ADVANCED SURFACE SHIP DESIGN SYNTHESIS UTILIZING AUTOMATED OPTIMIZATION TECHNIQUES AND INTERACTIVE COMPUTER GRAPHICS

D. E. Calkins, D.Eng.,P.E.
Research Associate Professor

M. S. Pantazopoulos
Research Assistant

TABLE OF CONTENTS

VOLUME 1

PAGE

1.0 INTRODUCTION

During the past 20 **years,** major advances in engineering analysis and computational technology have occurred. The engineer has delegated the task of computation and analysis to the computer at every stage of the design process and become involved only with the decision-making process or with the investigation of the optimum model for each particular case.

In general terms, the design process of a system today is composed of the following steps:

(1) **Initialization Stage.** At this point, previous models are considered, and quick crude calculations are performed to obtain a baseline model which will be used as the starting point in the design process. Comparison studies of several different models are performed to satisfy the general design criteria to lead to the best model for use at the preliminary stage. This particular part of the initialization stage is called synthesis, and is the procedure for converting a set of requirements into a physical description of the model which satisfies the necessary requirements.

(2) **Preliminary Stage.** At this stage, the basic model is examined through more vigorous analytical techniques. The assumptions of the initialization stage are confirmed at a more detailed level in all calculations. More specific analytical and/or graphical proof that the proposed model's characteristics will satisfy the more important functional and environmental

requirements is sought through applied calculation procedures, with a higher degree of accuracy.

At this stage, modification of the current model **may** occur, but only in such a way that the overall design of the model remains approximately the same.

(3) Final Stage. This is a visual, analytical and descriptive presentation of the model to be built and operated, with credible proof that the proposed model will satisfy functional and environmental requirements in an optimal manner. During this stage there is no modification of the current model.

2.0. SHIP DESIGN AND THE COMPUTER

The design process of a ship is necessarily one of iteration. Design criteria are established, and then changes in various components of the ship are analyzed to determine how they affect the total system performance. The design criteria are then modified and reanalyzed until an optimum design has evolved. This design process. on the surface. seems relatively straightforward until the enormous complexity of present day ships is considered. If the designer tries to consider all variables in their most complex interrelationships. extending the design cycle beyond reasonable limits is risked. On the other hand. making simplifying assumptions in the analysis quite possibly compromises the design.

2.1. Computer-Aided Design and Engineering (CAD/CAE)

The computer is a tool which has revolutionized engineering design. In computer-aided design, the engineer is able to interact with the computer by making qualitative judgments based on externally displayed quantitative information. Here the governing philosophy is not only to keep the judgments in the designers' hands, but to make it easier to get the information they need to make those judgments.

If designers can formalize parts of the design process so they can be entered into the computer. where equivalent data representations can be manipulated rapidly and precisely. then they are free to concentrate on parts of the design activity which cannot be formally treated by mathematical analogy.

The design process is a mixture of imagination, know-how. design rules learned from formal education and experience. calculations. and repetitive modifications. Much of the design process consists of establishing procedures which solve part of the design problem using information available from the designer's knowledge, handbooks. and model and full scale test data. Much of this information can be stored and specified in computer programs.

The emergence and evolution of Computer-Aided Design and Engineering as an engineering technology is well documented in the literature and evidenced by the explosive growth of this industry. The CAD system market, $2 billion in 1983, is projected to exceed $9 billion by 1987. CAD sales have been increasing about 30 percent a year and are expected to increase to 40 percent over the next five years, so that by the end of the decade about one out of five engineers, designers ,and draftsmen will be using CAD/CAE systems.

Computer-Aided Design (CAD), which is the process of geometric modeling, includes the conception and synthesis of a system such as a ship, using the computer coupled with an interactive graphics capability to display and view the design. Three-dimensional wire frame models are the typical display format. The designer describes the shape of a structure with a geometric model constructed graphically on the CRT screen of the CAD system. The computer then converts this pictorial representation into a mathematical model which is stored in the computer data base for later use. The model may be used for other CAD functions, or it may be recalled and refined by the engineer at any point in the design process.

Computer-Aided Engineering (CAE) is the engineering analysis of the design concept or geometric model- created using CAD. With simple keyboard commands, the user may have the computer calculate, for example, the ship's weight, volume, surface area, moment of inertia, or center or gravity. Other analyses might include stresses and deflections, surface pressures and velocities, and system time or frequency domain dynamics.

The tool that ties these computer-aided functions together is Interactive Computer Graphics. The CRT display gives the engineer/designer speed and accuracy by mathematical computer description, and visualization from many viewpoints. It increases productivity by tying together analysis and design in a fast-responding closed loop design process. The engineer/designer converses with the computer via the keyboard, a light pen and a tutorial menu, in either an alphanumeric or pictorial mode. It is the man/machine link via interactive computer graphics that is important for the successful utilization of CAD/CAE capabilities.

CAD/CAE is especially useful for vehicle synthesis, such as ships or aircraft. As early as 1972, CAD/CAE with computer graphics was being utilized for ship design [1] at the Naval Ship Engineering Center and the Naval Ship Research and Development Center. Computer-aided ship hull surface definition couples the power of the computer with sophisticated mathematical techniques. Rogers [2] describes a Computer-Aided Ship Design (CASD) and Computer-Aided Manufacturing (CAM) program implemented on a PDP 11/45 mini-computer which is used for the design and manufacture of ship towing tank models. The General Aviation Synthesis Program (GASP), developed at NASA Ames [3], is written in FORTRAN and is implemented on an IBM 370/168 mainframe computer using TEKTRONIX 4010 terminals. The use of a mini-computer based aircraft Configuration Development System (CDS), which is an interactive graphics aircraft design, analysis and loft program, is described in [4]. CDS is written in FORTRAN and is implemented on a Sperry-Univac V-76 mini-computer through TEKTRONIX 4044 terminals.

2.2. Automated Optimum Design

A logical outgrowth of computer-aided design is automated optimum design, Calkins [5]. Here, as much of the design process as possible is moved into the computer to reduce the man-machine interface to a minimum. In other words, if the design concept, design limitations, and optimization goals are clearly established and can be stated in quantitative form, it is then possible to include the design process in the form of a mathematical programming problem.. It thus becomes possible to program the innermost cycle of the design process completely for solution by the computer. Thus the quantifiable part of design. the evaluation and optimizing functions, which have been the domain of the design engineer. can now be done by the computer.

The design process may be synthesized on the basis of a series of steps, as diagrammed in Figure 2.1. These steps relate judgments to be made. some of which are quantifiable. and others of which are qualitative. Design begins when a need is identified for a system to do something. The criteria to be used to evaluate proposed designs are then established. Then a design concept is generated. At this stage the designer draws heavily on ingenuity. creativity and past experience. Once a design concept has been originated. it is idealized by developing a model simulation to be used in predicting its behavior.

The predicted behavior can now be compared with the criteria in order to evaluate the design, that is to find the optimum design- If the first model does not satisfy the acceptance criteria, a new design concept must be sought. Finally, if none of the concepts yield a suitable design, the designer must then consider modifying the acceptance criteria.

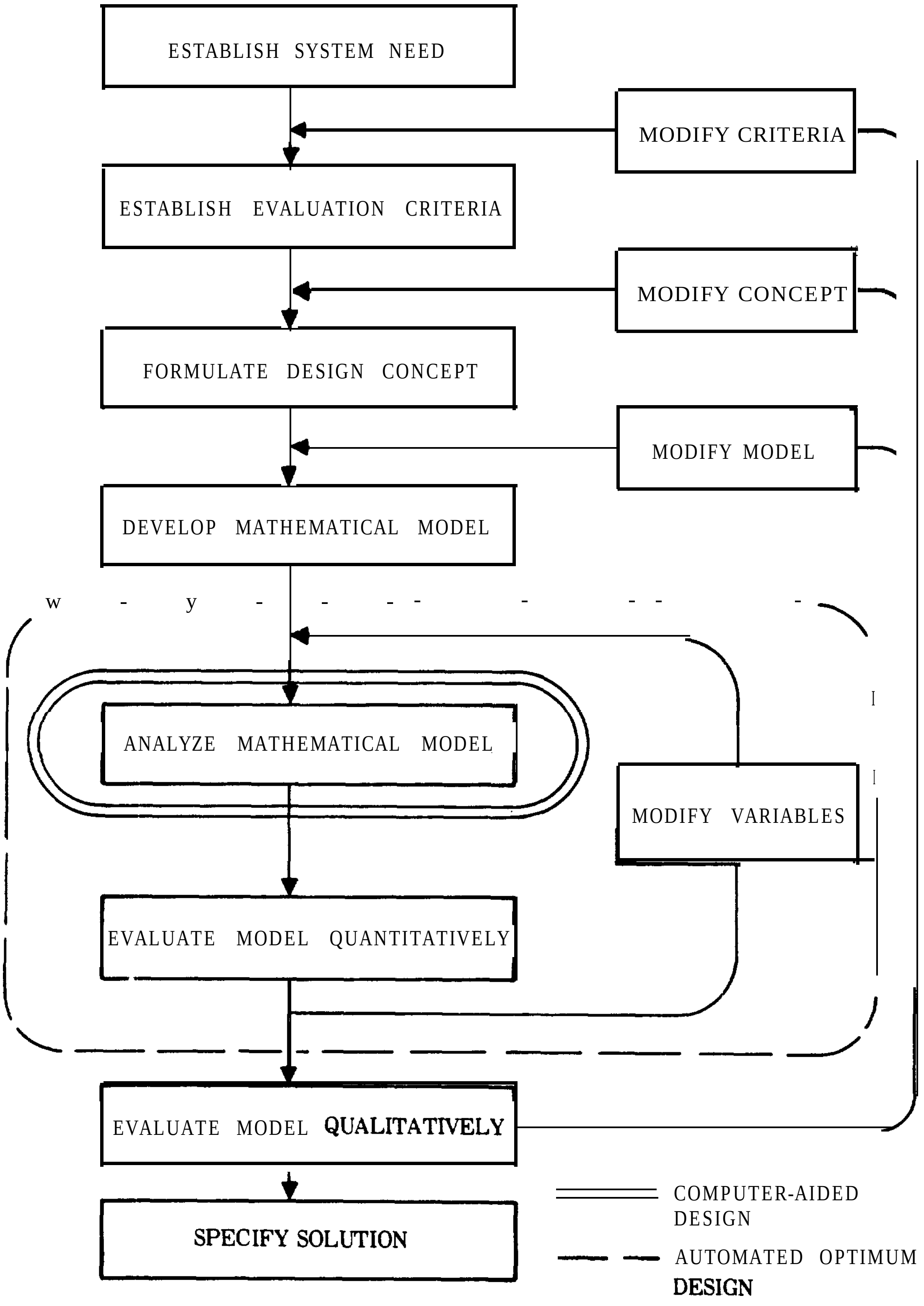


Fig. 2.1. THE DESIGN PROCESS

Programming the design process requires:

1. a list of behavior constraints. which **may** be regarded as limits to be imposed on the design
2. a valid analytical method. which requires simply the definition of the mathematical equations and techniques to be used
3. an objective function. which is some defined figure of merit used to choose among alternate designs.

This process of idealizing the system for analysis requires the judgment and experience of the engineer in making decisions, and thus represents that portion of the design process for which the computer cannot be programmed. The computer is used only for logical decision-making, leaving judgment to the designer.

The design process begins by describing the system by a set of quantities:

1. preassigned parameters--geometrical quantities fixed at the outset
2. design variables--geometrical quantities to be varied
3. design variable space--an n-dimensional Cartesian space in which there is a coordinate axis for each design variable.

A hypothetical two-dimensional design space is shown in Figure 2.2. The behavior constraint shown may be diagrammatically described as a surface in the design space that represents all designs on the verge of being rejected.

Another feature is a side constraint. A design on this surface verges on rejection for some external cause not explicitly related to the behavior restrictions. Side constraints are usually limits on the range and independence of the design **variables.** All of these

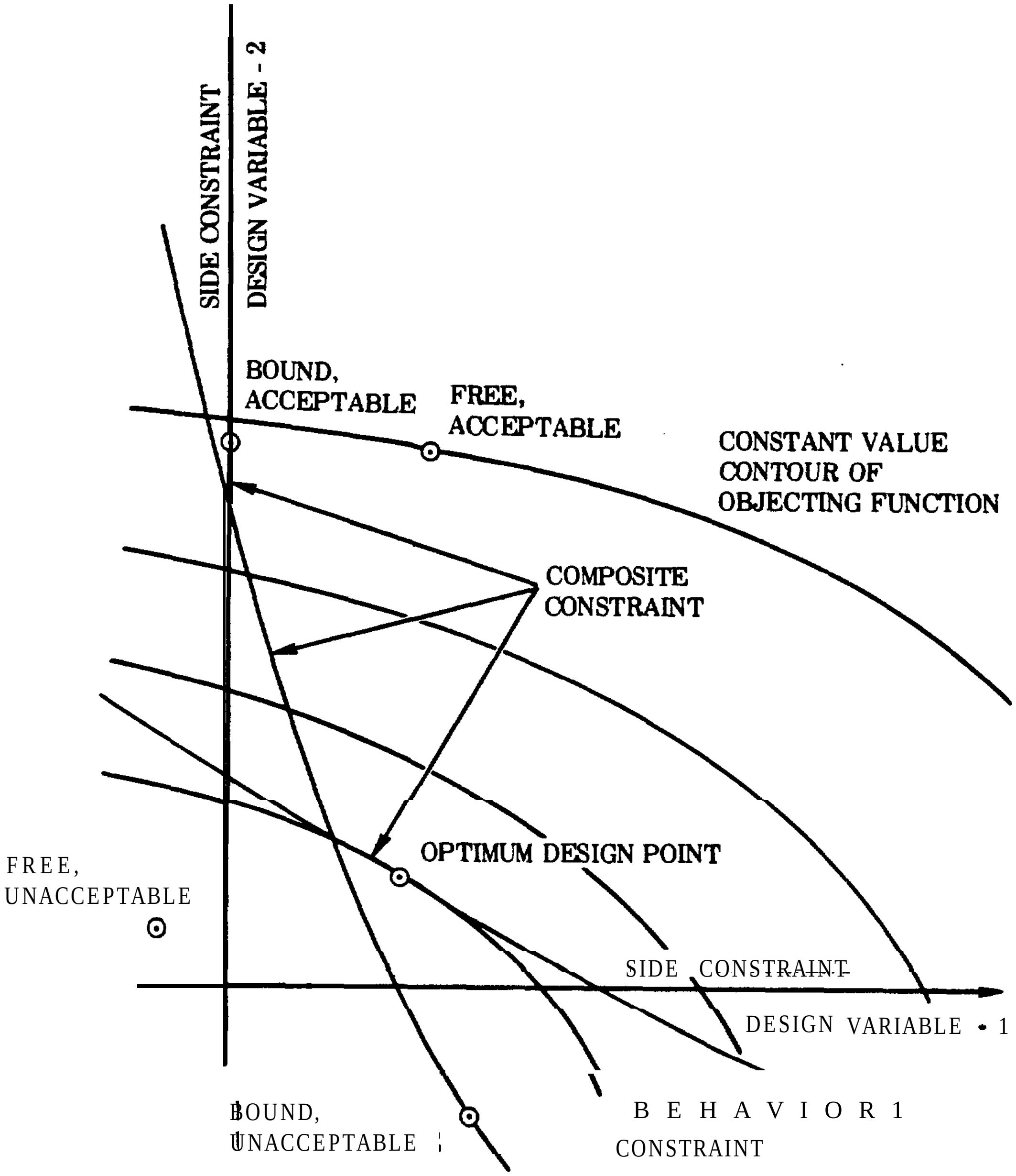


Fig. 2.2. TWO-DIMENSIONAL DESIGN SPACE

constraint surfaces collected together give us a composite constraint surface which separates the acceptance region of the design space from the **unacceptable.** A design that lies on such a surface is a bound point. and one that does not is a free point.

Therefore, any design may be identified as:

1. free and acceptable
2. bound and acceptable
3. free and unacceptable
4. bound and unacceptable.

The designer thus starts at an initial design point and analyzes. evaluates, and modifies successive designs until the optimum design point is reached. The purpose is to choose a path which meets the objective function without violating any of the imposed constraints.

2.3. Ship Synthesis Models

With an existing ship synthesis code- an extensive study of the proposed ship design may be performed. The comparison of the different outputs of these studies enables the engineer to decide which characteristics of the ship design satisfy the needed requirements.

In ship design, after specification of the mission requirements, the designer generates an initial design as a starting point. Then an inter-loop spiral procedure begins, with minor or major modifications and re-evaluation of the ship characteristics through different modules until a converged design is approached. or termination occurs if the intended requirements have not been met. This process, called design synthesis, modifies the current model through different modules of analysis. Critical ship data for the current model. such as hull lines, superstructure characteristics. foil system geometry and characteristics, fuel and range data, etc., are modified during this process.

Ship synthesis models 16, 7,and 8] started appearing in the late 1960's with the U.S. Navy's destroyer model DD07 and the Center for Naval Analysis Conceptual Design of Ships Model (CODESHIP). The design spiral approach forms the basis of the Hydrofoil Analysis and Design (HANDE) computer program, King and Devine [9]. HANDE combines the power of CAD/CAE with the logic of the design spiral. Three types of computational programs exist within HANDE: INITIALIZATION, SYNTHESIS and ANALYSIS. The INITIALIZATION module consists of a single program which utilizes simple empirical methods to provide an initial starting point for a new design under development with HANDE. Ten SYNTHESIS-type computational programs exist within HANDE. Each program is concerned with a single technological area of a hydrofoil ship design. In contrast to the INITIALIZATION program, each SYNTHESIS program utilizes rigorous analytical techniques in computation of ship data. The third type of computational program is called ANALYSIS, of which there are five. The principal difference between SYNTHESIS programs and the Analysis programs is that SYNTHESIS programs modify the current model, while ANALYSIS programs only provide additional information about it.

A similar computer program for planing hulls has been developed by the David W. Taylor Naval Ship Research and Development Center by Hubble 110,111. The planing hull feasiblity model PHFMOPT comprises nine technological areas, including Hull Geometry, Structures, Resistance, Thrust, Propulsion, Other Systems, Loads, Optimization, and Final Hull. A second program, PHPRLM, predicts the resistance, thrust requirements and vertical accelerations of a planing hull over an operational matrix of speeds and wave heights. A total of seven technological areas comprises **PHPRLM,** including Thrust Requirements, Propeller Characteristics, Power Requirements, Engine Torque-RPM Limits, Maximum Speed, Habitability Limits, and Propeller Selection.

