three parts: (1) threat damage model, (2) damage control effectiveness model, and (3) strategy and outcome model.

Generation of an Expected Damage Model based on the specific threat, equipment capability, and expected outcome, will materially assist the shipboard decision makers in two highly beneficial ways: (1) training, and (2) the actual control of damage. Decision training under stress with simulated major damage aboard the actual ship is essential for the effective control of damage in war. The major payoff of the Expected Damage Model would be as an aid to key decision makers by integrating predicted damage with reported damage, realistic assessment of shipboard capability, and the ability to evaluate alternative strategies. As observer reports further amplify damage, the expected damage model will be replaced with actual data or can be cancelled at any time.

The Expected Damage Model does not present a significant problem to develop, particularly for weapons, since the threat is largely "known" including; general time frame of susceptibility, method of delivery, weapons, payload, trajectories, and predicted effects. The basic techniques for combating various forms of damage are known, as are the techniques for estimating the outcomes of various strategies.

The "unknown" elements that are not fully defined until the occurrence of weapon detonation are; the specific weapon detonating at the target ship, location of detonation, time of detonation, and residual ship capability. However, threat information is available from seconds to minutes prior to detonation at various combat system stations with a confidence level inversely proportional to the time until weapon detonation. Making the threat information available to the Damage Control Officer allows prediction of ship impact, advance planning and preparation in the form of an initial defensive response.

With the occurrence of actual damage, all threat elements are "known". The damage control organization can immediately model the expected damage and implement a strategy to combat the damage, substantially speeding the initial allocation of resources to the scene(s).

Architectural Concept. The advent of Data Transfer Networks (DTN), such as SDMS and SHINPADS, provides the shipboard capability to integrate all damage control related data in a format suitable for computer information processing. The DTN also provides for area concentrators to receive on scene observer data, and remote sensor data and equipment status reports for relay to computerized consoles at key control stations. Thus, the DTN provides the basic architecture to accommodate automated damage control information handling requirements. The DTN can also be utilized to disseminate information to the Repair Officers in a concise format.

CONCLUSION

This paper has presented a means to achieve significant improvement in the detection, evaluation, and control of damage by incorporating modern command control capabilities into new ship construction programs.

The proof will only be found in the doing and testing. Analyses must be performed that will lead to a test bed and progress to an engineering development model at an appropriate facility. This must be followed by shipboard testing and evaluation, and finally, by incorporation in new ship construction programs.