ASSET (Advanced Surface Ship Evaluation Tool), 1982, developed by the Boeing Company [12], is the most integrated, versatile and easily used synthesis model.

The ultimate result of using a synthesis model is the ability to produce a far more detailed and accurate design earlier in the design sequence, thereby saving time and money and providing more reliable guidance in the design selection process.

The computer software, even though very powerful, provides only an analysis of a proposed design, with the engineer making the actual decisions. In this aspect lies the disadvantage of this approach. When the designer wishes to conduct a parametric study to evaluate a variety of designs, i.e., looking for an optimum configuration, hundreds of designs must be generated. This requires additional personnel for the tasks of running the programs and making hand plots to determine the influence of the different parameters on the model's particulars. At this point, the necessity of a more efficient means of configuration tracking evaluation becomes apparent.

2.4 Ship Synthesis and Optimization

The main objective of this study is to provide the designer with a tool which can accelerate the conceptual design stage and still produce high quality designs. This will be accomplished using automated optimization techniques. Instead of using a synthesis model to generate hundreds of designs and then manually selecting one which appears to be the "best," or optimum, the computer will be used to make decisions based on the limitations Or constraints, and design requirements coded in mathematical terms. While the task of making decisions is left to the computer, the designer is involved with the judgment and checking of these decisions, based on experience.

While the automated design process has been described and analyzed in several references, little work has been reported dealing with the basic question of how this automation is accomplished most effectively. Several automated synthesis models have been developed since the 1960's. In "Least Cost Ship Characteristics" by Murphy, Sabat and Taylor [13], the coefficients affecting the size and cost of a ship were varied over a range of finite step sizes. Based on that model, Mandel and Leopold [14] introduced a random-search technique with the objective being the minimization of a function which combined cost, required payload weight and required payload volume. In 1975, the Canadian Concept Exploration Model (CEM) [15] utilized a new approach to ship design with an evaluation of a number of ships in the form of a matrix exploration. The unacceptable designs were then eliminated through different imposed criteria. Recently, a new automated model, REED/COPES, Jenkins [16], which interfaces a ship synthesis model REED, with COPES (Control Program for Engineering Synthesis) by Vanderplaats [17] was described. This is a versatile model with freedom in choosing design variables, objective functions or required constraints.

The objective of the present study is to interface the ship synthesis model program ASSET [12] with the automated optimization program COPES [17], and in addition to add graphics capability in order to enhance the information output and perception. In other words, the advantages of an optimization study, i.e. the possibility of finding an optimal solution quickly, will be combined with the advantages of a ship synthesis parametric study. Some of the resulting advantages of the new system are:

(a) further reduction of the computational time necessary to do a parametric study

(b) graphical visualization of the design space and perception of the optimimum design, as well as information in the neighborhood of the optimum

(c) ability to conduct studies on designs which are optimized with respect to different objectives while the same design standards are maintained.

3.0. ADVANCED SURFACE SHIP EVALUATION TOOL (ASSET)

The Advanced Surface Ship Evaluation Tool (ASSET) [12] is an interactive computer program for the conceptual design and evaluation of surface combatants, including frigates, destroyers, and cruisers. The program provides an integrated ship design perspective as well as a capability to study and analyze individual ship functional components. The program includes a data management function which enables utilization of a repository of prior ship designs or their functional components.

3.1 ASSET Program Structure

Three types of computational programs exist within ASSET: INITIALIZATION, SYNTHESIS and ANALYSIS. The breakdown of programs within each type is shown in Figure 3.1.

The INITIALIZATION section of ASSET consists of a single program. It utilizes simple parametric methods to calculate a variety of ship data. As its name implies, a primary function of the INITIALIZATION program is to provide an initial starting point for a new design under development within ASSET. A secondary use of the INITIALIZATION program is in performance of high-level parametric trade studies.

Seven Synthesis-type computational programs exist within ASSET. These include:

1. hull geometry formulation
2. structural sizing
3. resistance evaluation
4. propeller sizing and location
5. machinery sizing
6. weight estimation.

Each program is concerned with a single technological area of the

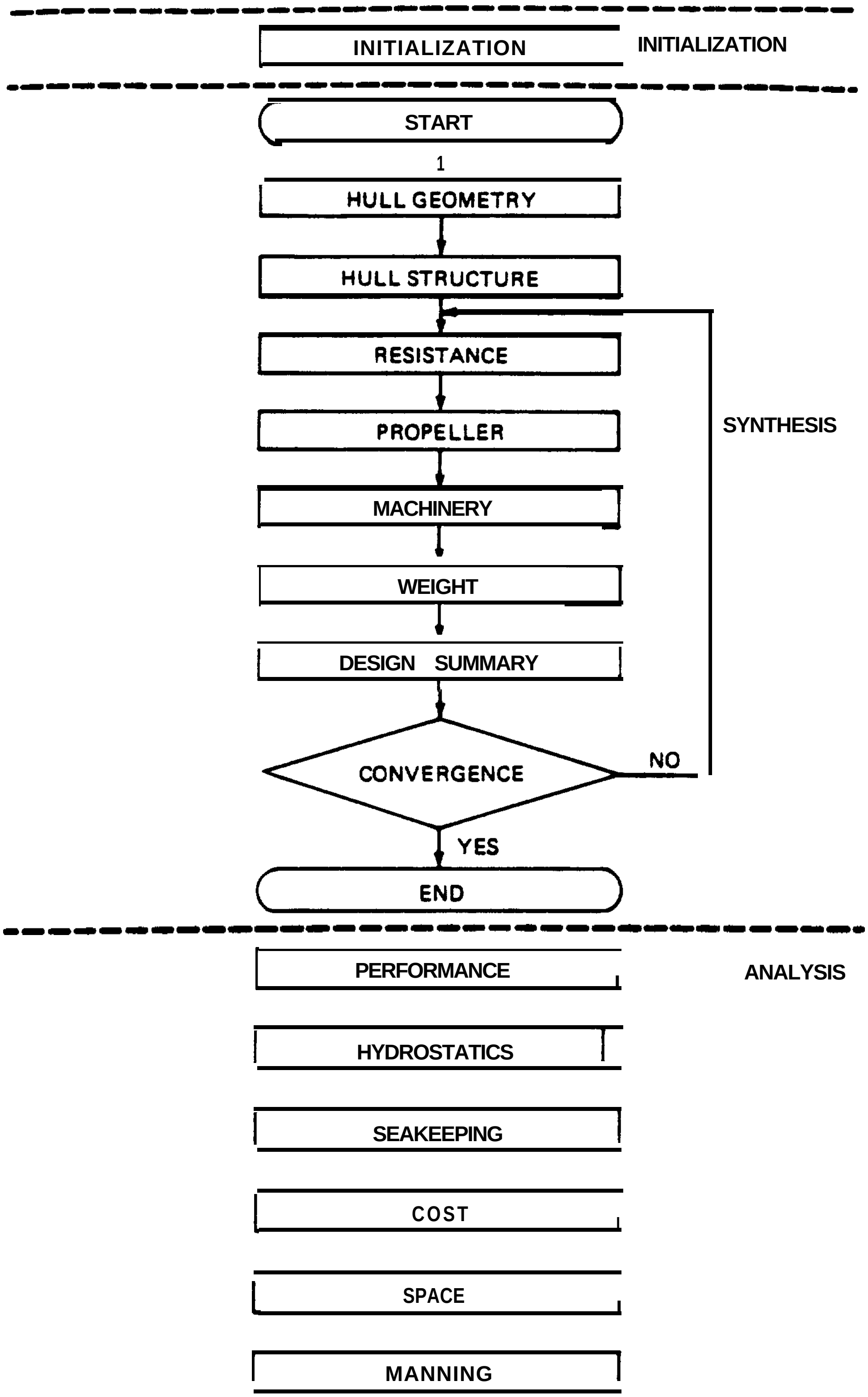


Fig. 3.1. ASSET Computational Modules

ship design, In contrast to the INITIALIZATION program, each Synthesis program utilizes rigorous analytical techniques in computation of ship data.

The third type of computational program is called ANALYSIS, of which there are six, including:

1. performance
2. hydrostatics
3. seakeeping
4. space
5. manning
6. cost.

Depending on the module, either parametric or rigorous analytical techniques are employed. The principal difference between SYNTHESIS programs and ANALYSIS programs is that SYNTHESIS programs modify the current model. ANALYSIS programs do not modify the current model, but provide additional information about it. Also, unlike ANALYSIS programs, SYNTHESIS programs can be employed in an iterative loop to generate a ship design.

Because of the complexity of ASSET, only three of the computational modules which exist within the program have been selected to be used in the present study. These include: INITIALIZATION, COST and SPACE. The nomenclature for each module is contained in Appendix A.

3.2. INITIALIZATION Module

The INITIALIZATION module has two primary functions. Its first function is to provide a capability to synthesize and analyze ship designs on a gross level. Its second function is to provide a starting point for a ship design that is to be further developed or refined by the synthesis computational section. To perform its primary functions, the INITIALIZATION module has been designed to mirror the structure of both the SYNTHESIS and ANALYSIS computational sections. Many of the same engineering technologies present in the SYNTHESIS or ANALYSIS sections are consequently also present in the INITIALIZATION module. But whereas the SYNTHESIS and ANALYSIS sections use relatively rigorous computational techniques and procedures to derive design data, the INITIALIZATION module utilizes simple parametric and empirical techniques to calculate its design data.

Within the INITIALIZATION module are included Mini-Synthesis and Mini-Analysis sections. The Mini-Synthesis section identifies and sizes ship components for the purpose of defining the overall ship. The Mini-Analysis section provides the designer with additional data regarding the ship as defined. The Mini-Synthesis section of the INITIALIZATION module operates in the same iterative sequence as does the ASSET synthesis section.

The Mini-Synthesis process, as shown in Figure 3.2,involves six sub-modules: hull geometry, hull structure, resistance, propeller, machinery and weight. The sub-modules are automatically executed via the interactive loop shown in Figure 3.2. The iterations are terminated when two passes through the iterative loop produce essentially the same design, or when an iteration limit has been reached. A listing of the each function of each Mini-Synthesis submodule is given in Table 3.1.

Following the Mini-Synthesis process, the Mini-Analysis section of INITIALIZATION is executed, Figure 3.2. Only two Mini-Analysis submodules exist: hydrostatics and **seakeeping.** These submodules provide additional information about the design derived from the

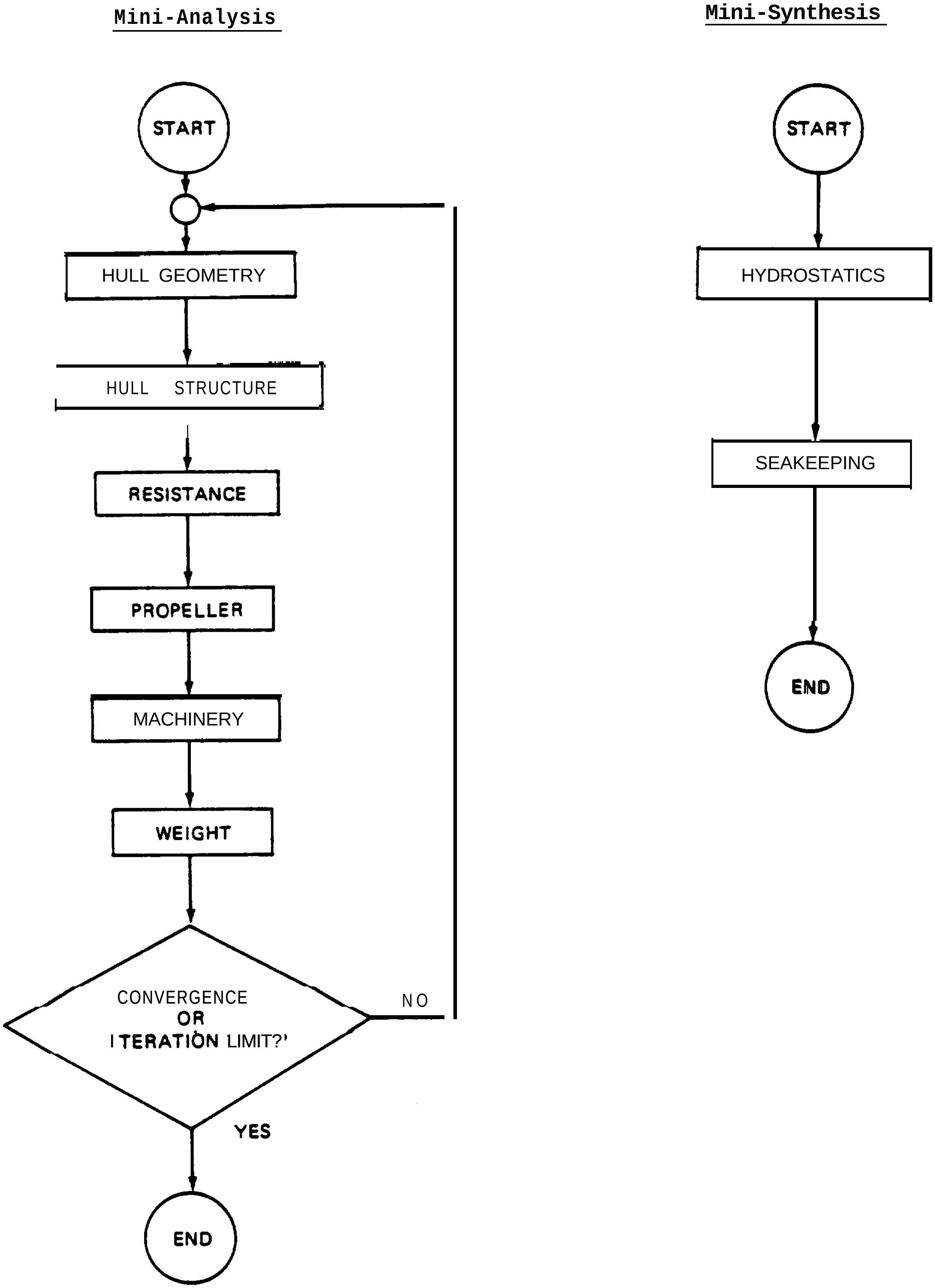


Fig. 3.2. Mini-Analysis and Mini-Synthesis Processes of Initialization Module.

SUBMODULE	FUNCTION
Hull Geometry	Establish principal dimensions of hull, including length, beam, draft, depth, hullform coefficients wetted surface area, and internal volume.
Hull Structure	Determine smeared thickness of primary and secondary hull structure, and establish hull and deckhouse material properties.
Resistance	Calculate ship resistance at design and range speeds.
Propeller	Perform propeller sizing and calculate propeller efficiencies at design and range speeds.
Machinery	Perform sizing of main and secondary propulsion machinery and of electric plant.
Weight	Calculate ship weights.

Table 3.1. Function of Mini-Synthesis Subroutines of Initialization Module

SUBMODULE	FUNCTION
Hydrostatics	Calculate intact and damaged GM and freeboard requirements.
Seakeeping	Estimate ship roll period.

Table 3.1. Continued. Function of Mini-Analysis Subroutines of Initialization Module

Mini-Synthesis process. The function of each Mini-Analysis submodule is listed in Table 3.1.

The list of input and output parameters to the INITIALIZATION module is contained in Table 3.2.

3.3. COST Module

The ASSET COST Analysis Module calculates ship acquisition and life cycle. The intent of the module is to provide data which can be used to evaluate the relative costs of competing systems of ships.

The ASSET COST Analysis Module consists of two principal sections. The first section pertains to ship acquisition costs. cost estimating relationships **(CERs)** are used to calculate lead and follow ship construction costs, profit, cost of change orders, NAVSEA support costs, post-delivery charges, outfitting costs, and costs of hull/mechanical/electrical plus growth. Construction costs are calculated as the sum of costs for each major Ship Work Breakdown Structure (SWBS) group. Principal data used by the **CERs** are weights categorized according to the SWBS and a series of user-specified cost factors that may be used to account for differing costs of differing technologies. The cost of ship payload may either be input by the user or estimated by the module.

The second principal section of the ASSET COST Analysis Module addresses life-cycle costs. Life cycle costs are considered in three major categories: research and development, investment, and operations and support. The life-cycle cost estimating algorithms utilize a wide variety of data to calculate costs in each of the three categories. The most important datum used by this section is an adjusted first ship cost, which is determined in the previous section. Other data include number of ships required, annual operating hours,

DEFAULT	PARAMETER NAME	ARRAY SIZE	UNITS ENGLISH	UNITS METRIC
	SHIP REQ			
	MISSION			
	DESIGN MODE IND			
	RANGE		MI	MI
	DESIGN SPEED REQ		UT	UT
•	RANGE SPEED REQ		UT	KT
•	MISSION DURATION		DAYS	DAYS
	PAYLOAD			
•	C+S ITEM WT ARRAY	(5X1)	LTON	MTON
•	ARM ITEM WT ARRAY	(5X1)	LTON	MTON
•	AMMO ITEM WT ARRAY	(4X1)	LTON	MTON
•	USABLE AV FUEL WT		LTON	MTON
•	CARGO WT ARRAY	(5X1)	LTON	MTON
•	NO CREW ARRAY	(3X1)		
	HULL			
	HULL GEOMETRY			
	HULL SIZE IND			
	LBP		FT	M
•	LBP/B		RATIO	RATIO
•	LBP/D		RATIO	RATIO
•	T/D		RATIO	RATIO
•	PRISMATIC COEF		RATIO	RATIO
•	MAX SECTION COEF		RATIO	RATIO
	HULL MATERIALS			
	HULL MTRL NPE IND			
	HULL MTRL DENSITY		LBM/FT3	KG/M3
	HULL MAX PRIM STRESS		KSI	MPA
•	HULL MARGINAL STRESS		KSI	WA
	HULL LOADS			
	HULL LOADS IND			
	HOGGING BM		FT-LTON	M-MTON
	SAGGING BM		FT-LTON	M-MTON
	DKHS GEOMETRY			
•	DKHS VOLUME FRAC		FRACTION	FRACTION
	DKHS MATERIALS			
	DKHS MTRL TYPE IND			
	DKHS MTRL DENSITY		LBM/FT3	KG/M3
	PROPULSION			
	WIN ENGINE			
	MAIN ENG NPE IND		-	-
	MAIN ENG SIZE IND		-	-
	MAIN NO ENG			
	MAIN CONT HP AVAIL		HP	KW
	WIN ENG SFC		LBM/HP-HR	KG/KW-HR
•	MAIN PWR MARGIN FAC			
	SEC ENGINE			
	SEC ENG NPE IND		-	-
	SEC ENG SIZE IND			-
	SEC NO ENG			
	SEC CONT HP AVAIL		HP	KW

Table 3.2. Parameters Used as Input to Initialization Module

DEFAULT	PARAMETER NAME	ARRAY SIZE	UNITS ENGLISH	UNITS METRIC
	SEC ENG SFC		LBM/HP-HR	KG/KW-HR
•	SEC PWR MARGIN FAC			
	ELECTRIC PLANT			
	ELECTRIC SYS KW IND			
	SSPU ENG TYPE IND			
	ELECTRIC SYS KW		KW	KW
	TRANSMISSION			
	TRANS EFF IND			
	TRANS TYPE IND			
	DESIGN TRANS EFF			
	RANGE TRANS EFF			
	PROPELLER			
	PROP TYPE IND			
	PROP SIZE IND			
	NO PROP SHAFTS			
•	THRUST DED COEF			
•	WAKE FRAC			
•	REL ROTATE EFF			
	PROP DIA		FT	M
	RESISTANCE			
	FRICTION LINE IND			
•	DRAG MARGIN FAC			
•	WORM CURVE ARRAY	(31X1)		
•	CORRELATION ALLOW			
	WEIGHTS			
	SHIP CG INPUT IND			
	FULL LOAD WT		LTON	MTON
	FULL LOAD CG ARRAY	(2X1)	RATIO	RATIO
•	WT MARGIN FACTOR			
•	WT ADJ ARRAY	(7X1)	LTON	MTON
	USABLE FUEL WT		LTON	MTON

Table 3.2. Continued. Parameters Used as Input to Initialization Module

Menu Number	Item Title
1	Summary
2	Hull Geometry
3	Hull Structure
4	Resistance
5	Propeller
6	Machinery
7	Weight
8	Hydrostatics and Seakeeping

Table 3.2. Continued. Initialization Module Printed Output Menus

fuel costs, fuel consumption rates, crew number **and** profile, rates of ship construction and learning, and the beginning and ending years of each of the three major life-cycle categories.

Additional COST Analysis Module features **include** use of NAVSEA allowances and format for presentation of acquisition costs, use of standard discounting (present worth) analysis to **reflect** variations in the time expenditures of alternative concepts, and use of an extensive set of default values for input data to help the user to quickly initiate cost studies. The module utilizes fiscal year 1981 base year dollars for its algorithms, but a variable inflation rate capability is provided to permit cost estimates to be expressed in any year dollars from FY 1977 through FY 1991. The rate is a constant 7 percent from 1984 on. The module is also sensitive to costs associated with the underway replenishment (UNREP) that will be used to support the ship fleet. The calculation sequence employed by the COST Analysis Module is a seven-step process.

The current model is the sole source of input data for the COST Analysis Module. A listing of all current model data used as input by the module is given in Table 3.3.

Three printed output items can be produced by the COST Analysis Module. The menu of printed output items for the COST Analysis Module is shown in Table 3.3, which gives the menu number corresponding to each printed output item.

3.4. SPACE Module

The ASSET SPACE Analysis Module presents the designer with an estimate of what internal volume and deck area the current ship model requires. The space required estimation output is not used in any

DEFAULT	PARAMETER NAME	ARRAY	UNITS	
		SIZE	ENGLISH	METRIC
	SHIP REQ			
	MISSION			
	RANGE		MI	MI
	RANGE SPEED REQ		KT	KT
	PAYLOAD			
•	C+S ITEM WT ARRAY	(5X1)	LTON	MTON
•	ARM ITEM WT ARRAY	(5X1)	LTON	MTON
•	AMMO ITEM WT ARRAY	(4X1)	LTON	MTON
•	USABLE AV FUEL WT		LTON	MTON
•	CARGO WT ARRAY	(5X1)	LTON	MTON
	NO CREW ARRAY	(3X1)		
•	AIRCRAFT VOL REQ		FT3	M3
	PROPULSION			
	MAIN ENGINE			
	MAIN NO ENG			
	MAIN CONT HP AVAIL		HP	KW
	SEC ENGINE			
	SEC ENG TYPE IND		-	
	SEC NO ENG			
	SEC CONT HP AVAIL		HP	KW
	WEIGHTS			
	FULL LOAD WT		LTON	MTON
	SHIP WT ARRAY	(8X1)	LTON	MTON
	USABLE FUEL WT		LTON	MTON
	COST FACTORS			
	ECONOMIC FACTORS			
•	YEAR $			
•	INFLATION RATE ARRAY	(15X1)	PERCENT	PERCENT
•	PRODUCTION RATE		SHIPS/YR	SHIPS/YR
•	LEARNING RATE			
•	FUEL COST		$/US GAL	$/L
	PAYLOAD COST FACTORS			
•	PAYLOAD T+E COST		$M	$M
•	LEAD PAYLOAD COST		$M	$M
•	FOLLOW PAYLOAD COST		$M	$M
•	ANNUAL TRNG ORD COST		$M	$M
•	PAYLOAD FUEL RATE		LTON/HR	MTON/HR
	SHIP COST FACTORS			
	IOC DATE		YR	YR
•	R+D PROGRAM LENGTH		YRS	YRS
•	NO OF SHIPS ACQUIRED			
•	PROFIT FRAC			
•	SERVICE LIFE		YRS	YRS
•	ANNUAL OPERATING HRS		HR	HR
•	TECH ADV COST		$M	$M
•	ADDL FACILITY COST		$M	$M
•	DEFERRED MMHRS REQ		HR/WK	HR/WK
•	UNREP UNIT CAPACITY		LTON/YR	MTON/YR
•	UNREP UNIT COST		$M	$M
•	UNREP O+S COST		$M	$M
•	KN FACTOR ARRAY	(9X1)		
•	SHIP FUEL RATE		LTON/HR	MTON/HR

Table 3.3. Parameters Used as Input to Cost Module

other modules and is only intended to guide the designer in the preparation of general arrangement.

The space required within any ship can be broken down into two types, namely, internal deck area and volume. The internal deck area required is estimated where equipment or space are normally located within that part of the ship where a standard deck height exists. Most of the space within a ship is utilized in this manner. The volume required is estimated where the nature of the space does not permit the deck area method to be employed. Examples are the main machinery spaces, the helicopter hanger, and fuel tanks. The total space required can be found by multiplying the internal deck area required by the average internal deck height and adding that to the volume required.

The proposed U.S. Navy Ship Space Classification System (SSCS 1969) has been used as the basis for classifying shipboard spaces. Under this system, shipboard space is divided into three primary categories, indicated by the first digit of the group number. Each succeeding digit represents a further subdivision of the superior subdivision. A unit of space is classified by the assignment of a complete four-digit group number. Because it is impossible to restrict the ASSET synthesis to the four-digit level, the lowest possible level is estimated with summaries provided at the three- and two-digit levels where appropriate.

The current model is the sole source of input data for the SPACE Analysis Module. A listing of all current model data used as input by the module is given in Table 3.4. The SPACE module input includes: Mission Duration, Payload, Actual Ship Volume, Engine Number and Rating, Weights, and Subsystem Data.

DEFAULT	PARAMETER NAME	ARRAY SIZE	UNITS ENGLISH	UNITS METRIC
	SHIP REQ			
	MISSION			
·	MISSION DURATION		DAYS	DAYS
	PAYLOAD			
·	C+S ITEM WT ARRAY	(5X1)	LTON	MTON
·	AMMO ITEM WT ARRAY	(4X1)	LTON	MTON
·	USABLE AV FUEL WT		LTON	MTON
	NO CREW ARRAY	(3X1)		
·	ARM SPACE ARRAY	(4X1)	FT2	M2
·	AIRCRAFT VOL REQ		FT3	M3
	HULL			
	HULL GEOMETRY			
	LBP		FT	M
	LBP/B		RATIO	RATIO
	HULL VOLUME		FT3	M3
	DKHS GEOMETRY			
	DKHS VOLUME		FT3	M3
	PROPULSION			
	MAIN ENGINE			
	MAIN NO Effi			
	MAIN CONT HP AVAIL		HP	KW
	SEC ENGINE			
	SEC ENG TYPE IND			-
	ELECTRIC PLANT			
	SSPU ENG TYPE IND			
	ELECTRIC SYS KW		KW	KW
	WEIGHTS			
	FULL LOAD WT		LTCN	MTON
	SHIP WT ARRAY	(8X1)	LTON	MTON
	USABLE FUEL WT		LTON	MTON

Table 3.4. Parameters Used as Input to Space Module

Four printed output items can be produced by the Space Analysis Module. The menu of printed output items for the Space Analysis Module is shown in Table 3.4.

4.0 OPTIMIZATION TECHNIQUES

During the last 20 years, major advances in engineering analysis and computer technology have been realized. This has lead to an emphasis on automated design and optimization in all fields of science and engineering. Automated design may be realized by a number of design methods, or numerical optimization techniques. These techniques are very flexible and can solve a large percentage of practical design problems.

4.1. Optimization Method Definitions

The definition of terminology utilized in optimization methods includes:

Design Variables: Those parameters which the optimization technique is allowed to change in order to improve the design. In ship design these might include length, beam, prismatic coefficient, etc. Two types of design variables, may be considered, independent and dependent. If two or more variables are always reauired to have the same value or be in a constant ratio. one is the independent variable, while the remaining ones are dependent variables.

Constraints: The design restrictions which must be satisfied in order to produce an acceptable design are collectively called constraints and may include any parameter which must not exceed specified bounds for the design to be acceptable. There are two kinds of constraints:

Side-: A constraint that restricts the range of a design variable for reasons other than the direct consideration of performance, i.e., minimum draft.

Behavioral Constraint: A constraint that restricts the quantities which characterize the state of the model.

Objective Function: The parameter which is to be minimized or maximized during optimization. It must be a function of the design variables to be meaningful.

Feasible Design: A design which satisfies the specified imposed constraints is called a feasible or acceptable design.

Infeasible Design: A design in which constraints are violated is called an infeasible or unacceptable design.

4.2. COPES/CONMIN optimization Program

The most general problem of design optimization can be stated as: From all designs that satisfy certain constraints, select one which optimizes (maximizes or minimizes) a specified set of design variables. Numerical optimization procedures are used to solve the n-dimensional, non-linear, constrained or unconstrained optimization problems. Two of the most powerful methods are the "Method of Feasible Directions" for the constrained problem and the "Conjugate Directions Method" for the unconstrained problem. These two methods are the primary ones used in the COPES/CONMIN optimizer [17]. COPES, which is a FORTRAN Control Program for Engineering Analysis, uses the optimization program CONMIN, CONstrained function MINimization.

The COPES/CONMIN program is a general purpose, non-linear optimizer capable of handling large, constrained problems. It solves the general non-linear optimization problem stated as follows:

Optimize (minimize or maximize) $F(\bar{X})$

Subject to $G_i(\bar{X}) \leq 0;\ i = 1,m$

$$X_i^l \leq X_i \leq X_i^u \qquad (4.1)$$

where: F(X) is the objective function (function to be minimized or maximized).

The vector $\bar{X}$ contains the n design variables. $G_i(X)$ defines the constraints which are imposed by the designer on the optimization process where m is the total number of constraints. X_i^l and X_i^u are lower and upper limits, respectively, of the design variables. The terms F(X) and $G_i(X)$ may be explicit or implicit functions of the design variables X, but must be continuous. If the constraint inequality condition is violated for any constraints, that is $G_i(X) > 0$, that constraint is said to be violated. This situation may arise many times during the optimization Process, and the information will be used to guide the design to one which satisfies all of the constraints. If equality occurs in Eqn. 1, $G_i(X) = 0$, the constraint is said to be active, and if the inequality is met, $G(X)_i < 0$, the constraint is inactive For practical reasons, on a digital computer a constraint is active if its value is within a specified tolerance. The n-dimensional space consists of the n design variables X. Any design in this space which satisfies Eqn. 1 is defined as a feasible design. The failure to satisfy Eqn. 1 leads to an infeasible design, but is still useful in the process of reaching a feasible design. The feasible design which is at a minimum or maximum is said to be an optimal design.

It should be noted at this point that maximization of a function is the same as minimization of the negative of that function. Thus any design problem can be cast in the above form. The optimization program begins with an initial input X vector which may or may not define a feasible design. It should be emphasized that the starting point can be an infeasible design. The program has the ability to end up at a feasible design, which is the most powerful feature of this method.

4.3. Automated Optimization Algorithm

In the case of a ship synthesis, the initial vector $\{\bar{X}\}$ defines the designer's initial estimate of the ship data (displacement, performance, etc.).

The optimization process than proceeds iteratively by following the relationship

$$X^{n+1} = X^{n} + aS^{n} \tag{4.2}$$

where n is the number of iterations, S is a vector which indicates the direction of search in the non-dimensional design space and "a" is a scalar which defines the distance of travel in direction S.

The optimization process then proceeds in two steps:

(1) A direction S is determined which will reduce the objective function without violating constraints.

(2) The scalar "a" is determined so that either the objective function is minimized in the direction S, or a new constraint boundary is encountered.

Once these two steps are completed, the current situation is stated as follows: either the objective function has improved towards and has reached an optimum or at least a local minimum of the objective function, or no further improvement can be made in this direction and it is necessary to determine a new S vector, which will improve the design without violating the constraints. This continues on until the optimal design has been achieved at the point where no direction exists which will reduce the objective function further without violating the constraints. The method of feasible directions is described in detail in Vanderplatts [18].

4.4. Design Optimization Example

The following simple example of a two-variable design problem illustrates the algorithm of the method.

The life cycle cost per ship of a naval combatant is to minimized as a function of the following two design variables: length between perpendicular (LBP) and prismatic coefficient (CP), subject to two constraints. Constraint Cl requires that the calculated full load displacement is within a ten percent tolerance! of the estimated displacement based on an empirical formula. Constraint C2 requires that the minimum intact GM is equal to or greater than the estimation by a similar empirical formula.

The graphical representation of such a problem can be seen in Figure 4.1, where the lines of displacement represent constant value contours. Assume that point A is chosen as the initial design which satisfies our requirements. Then the program proceeds through the following steps:

a) Each of the X variables (design variables) is perturbed to determine its effect on the life cycle cost per ship (objective function). The gradient of the life cycle cost per ship function is calculated by the finite difference method. Because at this point no constraints are active or violated, the greatest improvement in the objective function (minimization) is obtained by moving in the negative gradient or steepest descent direction, so that S = -V(SYSCPS)

b) After the S determination, the scalar "a" in Eqn. 2 must be determined so that either the objective function is

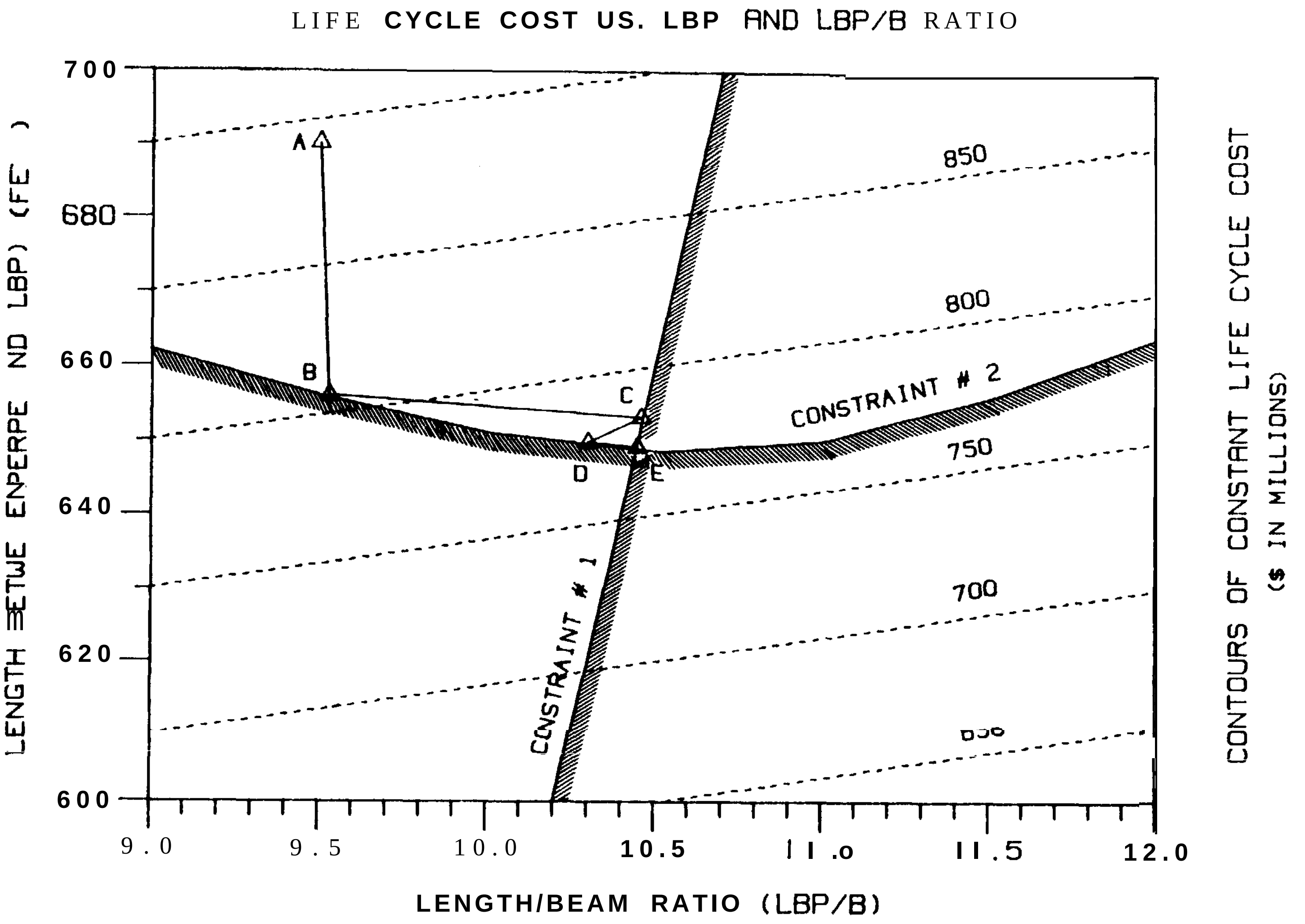


Figure 4.1. Two Design Variable Design Space for Minimization of the Life Cycle Cost as a Function of the Length Between Perpendiculars, and the Length to Beam Ratio

minimized in this direction or some constraint boundary is encountered. A one-dimensional search is done in the direction S to determine the appropriate value for **"a"** so that an improved design can be achieved at point B. Because the constraint Cl is encountered, no further improvement can be achieved in this direction, without violating the Cl constraint.

c) A new direction is then found which will reduce the objective function without violating the Cl constraint. Such a direction can be found by solving a linear programming subproblem with a single quadratic constraint For more details on the solution of such a problem, see Vanderplatts [18]. If no such direction exists, then the current point is considered a global or at least a local optimum. In this example such a direction exists and the design proceeds from point B to C, where the constraint C2 is encountered.

d) The subproaram is then solved again, resulting in a further reduction of the objective function and an active constraint at point D.

e) Finally, from point D, the one-dimensional search yields a solution at point E, which is the vertex of the constraints and as may be easily seen, the minimum life cycle per ship design point. At this point, the problem perturbs the design variables to obtain the gradient of the objective function and both active constraints or the linear subproblem is solved once more. The solution this time is zero, which means that an **optimum** has been achieved. Point E is clearly the optimum, since no direction exists at this point which will reduce the objective function any further without violating one or both of the constraints.

It should be pointed out that in conceptual ship design, it is possible that the initial starting point in the design space may be in the infeasible region. The program will. proceed toward the feasible region. with a minimal increase in the objective function. The above described method is directly extendable to the n-dimensional problem. Additional constraints may also be added without increasing the complexity of the design process.

4.5. COPES/CONMIN Organization

The purpose of the COPES program is to provide automated design and tradeoff capability. The user must provide a FORTRAN analysis program, in this case, ASSET, for the analysis of the particular problem being considered. This analysis program is written according to a simple set of guidelines so that it can be easily coupled to the COPES program for automated design synthesis. The main task of COPES/CONMIN is to read and organize data which identify the objective function, design variables and constraints, to couple the analysis code to the optimization routine and finally to perform optimization. There are some simple requirements for using this capability:

a) The analysis routine must be written in a standard language for easy transfer from one computer to another, with subroutine ANALIZ as the main routine.

b) The analysis routine must be segmented into input, execution and output, with the code written in subroutine form, so that it may be called for execution with different values of the design variables.

c) All parameters, which may be design variables, the objective function or constraints, must be stored in a. single labeled common block called **GLOBCM,** for easy access by the optimization program.

The analysis code is called for input only once, but may be called for execution many times during the design optimization process. The analysis code is called twice for output, first to print the results of the analysis of the initial design, and again to print the results for the optimized design. The analysis code may be called for output more than twice at the user's option to print intermediate design information. In order to execute the COPES program, it is necessary to provide formatted or unformatted data, followed by data for the **ANALIZ** subroutine which is coupled to COPES.

5.0 COMPUTER GRAPHICS

While the most important advantage of the optimization program is the ability to find optimal solutions quickly and thus inexpensively, there can be certain disadvantages to using it. Lack of perception of the optimal design and lack of information in the neighborhood of the optimum point make designers hesitant to use these techniques. The COPES/CONMIN optimization program previously described can solve the problem of lack of information around the optimum point. This is because the design can begin as an infeasible design and proceed to a feasible optimum one. The COPES program maintains a record of all the designs tried, which the designer may then examine and use for further information and design decisions.

The COPES program also identifies those constraints and design variables that are active or violated: therefore the designer will have information on what is most critical to the design. But while the above features are of assistance to the designer, the output will still be simply a number which is left to the designer to accept. Moreover, the complexity of the design optimization problem is indicated by the fact that some design input variables are varied to reach an optimum of the objective function. The objective function itself may vary because of its dependence on other parameters which also change for different input data through the iteration process..

5.1. Optimization Graphics

The visualization of the optimization process is complex. Therefore, graphical perception of the design optimum is necessary to make the engineer feel more confident,.as it provides him with an additional check of the solution.

While the automated optimization routine, COPES, handles the numerical aspects of the problem, a graphic visualization of the process is desirable. An interactive computer graphics capability clearly would be beneficial to this process. The best presentation of an objective function as a function of some design variables is through the use of 2- and 3-D computer graphics. The use of 3-dimensional mesh perspectives allows direct viewing of the objective function as a function of two variables. The contour quickly locates the maximum design point in addition to locating nearby maximums which might also be of interest to the designer. These graphics give a complete picture of the design space for any two design variables at a time. If the design variables number more than two, the variation of the objective function can be represented by a greater number of diagrams, on which the engineer must make some kind of judgment.

To make the visualization of the optimum design even clearer, or to vary the objective function with each one of the design variables separately in 3-D space, projection of the 3-D diagram onto the two vertical planes, 2-D portrayal, is desirable. This is especially important, because in ship design the condition of flat-laxity is noted and an explicit presentation of the design space may be needed. Thus a complete graphical output consists of the general 3-D plot of the design space complemented by the two 2-D projections.

5.2. Graphics Programs

From the available graphics programs at the Academic Computer Center at the University of Washington, the following two were selected for the needs of this study:

1) PICTURE - for three-dimensional displays
2) SIMPLOT - for simple or complex 2-D diagrams.

5.2.1. PICTURE

PICTURE is a FORTRAN subroutine that produces perspective displays of three-dimensional surfaces on a drum plotter, the Gould electrostatic plotter, Tektronix graphics terminals, or other plotting devices. It can remove hidden lines, draw both the upper and lower sides of a surface, and draw a perspective box surrounding the surface. The subroutine was developed by Melvin Prueitt at the Los Alamos Scientific Laboratory, [19].

The user must provide a data file through some simple FORTRAN arguments. Based on this data file, PICTURE creates a three-dimensional array, with specific values for each **X,Y,Z** point. How the user wants the array depicted, such as selecting view point, scale factor, enclosing box frame dimensions, etc., is variable through specification of the corresponding parameters. More about PICTURE's features and uses may be found in the PICTURE user's guide in Appendix B.

5.2.2. SIMPLOT

SIMPLOT is the University of Washington **SIMple PLOTting** system [20]. SIMPLOT can be used to display pictures and diagrams made up of points, lines, wires, bars, etc., that represent a series of numbers or variables given to SIMPLOT.

SIMPLOT runs through a set of 18 simple commands. These commands are of three types:

a) descriptive commands, which describe the numbers to be displayed (input data) and the the output device
b) plotting commands, which tell SIMPLOT which of the numbers in the input data to display and how to display them
c) optional commands, which tell SIMPLOT what extra things, if any, are to be included in the display (labels, legends,

statistics), and how the display is to differ from the standard picture layout, if at all..

All the above commands must be contained in a command file, which is read by SIMPLOT upon execution. More details about SIMPLOT's use and features may be found in the SIMPLOT Users Manual in Appendix C.

5.3. Design Example

As a a non-automated example of the optimization/graphics design procedure, the analysis program HYCAT was selected, Calkins [21]. HYCAT is a computer program which was written to compute the foilborne performance of a hybrid Hydrofoil CATamaran, HYCAT. The program is based on a lift and drag performance analysis algorithm. The inputs to the program are hull beam, foil aspect ratios and material, gallons of fuel, number of passengers and number of engines. The program then produces a geometry statement, a weight statement and a performance statement, in addition to 2-D plots of range versus speed for each value of aspect ratio and hull beam. The computer model assumes that the foils have incidence control and are adjusted so that the hull trim is zero degrees over the foilborne speed range. The problem then is to determine the maximum foilborne range as a function of the speed and hydrofoil aspect ratio while behavioral constraints on the speed and the thickness to chord ratio are imposed. The problem stated mathematically is:

$$\max R = f(ARF, ARA, U) \text{ for } BM = \text{constant}$$

where:

R = range (N.M.)

ARF = aspect ratio forward

ARA = aspect ratio aft

U = speed (kn)

BM = hull beam (ft)

Geometric constraints:

$$20.0 \leq U \leq 50.0$$
$$1.0 \leq ARF \leq 10.0$$
$$1.0 \leq ARA \leq 10.0$$

Behavior constraints:

$$U \leq U_{max} \text{ or } \frac{U}{U_{max}} \leq 1.0$$
$$U \leq 0.9\, U_{cavf} \text{ or } \frac{U}{U_{cavf}} \leq 0.9$$
$$U \leq 0.9\, U_{cava} \text{ or } \frac{U}{U_{cava}} \leq 0.9 \qquad (5.1)$$
$$0.04 \leq TCF \leq 0.2$$
$$0.04 \leq TCA \leq 0.2$$

where:

U_{max} = maximum speed imposed by the maximum horsepower

U_{cavf} and U_{cava} = cavitation speed forward and aft

TCF and TCA = thickness/chord ratio forward and aft.

The program HYCAT was run for cases covering the operating speed range for foil aspect ratios from 1 to 15, Pantazopoulos [22]. From the output data, a sensitivity study of the design variables was made. Figure 5.1 shows a typical plot of range versus speed for aspect ratios of 8 forward and aft, in addition to the speed constraints.. Figure 5.2 shows the variation of horsepower as a function of speed. For the maximum installed SHP of 3200, the corresponding maximum speed was determined as indicated in Figure 5.2. Figure 5.3 shows the variation of required thickness/chord ratio as a function of aspect ratio with the imposed constraints also shown. Cavitation speed is shown in Figure 5.4, as a function of aspect ratio, and as a function of speed corresponding to maximum range in Figure 5.5. Individual

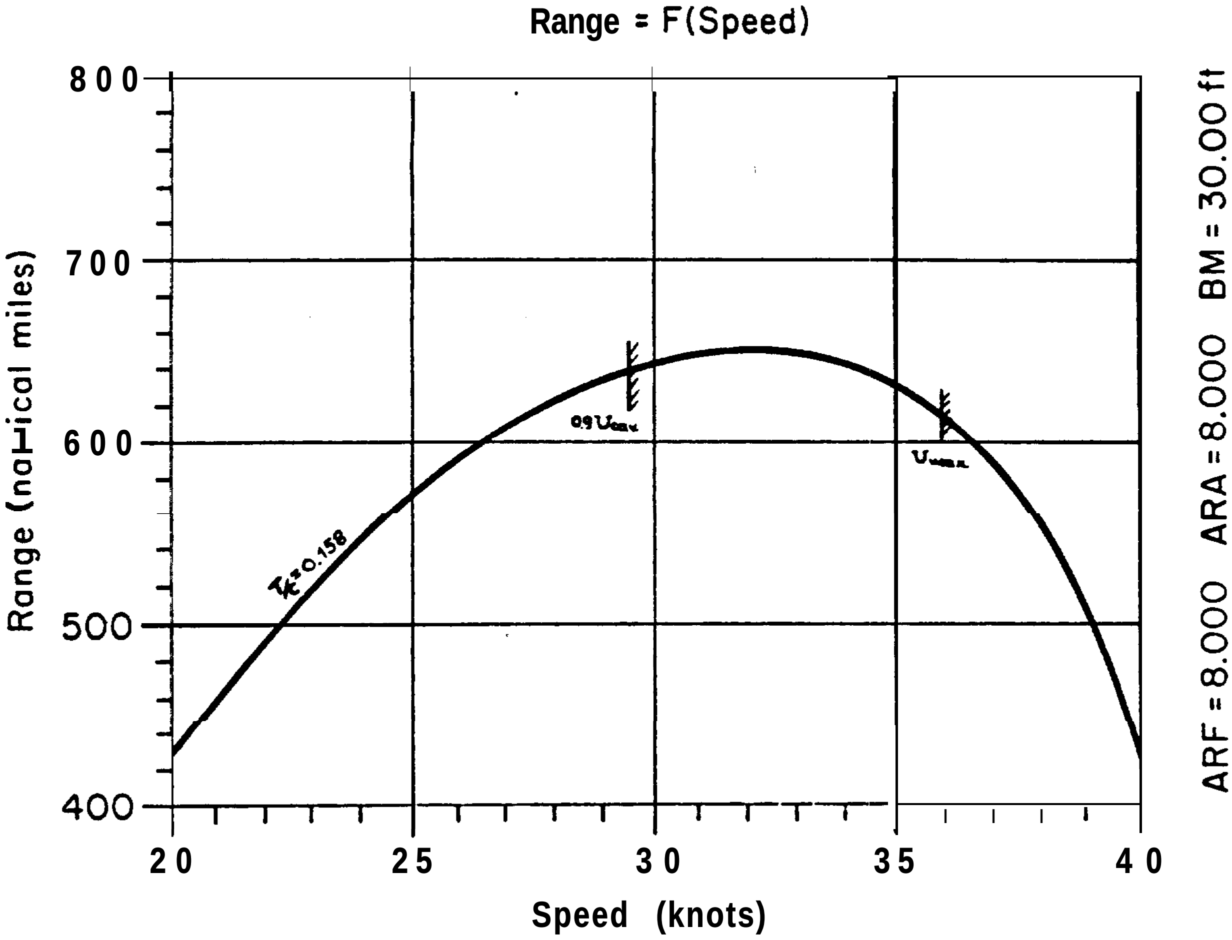


Fig. 5.1. Variation of range versus speed for aspect ratio 8.0 .

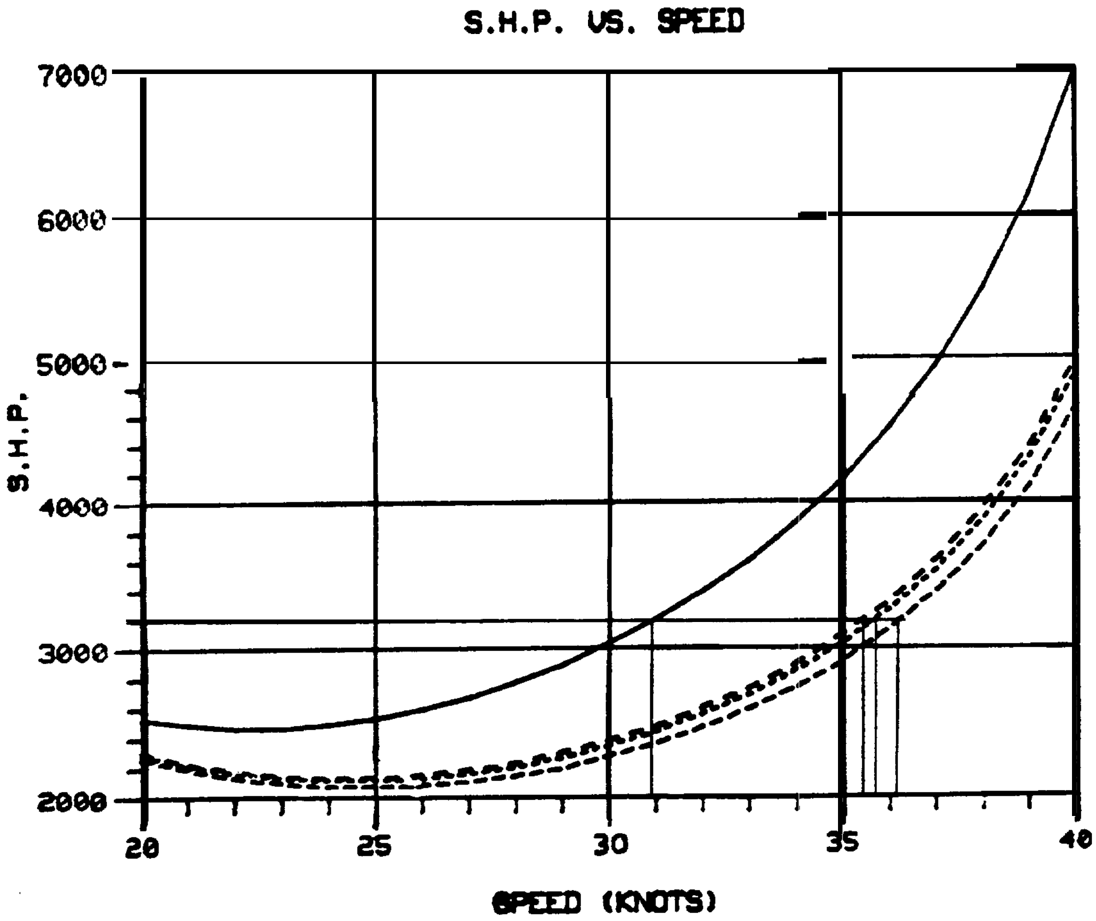


Fig. 5.2. Variation of SHP versus speed (aspect ratios:1.0,5.0,10.0)

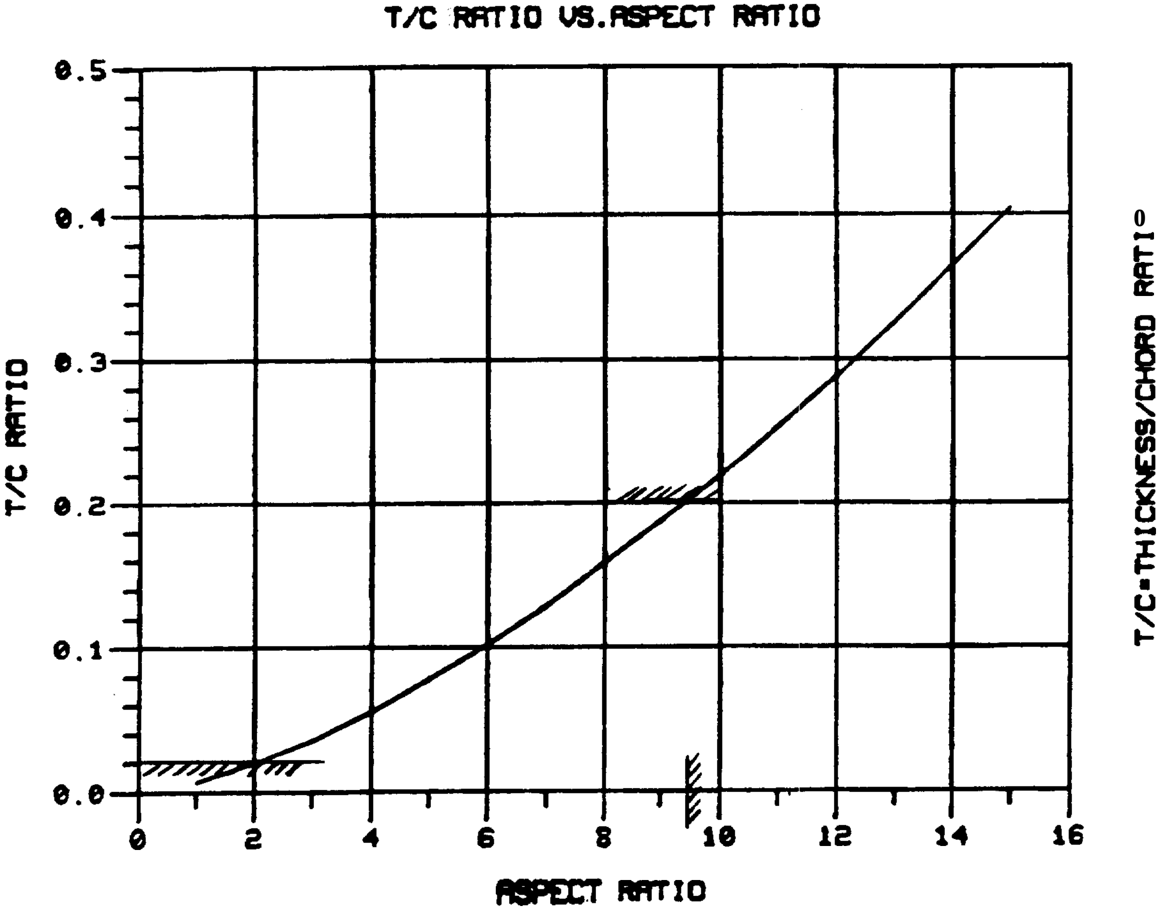


Fig. 5.3. Variation of t/c versus aspect ratio

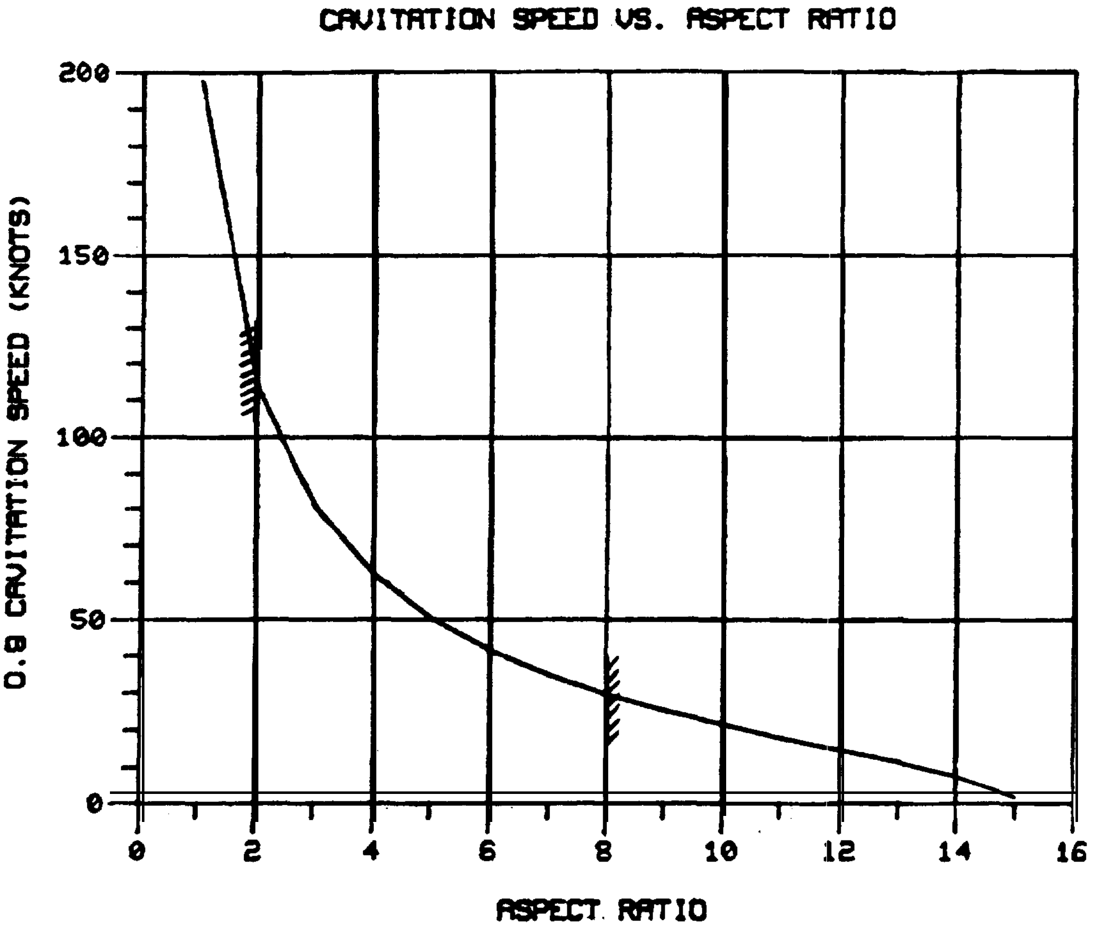


Fig. 5.4. Variation of cavitation speed versus aspect ratio

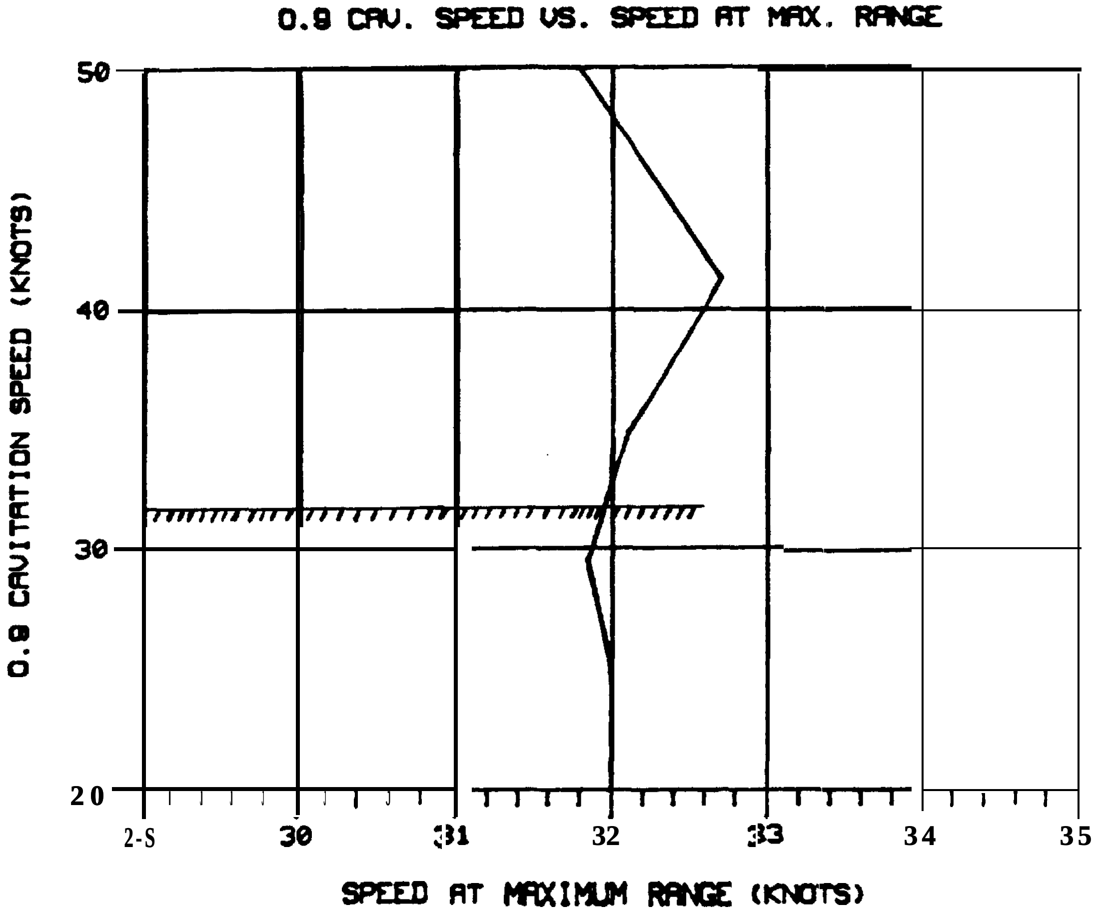


Fig. 5.5. Variation of cavitation speed versus speed at maximum range.

studies of each constraint in consideration of the above sensitivity studies finally led to the defined limited design space shown in Figure 5.6. The maximum range for each aspect ratio was also found after the constraints were imposed, Figure 5.7, and the desired maximum range was found. Also shown in Figure 5.7 is the variation of maximum range versus aspect ratio without the imposed constraints to examine their effect on the objective function.- The above manual method consumed a great deal of time, especially in the analysis process, to determine the optimum (maximum) range.

For this problem, a 3-D computer graphics routine which generates a perspective view of the three-dimensional design space was used to visualize the location of the optimum. PICTURE was selected as the most versatile tool to fulfill these needs. The data file created by HYCAT mode was used as input for the 3-D PICTURE program. Three-dimensional perspective plots are shown in Figure 5.8, with:

OX axis - aspect ratio
OY axis - speed
OZ axis - foilborne range.

The design constraints are also shown on the plot to define the feasible design space where optimization may proceed. The location of the maximum range is easily visualized.

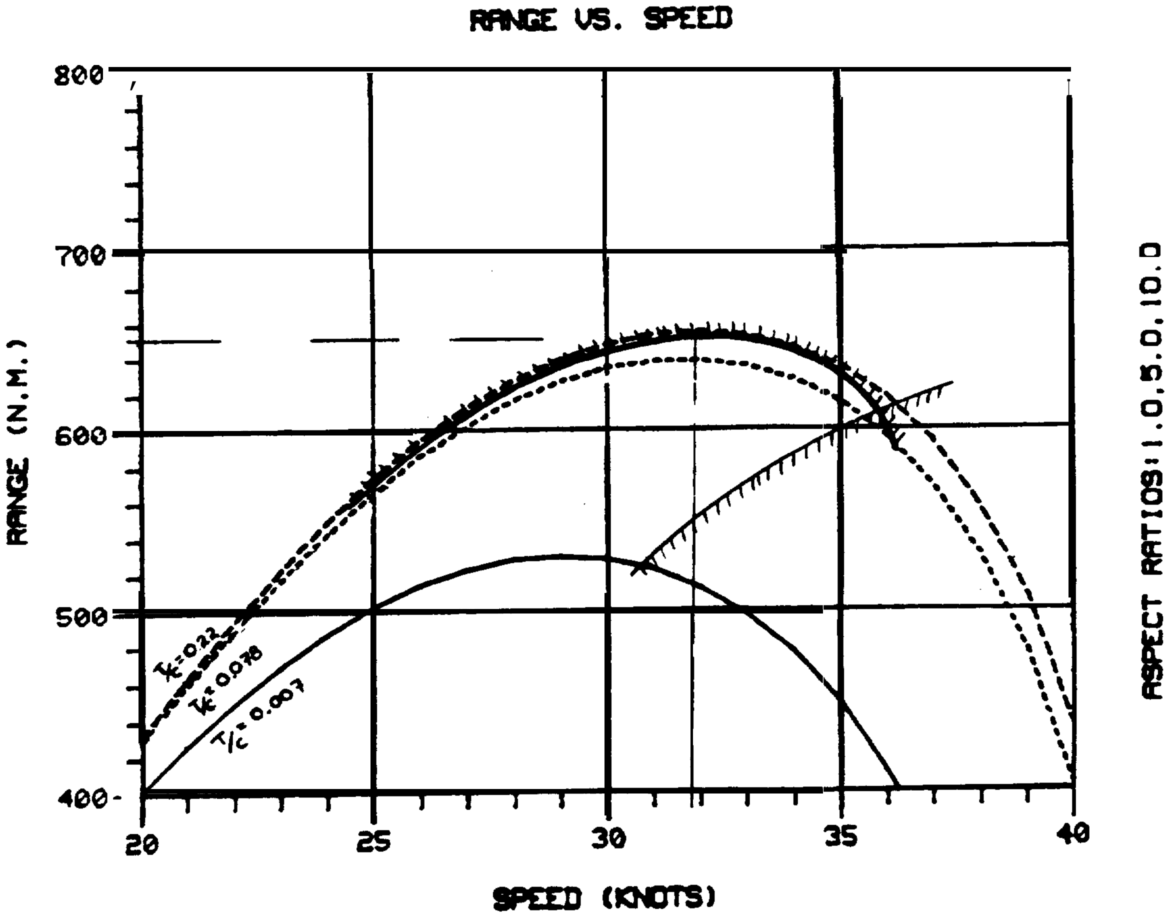


Fig. 5.6 Variation of range versus speed
Constraints are imposed on the design space.

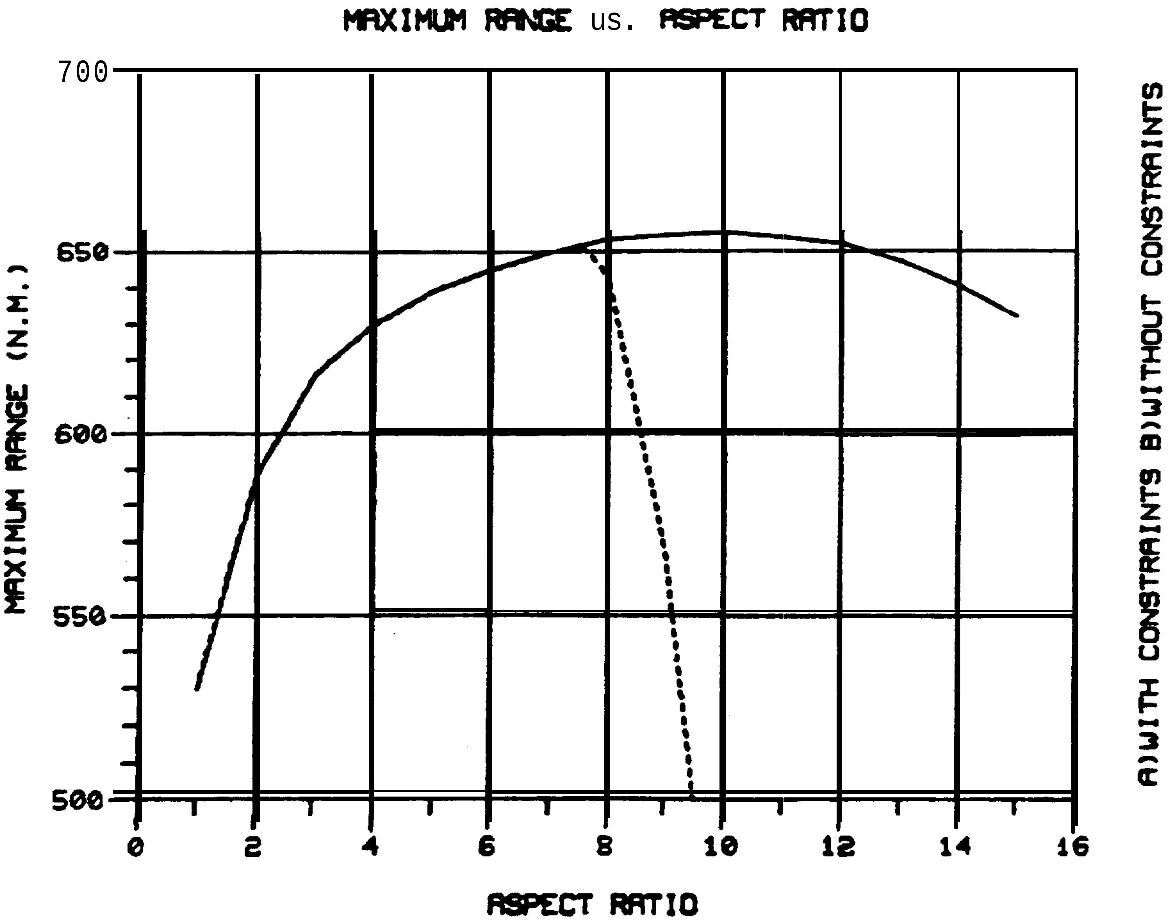


Fig. 5.7. Variation of maximum range versus aspect ratio
---- with constraints,——— without constraints

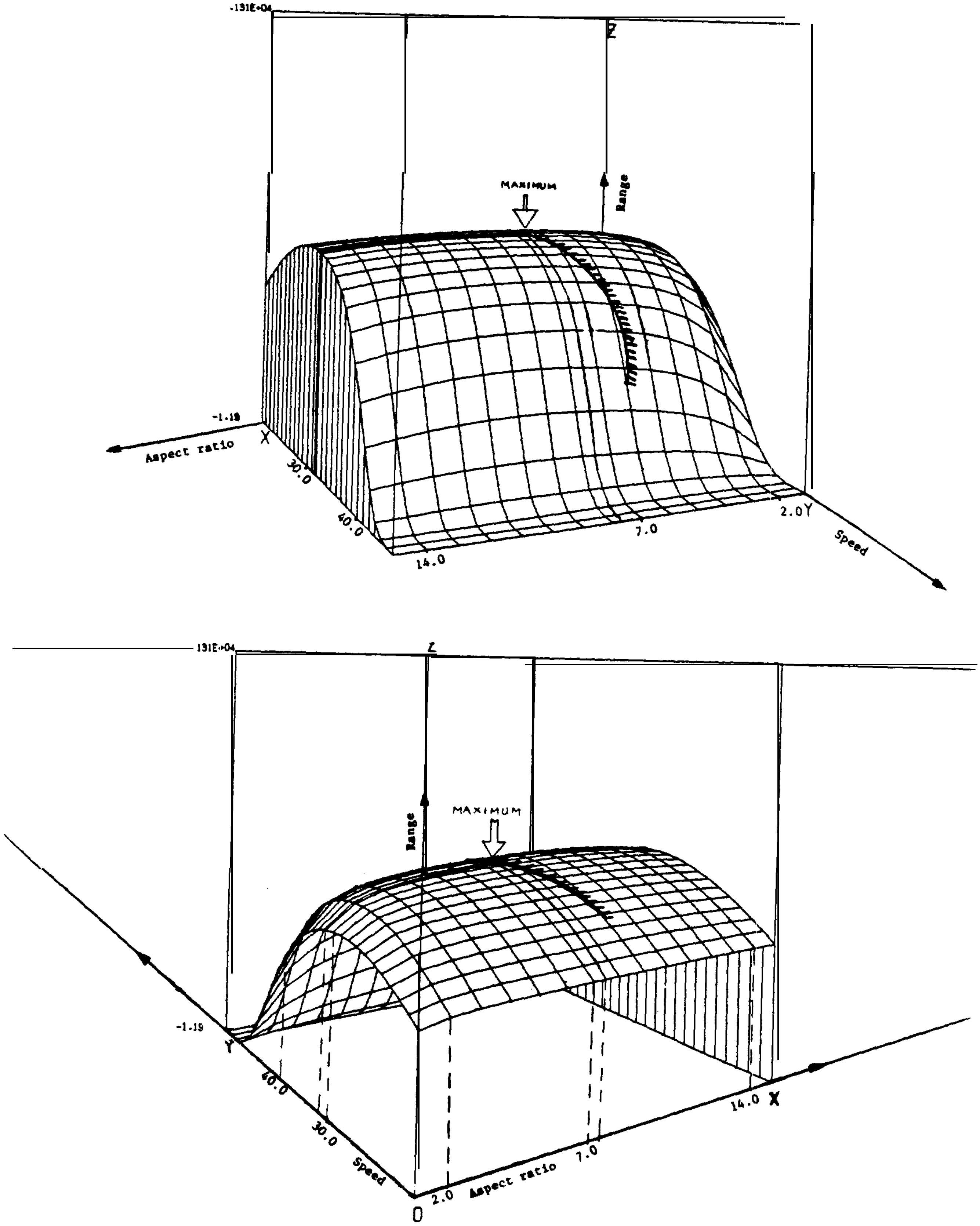


Fig. 5.8. 3-D plot for visualization of maximum range .

6.0. ASSET/COPES ENSEMBLE

In this section, the coupling process of the **INITIALIZATION, COST** and SPACE modules of the ASSET program with the necessary control subroutines and the COPES/CONMIN optimizer is presented. In this effort, the goal was to introduce the fewest possible modifications to the three modules, so that their structure and organization would not be changed drastically from their format in the ASSET program.

6.1. Design Variable Selection

One of the most important decisions in the development of the **ASSET/CONMIN** model is the selection of design variables.. The ship designer must chose the critical variables for the design process from a wide spectrum of parameters.

A review of the current literature shows that each synthesis model selects the design variables based on different criteria. There are, however, some generally acceptable parameters used in all the models. Both the optimization methods of Murphy, Sabat and Taylor [13] and of Mandel and Leopold [14] use displacement (A), prismatic coefficient (C_p), speed-length ratio, beam-draft ratio (B/T), and length-depth ratio (L/D) as the design variables. Lewis [14] later added the midship section coefficient (C,) to the above five variables. The Canadian CEM [15] used the load waterline length (LWL), length-displacement ratio (L/A), prismatic coefficient (C_p), block coefficient (C_v), beam-draft ratio (B/T and length-depth ratio (L/D) as design variables. Watson [15] proposed replacing the block coefficient, beam-draft ratio and length-depth ratio with length-beam ratio (L/B), beam-depth ratio (B/D) and draft-depth ratio (T/D). Recently Jenkins [16], with the REED/COPES model, chose as design variables the length between perpendiculars (LBP), length-beam ratio (L/B), beam-draft ratio (B/T), prismatic coefficient (Cp) and midship coefficient (C_x).

As is obvious from the above, there are some generally acceptable parameters used as design variables. Length is one of the major dimensions involved in ship design, and it is apparent that the displacement and cost of a ship will be dependent on the length. This relationship is seen in Figure 6.1. which shows the correlation between length and displacement of all conventional types of naval combatants of the U.S. Navy. The following statement by Saunders [23] supports the selection of length as a design variable:

> In the group of underwater form coefficients and parameters developed through the years, the ship length logically appears as one of the principal dimensions. It is related directly and indirectly to the beam, draft, displacement weight, displacement volume, and to many other factors.

The dimensionless form coefficients are very useful parameters in ship size and form estimation and are conveniently used because of their non-dimensional character. The prismatic coefficient, C_p, and midship section coefficient, C_x, give a good indication of the ship form and size. Accordingly, the prismatic coefficient, which indicates the fullness of the underwater hull, was selected as the second design variable.

The midship section coefficient, C_x, was selected as the third design variable. This coefficient relates the area of the midship section to the area equal to the beam and the draft at that section. It is useful in the estimation of the hull strength, the initial power requirements and is also important to the ship motions in a seakeeping study.

Finally, for the level of the present study, the length-beam ratio, L/B, was selected as the fourth design variable. This

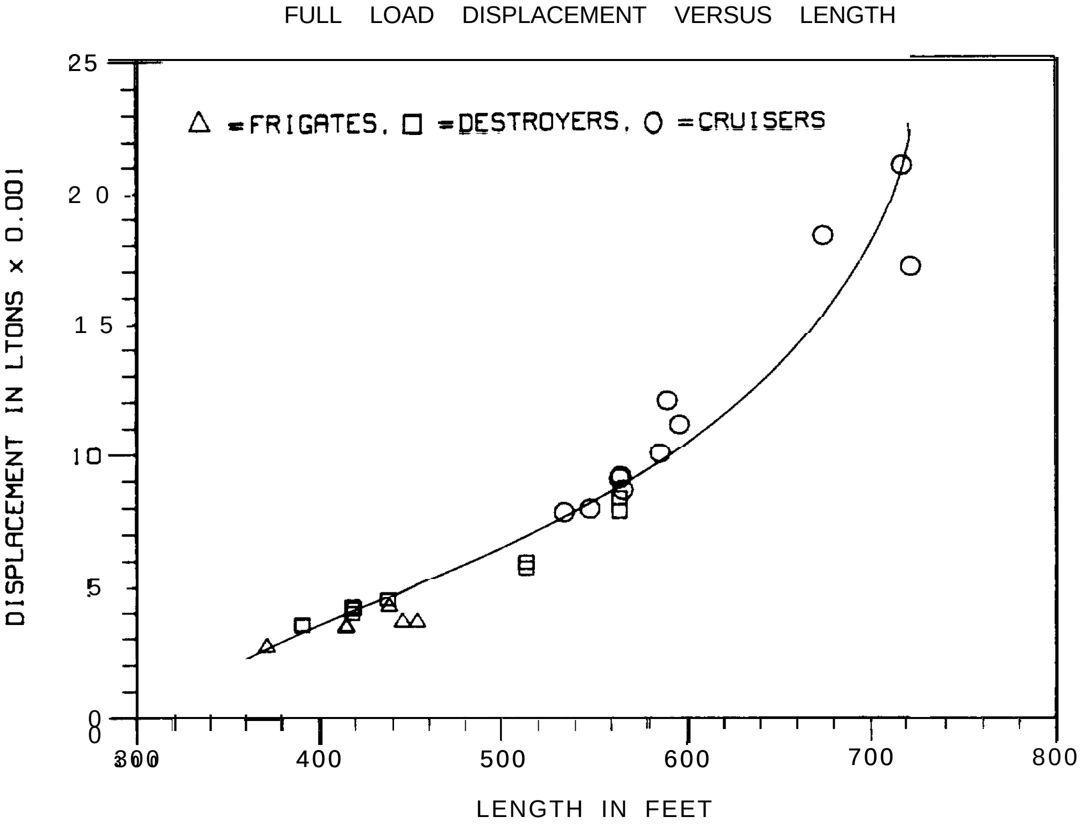


Fig. **6.1.** Correlation of Full Load Displacement and Length for the Conventional Types of the U.S. Navy Combatants dimensionless coefficient is fundamental to powering and maneuverability calculations. Consideration of the list of available input variables for the INITIALIZATION module excluded selection of additional design variables, such as other ship form coefficients and dimensions. It should be emphasized, however, that the designer may designate any of the other variables used in this model as design variables.

6.2. Objective Function Selection

An equally important decision is the selection of the objective function. An extensive review of literature pertaining to conceptual ship design and optimization models showed that there is no unique approach to the problem. Indeed, there are two factors which seem to dominate the field: cost and size. Cost is expressed as acquisition or profit, and size as displacement or volume. Thus, Nowacki [24] worked with a single economic figure of merit, required freight rate (RFR), for the objective function of commercial vessels. Mandel and Leopold [14] used a three-term weighted optimization criterion as an objective function. In this criterion, the first term represents the cost, while the other two take into account the owner's requirements of payload weight and payload volume. Leopold and Reuter [25] later established a multiple-term optimization criterion and proposed that the terms cost, effectiveness, flexibility, availability, habitability, vulnerability, and survivability might be used as an objective function. On the other hand, Eames [15], in his CEM, recommended "that a sensible objective for concept exploration is to find the minimum size of ship required to achieve a given payload, speed, and range." He modeled this requirement with the separate or simultaneous maximization of: operational weight ratio, operational volume ratio, and transport effectiveness. Jenkins [16], guided by Manning and Saunders [23], used the displacement as an objective function. **The** minimization of the displacement required to support the **specified** payload items was used as an indicator of relative size and cost.

As may be realized from all of the above, it seems necessary to examine both factors, cost and size, in order to be complete in the specification of an optimum. The main characteristic of military vessels is the satisfaction of the "military requirements.' In contrast to commercial vessels, the military vessel's mission is to deliver a military payload at a specified time and place and to provide services when and where needed, rather than to be profitable. This justifies the choice of the minimum displacement as an objective function.

It is also important that in using minimum displacement as a measure of merit for the military vessel, the highest ratio of military payload to displacement is obtained. Naval combatants of the frigate type are considered to be space-dominated. As discussed previously, a successful naval combatant is characterized by satisfaction of the "military requirements." Since the military payload is a factor related to those requirements, it may be defined in terms of as the operational weight and operational volume. "Operational volume" corresponds to "operational weight," which is defined as the difference between the displacement and the total weight of the primary and secondary hull structure, superstructure, machinery, auxiliary systems and outfit, and fuel. "Operational volume" is defined as the difference between the machinery, personnel, outfit and fuel volume and the total volume. Thus the ratio of the operational volume to total volume would be an indication of the successful distribution of the internal volume. The maximization of this ratio could provide a valid measure of the relative efficiency of the ship volume. This objective function is combined with the minimization of the displacement for a better representation of the size of the ship and of the military mission required percentage of the internal volume.

In addition, the overall life cycle cost of the ship may be the most valuable component for comparison of the different ship concepts and configurations. Minimization of the overall life cycle cost of the ship may result in a far more meaningful search for the optimum design than consideration of only the displacement and volume.

A combination of three individual objective functions has therefore been selected. Minimization of the full load displacement is the first objective function, achieved by the use of the INITIALIZATION module. Maximization of the operational/total volume ratio is the second objective function, achieved by the use of the SPACE module. Minimization of the overall life cycle cost is the final objective function, achieved by the use of the COST module. The selection of these three objective functions correlates with the selection of the three modules, INITIALIZATION, COST and SPACE from the ASSET model.

6.3. Design Constraints

The imposition of the design constraints limits the acceptable design outputs of the COPES/CONMIN optimizer to one design which satisfies the imposed constraints in an optimum fashion.

In order to couple the three ASSET modules with COPES/CONMIN, it was necessary to make several modifications to the modules. All of the following constraints have been introduced mainly in the INITIALIZTION module, as it is responsible for providing a satisfactory initial ship model. The constraints may also be imposed indirectly on the other two modules, COST and SPACE, through the INITIALIZATION module, but were not for this study.

A series of constraints related to the displacement, residuary resistance coefficients, stability requirements, available and required horsepower, usable fuel weight, and propeller calculations was added to the program code. The INITIALIZATION module gives a series of warning messages when extrapolation in the calculation beyond defined limits occurs, unsatisfied minimum requirements exist, or non-convergence of the displacement and the full load weight occurs. The proposed constraints, without limiting the module's ability to give these warning messages during the design process, succeed in establishing an optimum design which satisfies all of the imposed constraints. In other words, the final product of the INITIALIZATION module is a design without warning messages related to the above constraints, and therefore an acceptable one.

The convergence loop in subroutine INITLZ of the INITIALIZATION module was not changed for the wetted surface area, smeared thickness of the primary hull structure, design and range drag, design and range speed delivered horsepower, range and usable fuel weight, or full load weight. This does not limit the optimizer, and in addition gives a series of warning messages when convergence does not occur. Leaving this iterative loop unchanged prevents the need for additional constraints which would have replaced the convergence loops.

The introduced constraints and the corresponding subroutines are:

$$\text{RATIO1} = (0.85/\text{RATIO}) - 1.0 \leq 0$$

$$RATIO2 = (RATIO/1.15) - 1.0 \leq 0, \tag{6.1}$$

The displacement on design waterline is constrained to be within 15 percent of the full load weight based on the existing tolerance in the ASSET program. Should the designer desire to change the tolerance, it is accomplished by changing the limits of the arguments in the input data for the COPES/CONMIN optimizer.

In the subroutine IHYSTA, the added arguments:

$$GMRREQ, = (GMMIN/GM) - 1.0 \leq 0 \text{ and}$$

$$FBRREQ, = (FBDMIN/FBDACT) - 1.0 \leq 0 \tag{6.2}$$

limit minimum intact GM and minimum intact freeboard for intact stability requirements. The above statements mean respectively that:

a) The calculated actual intact GM must be greater than the minimum permissable intact GM.

b) The calculated actual freeboard must be greater than the minimum permissible freeboard.

The requirement that the main engine rated continuous horsepower be greater than the required horsepower was introduced in the subroutine IMACHY with the argument:

$$PHPRAT = (PHPREQ/PHPAVL) - 1.0 \leq 0. \tag{6.3}$$

Similarly, in the same subroutine a statement was added which restricts the secondary engine available continuous horsepower to being less than that required through the argument:

$$SHPRAT = (SHPREQ/SHPAVL) - 1.0 \leq 0. \tag{6.4}$$

The greatest number of constraints was added in subroutine CTRYLR due to the limitations of the Taylor Standard Series power estimation.

The added arguments:

$$BT1 = (2.25/BT) - 1.0 \leq 0$$

$$BT2 = (BT/3.75) - 1.0 \leq 0 \tag{6.5}$$

require the beam to draft ratio to be greater than 2.25 and less than 3.75 in order to use the residual resistance coefficients in the Taylor Standard Series without extrapolation.

Constraints on the limits of the prismatic coefficient introduced by the arguments:

$$CP1 = (0.52/CP) - 1.0 \leq 0$$

$$CP2 = (CP/0.68) - 1.0 \leq 0 \tag{6.6}$$

require that the prismatic coefficient must be greater than 0.52 and less than 0.68 to avoid extrapolation.

Similarly, the volumetric coefficient should be greater than 0.001 and less than 0.002, which is satisfied by the added arguments:

$$CV1 = (0.001/CV) - 1.0 \leq 0 \text{ and}$$

$$Cv2 = (CV/0.002) - 1.0 \leq 0. \tag{6.7}$$

Finally, the speed to length ratio must be positive and less than two, which was introduced by the following arguments:

$$SL1 = SL(-1.0) \leq 0 \text{ and}$$

$$SL2 = (SL/2.0) - 1.0 \leq 0. \tag{6.8}$$

Requirements on the calculations of the open water propeller characteristics were introduced in the subroutine IPLIBR. The necessary arguments for this were:

$$PCHDIA1 = (0.68/PCHDIA) - 1.0 \leq 0 \text{ and}$$

$$PCHDIA2 = (PCHDIA/3.4) - 1.0 \leq 0, \tag{6.9}$$

which require that the pitch/diameter ratio must be greater than 0.68 and less than 3.4.

The remaining constraints appear in the subroutine INITLZ. and require that the usable fuel weight must be positive. The usable fuel weight is determined by subtraction of the weight of lightship and the weight of all full loads (except fuel) from the fixed full load weight. Logically, the fuel weight must be positive. This was achieved by addition of the argument:

$$WTFUEL1 = WTFUEL(-1.0) \leq 0. \tag{6.10}$$

As mentioned previously the above constraints have been added to the INITIALIZATION module. but could have been imposed indirectly on the other two modules. Should any further constraints be desired, the designer need only identify them, input simple FORTRAN arguments in the program code, specify their upper and lower limits, and add the necessary data to the input data of the COPES/CONMIN optimizer.

6.4. GLOBCM Statement

The COPES/CONMIN program requires the COMMON/GLOBCM/ statement, which is a labeled common block that contains all of the objective functions, the design variables. the constraints and all necessary parameters for the optimization process. The /GLOBCM/ statement must appear in each subroutine in which the variable or parameter is used. The optimizer COPES/CONMIN uses this block as a catalog to identify the location of the objective function, the design variables, the constraints, and all of the input data variables, as well as the output result parameters.

It must be emphasized here that the **/GLOBCM/** labeled common block contains all the above mentioned items for all three modules, INITIALIZATION, COST and SPACE. **This** is because of the ASSET/COPES model organization. Table 6.1 shows the contents of the **/GLOBCM/** labeled common block, while Appendix F is a complete list of the elements with the corresponding global location for each.

6.5. Model Organization

Additional modifications to the three ASSET modules **were** required, primarily related to the FORTRAN language rules, but also related to the structure of the ASSET/COPES model. The addition of the **/GLOBCM/** labeled common block, necessary for **coupling** purposes, entailed the removal of the **/CMPL/** labeled common block in all three modules. The **/CMPL/** block represented the model parameter list **(MPL)** of each module of the ASSET program. Now all of the necessary input data are read from an input file instead of being transferred from the data bank through the current model process. To serve this purpose, READ statements have been added to the code of the program. Another list of added arguments was used to initialize all the control parameters (i.e. menu number for printed output, etc.), which were included in the **/CIOCON/** labeled common block; see ASSET manuals. Additional modifications include:

a) changes of the parameter names where conflicts occurred

b) addition of statements to specify different parameters

c) removal of unnecessary arguments.

A brief but complete guide for all the modifications is shown in Table 6.2 for the INITIALIZATION module, Table 6.3 for the COST module, and Table 6.4 for the SPACE module.

As stated previously, it is necessary for the operation of

```
 COMMON/ GLOBCM /
$ BMOMHG,BMOMSG,CA,CGFULL(2),CGI,CREW(3),DDELHP,DHRHO,DHVOLF,
$ DHVOL,DMISSN,DMODEI,DMTRLI,DRAGD,DRAGFC,DRAGR,EFFYRR,EFFYTD,
$ EFFYTR,FLINEI,HCP,HCX,HLBPB,HLBPD,HLBP,HMARGS,HMAXPS,HMTRLI,
$ HSIZEI,HTD,HULRHO,HULVOL,PDIAM,PHPAVL,PHPREQ,PLOADI,PMARGN,
$ PNOENG,PROPNO,PROPSI,PROPTI,PSFC,PSIZEI,PTYPEI,RANGE,RDELHP,
$ SHPAVL,SHPREQ,SMARGN,SMOIM,SNOENG,SSFC,SSIZEI,SSPUI,STYPEI,
$ SYSKWI,SYSKW,TDCOEF,TEFFYI,TTYPEI,VDESGN,VRANGE,WAKEFR,WORM(31),
$ WTADJ(7),WTAGAS,WTAMMO(4),WTARM(5),WTCRGO(5),WTCS(5),WTFUEL,
$ WTFULL,WTMRGN,WTSHIP(8),SURF,THICK1,AX,B,CWP,DISP,T,XKB,HBT,
$ CV,SL,CP,BT,RATIO,FBRREQ,GMRREQ,PHPRAT,SHPRAT,RATIO1,RATIO2,
$ BT1,BT2,CP1,CP2,CV1,CV2,SL1,SL2,WTFUEL1,PCHDIA1,PCHDIA2,ADFACC,
$ AIRVOL,ANOPHR,ANORDC,FUELC,FUELRP,FUELRS,PFFRAC,PLDFSC,PLDLSC,
$ PLDTEC,RATEPD,RDLGTH,SERVLF,SHIPNO,TCHADC,YRDLLR,CLTOT,CLALL,
$ CADJ,CFALL,CFOLLW,CFOUTF,CFTOT,PFOLLW,CLEAD,CLOUTF,PLEAD,CEGY,
$ CISP,CISS,CMTC,COASOC,COPS,CPPE,CPSE,CPTE,CSDD,CSPE,CSSE,CSTE,
$ DCLFPS,DCLIFE,SYSCPS,SYSCST,ARMSPC(4),ACTVOL,SSCS1(18),VSSCS1(18),
$ SSCS2(31),VSSCS2(31),SSCS3(35),VSSCS3(35),TOTARE,TOTVOL,OPTOTVL,
$ FKN(9),RATEIF(15),UOASC,UUCAPY,UUNITC,DFMMHR,DTIOC,RATELN
```

Table 6.1. Common Block GLOBCM of the ASSET/COPES Model

INITIALIZATION MODULE

SUBROUTINE	MODIFICATION
IHULGM	1) Addition of the **/GLOBCM/** labeled common block 2) Addition of the following arguments, for constraint purposes: **HBT=HLBPD/(HTD*HLBPB), RATIO1=(0.85/RATIO)-1.0,** and **RATIO2=(RATIO/1.15)-1.0** 3) Change of the dummy variables of the SUBROUTINE argument, because they are contained in the **/GLOBCM/** common block
IHYSTA	1) Addition of the **/GLOBCM/** labeled common block 2) Addition of the following arguments, for constraint purposes: GMRREQ=(GMMIN/GM)-1.0, and FBRREQ=(FBDMIN/FBDACT)-1.0 3) Change of the dummy variables of the SUBROUTINE statement, because they are contained in the **/GLOBCM/** common block
IMACHY	1) Addition of the **/GLOBCM/** labeled common block 2) Addition of the following arguments, **for** constraint purposes: **PHPRAT=(PHPREQ/PHPAVL)-1.0,** and **SHPRAT=(SHPREQ/SHPAVL)-1.0** 3) **Change** of the dummy variables of the SUBROUTINE statement, because they are contained in the **/GLOBCM/** common block

Table 6.2. INITIALIZATION Module Modifications.

INITIALIZATION MODULE

SUBROUTINE	**MODIFICATION**
ANALIZ (initially **INITLZ**)	1) Removal of the **/CSYNTH/** and **/CMPL/** labeled common blocks 2) Addition of the following labeled common blocks: **/GLOBCM/** for the coupling with the **COPES/CONMIN** optimizer **/CINFR/** with informations of the COST module **/CINFI/** with informations of the SPACE module **/CUNIT/** with specifications of the unit system **/GLOBCN/** with variables from the COST module **/GLOBCL/** with variables from the SPACE module 3) Addition of TF, RETURN, and CONTINUE statements to segment the subroutine ANALIZ in input, execution, and output 4) Addition of DATA statements for the variables of the **/CINFO/** labeled common block 5) Addition of READ statements to read control variables for output, unit-system, selected module, module data for the INITIALIZATION, COST, and SPACE modules 6) Addition of WRITE statements to write the module data of all three modules, for checking of the module input data 7) Addition of IF, and CONTINUE statements to control module Input data and module execution

Table 6.2. Continued. Complete Guide of the Modifications of the Initialization Module of the ASSET/COPES Model

INITIALIZATION MODULE

SUBROUTINE	MODIFICATION
ANALIZ(continued)	8) Addition of the argument **WTFUEL1=WTFUEL*(-1.0),** for constraint purposes 9) Addition of CALL statements for the COST and SPACE modules 10) **Initialization** of control variables of the **/CIOCON/** common block 11) Initialization of all the uninitialized variables of the **/GLOBCM/** labeled common block 12) Reformation of the dummy variables in the CALL statement of the subroutines **IHULGM, IHYSTA, IMACHY,** and **IPROP,** because they are contained in the **/GLOBCM/** labeled common block 13) Change of variable names, because of confliction with the names of the variables of the **COPES/CONMIN** program, **(i.e.ITMAX** to **IITMAX** etc.) 14) Transfer of the OUTPUT GENERATION part from the COST subroutine, as output is controlled by the **ANALIZ** subroutine. The control variable **IONCON** has been substituted by the name **ICONON** to avoid confliction 15) Transfer of the OUTPUT GENERATION part from the SPACE subroutine, as output is controlled by the **ANALIZ** subroutine. The control variable **IONCON** has been substituted by the name IOCNON to avoid confliction

Table 6.2. Continued. Complete Guide of the Modifications of the Initialization Module of the ASSET/COPES Model

INITIALIZATION MODULE

SUBROUTINE	MODIFICATION
INTMPL	1) Removal of the /CMPL/ labeled common block 2) Addition of the /GLOBCM/ labeled common block
IPLIBR	1) Addition of the following statements, for constraint purposes: PCHDIA1=(0.68/PCHDIA)-1.0 and PCHDIA2=(PCHDIA/3.40)-1.0 2) Addition of the /GLOBCM/ labeled common block 3) Change of the dummy variables in the SUBROUTINE statement, because they are contained in the /GLOBCM/ common block 4) Reformation of the dummy variables in the CALL statement of the subroutines LIBGVN, and OWSLOP; (variables are contained in the /GLOBCM/ labeled common block)
LPROP	1) Addition of the /GLOBCM/ labeled common block 2) Change of the dummy variables in the SUBROUTINE statement, because they are contained in the /GLOBCM/ common block 3) Reformation of the dummy variables in the CALL statement of the subroutine IPLIBR; (they are contained in the /GLOBCM/ labeled common block)
IRESIS	1) Change of the dummy variables in the CALL statement of the subroutine CRTYLR; (they are contained in the /GLOBCM/ block)

Table 6.2. Continued. Complete Guide of the Modifications of the Initialization Module of the ASSET/COPES Model

INITIALIZATION MODULE

SUBROUTINE	MODIFICATION
SHIP2	1) Addition of the /GLOBCM/ labeled common block 2) Addition of arguments to correspond the variable values included in the /GLOBCM/ block to the dummy variable values used in this subroutine (i.e. A1=AX, etc.)
LIBGVN	1) Addition of the /GLOBCM/ labeled common block 2) Change of the dummy variables in the SUBROUTINE statement, because they are contained in the /GLOBCM/ common block 3) Change of the variable names A, B, C, AA, BB, CC, AAA, BBB, and CCC to A6, B6, C6, A7, B7, C7, A8, B8, and C8 respectively, because of confliction with variable names of the COPES/CONMIN program
OWSLOP	1) Addition of the /GLOBCM/ labeled common block 2) Change of the dummy variables in the SUBROUTINE statement, because they are contained in the /GLOBCM/ common block

Table 6.2. Continued. Complete Guide of the Modifications of the Initialization Module of the ASSET/COPES Model

INITIALIZATION MODULE

SUBROUTINE	MODIFICATION
CRTYLR	1) Addition of the **/GLOBCM/** labeled common block 2) Change of the dummy variables in the SUBROUTINE statement because they are contained in the **/GLOBCM/** common block 3) Removal of the DIMENSION **WORM(31)** argument, because it is contained in the **/GLOBCM/** labeled common block 4) Addition of the following statements, for constraint purposes: **BT1=(2.25/BT)-1.0, BT2=(BT/3.75)-1.0** **CP1=(0.52/CP)-1.0, CP2=(CP/0.68)-1.0** **CV1=(0.001/CV)-1.0, CV2=(CV/0.002)-1.0** **SL1=SL*(-1.0),** and **SL2=(SL/2.0)-1.0**

Table 6.2. Continued. Complete Guide of the Modifications of the Initialization Module of the ASSET/COPES Model

COST MODULE

SUBROUTINE	MODIFICATION
COST	1) Removal of the /CMPL/ labeled common block 2) Removal of the DIMENSION CMNAME(3), Z(18,6) statement, (arrays are contained in the /GLOBCM/ common block 3) Removal of the OUTPUT GENERATION part, because it has been transferred in the subroutine ANALIZ 4) Addition of the /GLOBCM/ labeled common block 5) Addition of the /GLOBCN/ labeled common block, (contains variables included in COST module other subroutines) 6) Addition of DATA statement to specify the PTITLE1, PVERSN1, and PDATE1 parameters 7) Initialization of the control parameters of the /CICOON/ labeled common block 8) Change of the /CINFO/ and /CIOCON/ names to /CINFR/ and /CICOON/ respectively, and the name of the IONCON parameter to ICONON; because of confliction with the corresponding names of the parameters of the subroutine ANALIZ 9) Reformation of the dummy variables in the CALL statement of the subroutines CTLEAD, CTFLLW, CTLIFE, CLFSUM; because the variables are contained in the /GLOBCM/ common block

Table 6.3. COST Module Modifications.

COST MODULE

SUBROUTINE	MODIFICATION
CSTERR	1) Change of the /CIOCON/ name to /CICOON/, as explained in subroutine COST
CSTMPL	1) Removal of the /CMPL/ labeled common block 2) Addition of the /GLOBCM/ labeled common block
CSTMSC	I) Change of the /CIOCON/ name to /CICOON/, as in subroutine COST
CTLEAD, CTFLLW, CLFSUM	1) Addition of the /GLOBCM/ labeled common block 2) Change of the dummy variables in the SUBROUTINE statement, because they are contained in the /GLOBCM/common block
CTLIFE	1) Addition of the /GLOBCM/ labeled common block 2) Change of the dummy variables in the SUBROUTINE statement, because they are contained in the /GLOBCM/ common block 3) Removal of the DIMENSION CREW(3) statement, because array is contained in the /GLOBCM/ labeled common block

Table 6.3. Continued. Complete Guide of the Modifications of the **COST** Module of the ASSET/COPES Model

SPACE MODULE

SUBROUTINE	**MODIFICATION**
SPACE	1) Removal of the **/CMPL/** labeled common block
	2) Kemoval of the DIMENSION **SSCS1(18),...etc.** statement, (arrays are contained in the **/GLOBCM/** common block
	3) Removal of the OUTPUT GENERATION part, because it has been transferred in the subroutine **ANALIZ**
	4) Addition of the **/GLOBCM/** labeled common block
	5) Addition of the **/GLOBCL/** labeled common block, which contains variables included in SPACE module other **subroutines**
	6) Addition of DATA statement to specify the **PTITLE2, PVERSN2,** and **PDATE2** parameters
	7) Addition of the argument: **OPTOTVL=(VSSCS1(1)/TOTVOL)*1**.0
	8) Initialization of the control parameters of the **/CIOOCN/** block
	9) Change of the names of the **/CINFO/** and **/CIOCON/** common blocks to **/CINFI/** and **/CIOOCN/** respectively, and the parameter **IONCON** to IOCNON, because of confliction with the corresponding names of the subroutine ANALIZ
	10) Reformation of the dummy variables in the CALL statement of the subroutines **SPCS1, SPCS2,** and SPCS3, (variables are contained in the **/GLOBCM/** labeled common block

Table 6.4. SPACE Module Modifications.

SPACE MODULE

SUBROUTINE	MODIFICATION
SPCERR	1) Change of the /CIOCON/ common block name to /CIOOCN/, as explained in subroutine SPACE
SPCMPL	1) Removal of the /CMPL/ labeled common block 2) Addition of the /GLOBCM/ labeled common block
SPCMSG	1) Change of the /CIOCON/ common block name to /CIOOCN/, as explained in subroutine SPACE
SPCS1, SPCS2	1) Addition of the /GLOBCM/ labeled common block 2) Change of the dummy variable list in the SUBROUTINE statement because the variables are contained in the /GLOBCM/ block 3) Removal of the DIMENSION ARMSPC(4), . .etc., and the DIMENSION CREW(3) statements of the SPCS1 and SPCS2 subroutines respectively; (arrays are contained in the /GLOBCM/ labeled common block
SPCS3	1) Addition of the /GLOBCM/ labeled common block 2) Change of the dummy variable list in the SUBROUTINE statement because the variables are contained in the /GLOBCM/ labeled common block

Table 6.4 Continued. **Complete Guide** of the Modifications of the SPACE Module of the **ASSET/COPES Model**

COPES/CONMIN to have the analysis portion, i.e., the three ASSET **modules.** of the program in subroutine form. This subroutine is called ANALIZ, and has the calling parameter **ICALC.** For ICALC = 1, the data for the operation of **COPES/CONMIN** are read in. For ICALC = 2, all of the analysis calculations are performed by the optimizer, and for ICALC = 3 the final results are printed out.

Based on these guidelines, it was decided that the INITIALIZATION module should be named subroutine **ANALIZ,** and that the other two modules, COST and SPACE, would be called as simple subroutines by the **ANALIZ** subroutine, at the user's desire. Provision also was made to segment the subroutine **ANALIZ** into input, execution and output, in accordance with the parameter ICALC. The INITIALIZATION module is the main executed module. If the user wants to execute only the INITIALIZATION module, he may input data for only this. If the user wants to execute the COST or the SPACE modules, he must input data for the COST or SPACE modules in addition to the INITIALIZATION module.

This was necessary for two reasons:

a) The COST and SPACE modules need input data in addition to the results of the INITIALIZATION module, as these two modules are in the analysis portion of the ASSET program.

b) The leading particulars of a ship are not a direct input in either the COST and SPACE modules, and so cannot be defined as design variables for these two modules.

The calling and execution sequence for the ASSET/COPES model is shown schematically in Figure 6.2.

6.6. Input

The program begins by reading the following data:

```
PROGRAM COPES

CALL ANALIZ(ICALC)

END

SUBROUTINE ANALIZ(ICALC)          [INITIALIZATION
                                   MODULE        ]

CALL COST

CALL SPACE

END
SUBROUTINE COST                   [COST
                                   MODULE]

END
SUBROUTINE SPACE                  [SPACE
                                   MODULE]

END
```

Fig. 6.2. Schematical Presentation of the Calling, and Execution Sequence of the ASSET/COPES Model

a) data necessary for control of the **COPES/CONMIN** optimizer, (i.e., control of printed output, objective' function, design variables, constraints, specifications, etc.)

b) variable name to control module execution

c) data to specify unit-system and numbers of menus to be printed in the output, for each module

d) data necessary for each module execution,

All four sets of data must be input to a data file (i.e., DATA5 for this study. The **COPES/CONMIN** optimizer accepts data for two options: a) simple analysis and b) numerical optimization. More details on this are in Ref. [17].

It is restated here that input data for the execution of the INITIALIZATION module are always required, no matter which objective function is to be optimized. Table 6.5 shows a complete set of input data for the execution of the INITIALIZATION module. An extensive reference on the input data, formats, other specifications and examples is included in the ASSET/COPES/GRAPICS User's Guide (optimization mode), Appendix E.

6.7. output

The ASSET/COPES model produces only printed output. The user controls the amount of printed output by specifying the necessary control variables. It should be emphasized here that results related not only to the optimum design may be printed, but also results related to the different stages of design before the optimum is achieved (i.e. neighborhood of the optimum, etc.). This is easily done by specifying the control variable for **COPES/CONMIN** output in the data related to the optimizer (data set A).

```
$ DATA FOR COPES
MINIMIZATION OF FULL LOAD DISPLACEMENT FOR FFG7 SHIP
2,4
3,,,,,,10
0.0
0.0
0,126,-1.0
300.0,700.0
7.0,12.0
0.52,0.68
0.7,0.9
1,28,1.0
2,26,1.0
3,24,1.0
4,25,1.0
2
150,152
-1.0+6,0.0,0.0
154,166
-1.0+6,0.0,0.0
END
$ DATA FOR INITIALIZATION MODULE
INTLZN
 193765432100000000000000000000
GIVEN        FUEL WT     AL 5086     ITTC         MS
GIVEN        GIVEN       GIVEN       CP           GIVEN
GT           GIVEN       DIESEL      NONE         GIVEN
GIVEN        MECH
75542.1      52756.3        0.5E-0130.513         0.6178
13.00        15.00       165.00      165.888      0.416231
30.00        0.0         1.0         0.978700     0.955
0.535600     0.74940     8.906       13.60        380.00
1.12         19.040      0.478333    489.024      16.50
20500.0      1.16916     2.0         1.0          0.435
5574.54            E    +37 0.1E+37    0.1E+3       0.1E+37
4000.0         0   %E   [illegible]  20.0        7  0.76E-01
1.08         1.20        1.20        1.26         1.29
1.27         0.85        1.11        1.01         0.93
0.88                     0.84        0.83         0.84
0.84         0.85        0.85                     0.86
0.87         0.87        0.87        [illegible]  0.87
0.87         0.87        0.87        0.87         0.87
0.87         0.0         0.0         0.0          0.0
0.0          0.0         0.0         60.62        40.21
9.3          21.98       0.0         32.07        35.12
22.3         5.58        1.70        0.0          0.0
0.0          0.0         0.0         33.98        6.65
19.55        19.60       18.71       569.401      0.0
5500.00      250000.0    70000.0     30000.0      6000.0
```

Table 6.5. Input Data for Execution of the Initialization Module of the ASSET/COPES Model.

The amount of printed output for each module is controlled by specifying the variable **IONCON** (data set C); see also ASSET/COPES/GRAPHICS User's Guide in Appendix E. Upon execution of the INITIALIZATION module, only printed output related to this module may be obtained. Upon execution of the COST or SPACE module, printed output related to the INITIALIZATION and each one of the above two modules may be obtained.

A complete set of all the outputs for each module is shown in Table 6.6, and an examole of the menu numbers 1 and 2 after the execution of the INITIALIZATION module (minimization of full load displacement), Fig. 6.3.

INITIALIZATION MODULE

Menu	Description
#1	Summary
#2	Hull Geometry
#3	Hull Structure
#4	Resistance
#5	Propeller
#6	Machinery
67	Weight
#8	Hydrostatics and Seakeeping

Optimization Information

COST MODULE

#1	Summary
#2	Unit Acquisition Costs
#3	Life-Cycle Costs

Optimization Information

SPACE MODULE

#1	Summary
#2	SSCS1 Space Requirements
#3	SSCS2 Space Requirements
#4	SSCS3 Space Requirements

Optimization Information

Table 6.6. INITIALIZATION, COST and SPACE Module Output.

```
              ASSET VERSION 0 - INITIALIZATION MODULE - 83/10/31 15.44.10

   MENU ITEM NO. 1 - SUMMARY

   RAIN ENG TYPE-CT          HULL SIZE-GIVEN          DESIGN MODE-FUEL UT
   MAIN ENG SIZE-GIVEN       HULL LOADS-GIVEN         SHIP CG INPUT-GIVEN
   SEC ENG TYPE-NONE         HULL MTRL TYPE-MS        ELECTRIC SYS KW-GIVEN
   SEC ENG SIZE-GIVEN        DKHS MTRL TYPE-AL 5086   SSPU ENG TYPE-DIESEL
   TRANS TYPE-MECH                                    FRICTION LINE-ITTC
   TRANS EFF-GIVEN
   PROP TYPE-CP
   PROP SIZE-GIVEN

   DESIGN SPEED REQ,KT            29.00     LBP, FT                        389.36
   RANGE SPEED REQ,KT             20.00     BEAM (ON DWL),FT                44.57
   RANGE, NM                    6383.01     DEPTH (MIDSHIP), FT             28.63
   MILITARY PAYLOAD, LTON         327.4     DRAFT (DWL),FT                  13.69
   NO CREW                         193.     HULL VOLUME,FT3               32 9ib.
   MISSION DURATION, DAYS          30.0     TOTAL SHIP VOL,FT3            4553 3P.

   ALLOW PRIM STRESS, KSI         17.92     DESIGN SPEED DRAG,LBF        304835.
   MIDSHIP MOI, FT2-IN2        135349.      RANGE SPEED DRAG, LBF         63940.

   PROPELLER DIA, FT              16.50     LIGHTSHIP WT,LTON             25t7.4
   NO PROP SHAFTS                    1.     WT MARGIN FACTOR               9.000
   DESIGN SPD DEL HP,HP          39481.     USABLE FUEL WT,LTON            *69.6
   RANGE SPD DEL HP,HP            5336.     FULL LOAD WT,LTON             3375.2

   MAIN NO ENG                       2.     NO ENG USED AT RANGE COND         1.
   MAIN CONT HP AVL/ENG, HP      20500.     MAIN PWR MARGIN FAC            1.016
   DSGN COND HP REQ/ENG,HP       20170.     DANCE COND HP REQ/ENG,HP       5587.
   FUEL CONS, NM/LTON            11.206     ELECTRIC SYS KW               4000.0

   MENU ITEM NO. 2 - HULL GEOMETRY

   LBP, FT                       389.36     PRISMATIC COEF                  .520
   BEAM (ON DWL), FT              44.57     MAX SECTION COEF                .900
   DRAFT (DWL), FT                13.69     BLOCK COEF                      .468
   DEPTH (MIDSHIP), FT            28.63     WATERPLANE COEF                 .710
   WETTED SURFACE, FT2          16962.8     DKHS VOLUME FRAC                .416
   DISP ON DWL,LT                3179.5     DKHS VOLUME, FT3             134412.
   MAX SECTION GIRTH, FT          61.62     VOL OF DISP,FT3              111209.
   MAX SECTION AREA, FT2          549.3     HULL VOLUME,FT3              322926.
   NO BULKHEADS                   14.22     TOTAL SHIP VOL, FT3          457336.
   NO LOWER DECKS                  2.58

      OPTIMIZATION RESULTS

   OBJECTIVE FUNCTION
   GLOBAL LOCATION 126      FUNCTION VALUE   .33752E+04

   DESIGN VARIABLES

              D. V.     GLOBAL       LOWER                          UPPER
      ID      NO.       VAR.NO.      BOUND          VALUE           BOUND
      1        1         28       .30000E+03     .38936E+03     .70000E+03
      2        2         26       .70000E+01     .87366E+01     .12000E+02
      3        3         24       .52000E+00     .52000E+00     .68000E+00
      4        4         25       .70000E+00     .90000E+00     .90000E+00

   DESIGN CONSTRAINTS

              GLOBAL       LOWER                        UPPER
      ID      VAR. NO.     BOUND           VALUE        BOUND
      1        150      -.10000E+07    -.80687E+00       0.
      3        151      -.10000E+07    -.27467E-02       0.
      5        152      -.10000E+07    -.16098E-01       0.
      7        154      -.10000E+07    -.17689E-01       0.
      9        155      -.10000E+07    -.28085E+00       0.
     11        156      -.10000E+07    -. 8057E+00       0.
     13        157      -.10000E+07    -.16601E+00       0.
     15        158      -.10000E+07    -.15510E-01       0.
     17        159      -.10000E+07    -.52325E+00       0.
     19        160      -.10000E+07    -. 0000E+00       0.
     21        161      -.10000E+07     .40373E-10       0.
     23        162      -.10000E+07    -.10136E+01       0.
     25        163      -.10000E+07    -.49321E+00       0.
     27        164      -.10000E+07    -.56960E+03       0.
     29        165      -.10000E+07    -.44453E+00       0.
     31        166      -.10000E+07    -.63994E+00       0.
```

Fig. 6.3. Printed Output of Menus 1 and 2, and Achieved Optimum for Minimization of Full Load Displacement

7.0. ASSET/GRAPHICS ENSEMBLE

7.1. Model Organization

In this section, the necessary modifications to the INITIALIZATION, COST and SPACE modules in order to form the ASSET/GRAPHICS subprogram program will be presented. The ASSET/GRAPHICS subprogram is responsible for all the graphical output which may be produced by the new synthesis system.

The ASSET/GRAPHICS effort was concerned with the development of a program which could produce 2-D and 3-D plots of the objective function as a function of the design variables for each one of the three modules. As has been discussed in the previous section, the selected objective functions and the design variables for each module were:

Objective Functions

1. full load displacement
2. life cycle cost per ship
3. ratio of operational volume to total volume

Design Variables

1. length between perpendiculars
2. LBP to beam ratio
3. prismatic coefficient
4. midship section coefficient.

The main objective of this model is the visualization of the design space using the the same design variables, objective functions and constraints as are used in the optimization model. Thus, the optimum can be perceived and located more clearly and specifically.

It was also the objective of this study to develop a graphics model which could be easily modified by the addition of more design variables or the selection of new objective functions. In this case, the user will be required to change or add some arguments and to'read the additional parameters.

The required program units for the ASSET/GRAPHICS model were the INITIALIZATION, COST and SPACE modules, plus the PICTURE and SIMPLOT graphics routines. In addition to these, a main program of simple FORTRAN arguments was written which transfers the control to the required module. Also, a series of subroutines was added to create the graphical output. Each one of these subroutines is responsible for generating a 3-D perspective and two 2-D projections for each set of one objective function and two design variables.

The main program was named **PICTUR** and the subroutines **PICT01, PICT02....PICT18.** Each of the subroutines then calls one of the modules for execution and creation of the graphical output. Subroutines **PICT01,** through PICT06 call the INITIALIZATION module. The output represents the design space when the objective function is the full load displacement.

Subroutines PICT07 to **PICT12** call the INITIALIZATION and COST modules. The output represents the design space when the objective function is the life cycle cost per ship. The calling of the INITIALIZATION module before the COST module was necessary for the following two reasons: (a) the COST module belongs to the analysis part of the ASSET program, and thus needs input data from the results of the INITIALIZATION module (i.e., total weight), and (b) the four design variables are not input directly to the COST module.

This calling process sequence was preferable to the process of first executing the INITIALIZATION module, then inputting the results to the COST module, and finally creating a graphical output after the design variables had been input again.

Subroutines PICT13 through PICT18 call the INITIALIZATION' and SPACE modules. The outputs represent the design space when the objective function is the ratio of operational volume to total volume. The reasons for calling the INITIALIZATION before the SPACE module are the same as those mentioned above for the COST module. A program control organization is shown in Figure 7.1.

7.2. ASSET Modifications

Additional modifications of the three modules were required in order to execute each one of the modules. These modifications were due primarily to the FORTRAN language rules, but others of primary importance are related to the structure of the **ASSET/GRAPHICS** model.

7.2.1. INITIALIZATION Module

One important change which was necessary for the graphics model was the addition of the labeled common block **/GLOBCM/**. In this common block are listed all the parameters, objective functions and design variables of all three modules. This was required for the graphics creation by each module and the control transfer from/to the main program. This common block was added to each subroutine in which the parameter was used. The contents of the **/GLOBCM/** labeled common block are shown in Table 7.1. The addition of the **/GLOBCM/** common block entailed the removal of the existing **/CMPL/** labeled common block, which represented the model parameter list (MPL), in all three modules. All necessary input data, contained in the GLOBCM common block, are now read from an input file instead of being transferred from the data bank through the current model process.

Another addition to the INITIALIZATION module was a list of

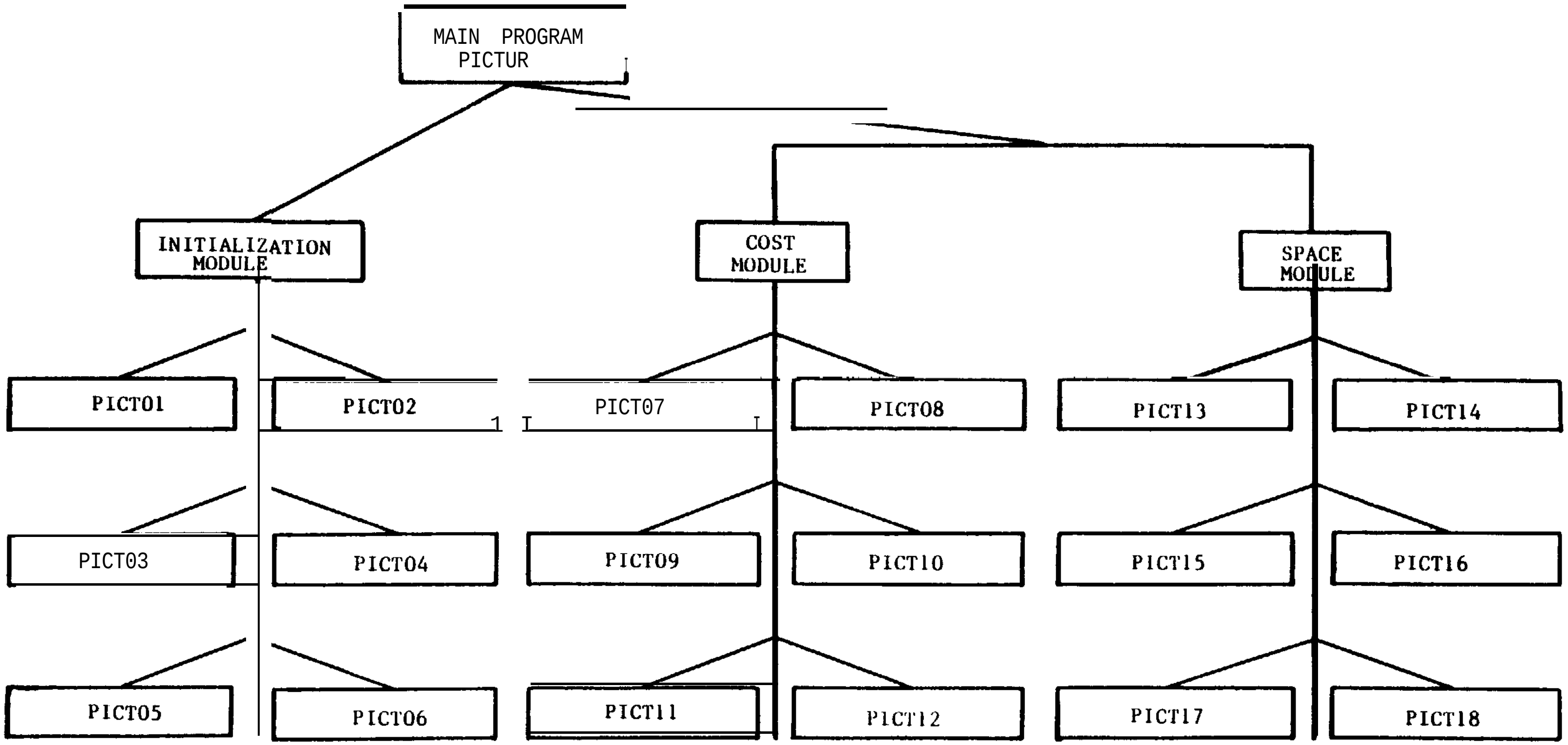


Fig. 7.1. Program Control Organization of the ASSET/GRAPHICS Model.

```
  COMMON/  GLOBCM  /
$ BMOMHG,BMOMSG,CA,CGFULL(2),CGI,CREW(3),DDELHP,DHRHO,DHVOLF,
$ DHVOL,DMISSN,DMODEI,DMTRLI,DRAGD,DRAGFC,DRAGR,EFFYRR,EFFYTD,
$ EFFYTR,FLINEI,HCP,HCX,HLBPB,HLBPD,HLBP,HMARGS,HMAXPS,HMTRLI,
$ HSIZEI,HTD,HULRHO,HULVOL,PDIAM,PHPAVL,PHPREQ,PLOADI,PMARGN,
$ PNOENG,PROPNO,PROPSI,PROPTI,PSFC,PSIZEI,PTYPEI,RANGE,RDELHP,
$ SHPAVL,SHPREQ,SMARGN,SMOIM,SNOENG,SSFC,SSIZEI,SSPUI,STYPEI,
$ SYSKWI,SYSKW,TDCOEF,TEFFYI,TTYPEI,VDESGN,VRANGE,WAKEFR,WORM(31),
$ WTADJ(7),WTAGAS,WTAMMO(4),WTARM(5),WTCRGO(5),WTCS(5),WTFUEL,
$ WTFULL,WTMRGN,WTSHIP(8),SURF,THICK1,AX,B,CWP,DISP,T,XKB,ADFACC,
$ AIRVOL,ANOPHR,ANORDC,DFMMHR,FKN(9),FUELC,FUELRP,FUELRS,PFFRAC,
$ PLDFSC,PLDLSC,PLDTEC,RATEIF(15),RATELN,RATEPD,RDLGTH,SERVLF,
$ SHIPNO,TCHADC,UOASC,UUCAPY,UUNITC,YRDLLR,CLTOT,CLALL,DCLFPS,
$ DCLIFE,CADJ,CFALL,CFOLLW,CFOUTF,SYSCPS,CFTOT,PFOLLW,CLOUTF,
$ CPPE,CPSE,CLEAD,SYSCST,PLEAD,CEGY,CISP,CISS,CMTC,COASOC,COPS,
$ CPTE,CSDD,ARMSPC(4),ACTVOL,SSCS1(18),VSSCS1(18),SSCS2(31),
$ VSSCS2(31),SSCS3(35),VSSCS3(35),TOTARE,TOTVOL,OPTOTVL,CSPE,
$ CSSE,CSTE
```

Table 7.1. Common Block GLOBCM of the ASSET/GRAPHICS Model

arguments in order to initialize the parameters which were included in the **/CIOCON/** labeled common block. These parameters are responsible for the baud rate of the terminal, graphical outputs, maximum iteration number, etc. (For more information see ASSET manuals.) Additional modifications in the program include:

a) changes in the parameter names where conflicts occurred

b) addition of DATA statements to specify different parameters

c) removal of unnecessary arguments and common blocks (i.e. **/CSYNTE/** labeled common block)

d) addition of the READ and WRITE statements to first read the input data and then write them as they had been input during the output stage for checking purposes.

Table 7.2 includes in brief all the changes in the INITIALIZATION module in each of its subroutines.

7.2.2. COST Module

The changes made in the COST module are of the same type as in the INITIALIZATION module. For a description of them, see Table 7.3.

7.2.3. SPACE Module

The changes made in the SPACE module are analogous to those made in the other two modules. Table 7.4 is a brief but complete list of these modifications. There is only one argument which is important for the graphics creation. This is the statement which specifies the ratio of the operational volume to the total volume, i.e., the objective function of the SPACE module. The following statement was added to the SPACE subroutine of the SPACE module:

```
OPTOVL  =  (VSSCS1(1)/TOTVOL)*1.0
```
(7.1)

INITIALIZATION MODULE

SUBROUTINE	MODIFICATION
INITLZ	1) Removal of the **/CSYNTH/** labeled common block
	2) Removal of the **/CMPL/** labeled common block
	3) Addition of the **/GLOBCM/** labeled common block
	4) Addition of READ and WRITE statements to read and write the module Input data, and to specify control parameter **IONCON**
	5) Addition of DATA statement to specify the PTITLE, PVERSN, and PDATE parameters
	6) Initialization of all the control parameters Included in the **/CIOCON/** labeled common block
INTHPL	1) Removal of the **/CMPL/** labeled common block
	2) Addition of the **/GLOBCM/** labeled common block
SHIP2	1) Addition of the **/GLOBCM/** labeled common block
	2) Addition of arguments necessary to correspond the values of the variables included in the **/GLOBCM/** common block to the values of the dummy variables used in this subroutine (i.e. **A1=AX,etc.)**

Table 7.2. INITIALIZATION Module Modifications for the ASSET/GRAPHICS Model.

COST MODULE

SUBROUTINE	MODIFICATION
COST	1) Removal of the /CMPL/ labeled common block 2) Addition of the /GLOBCM/ labeled common block 3) Addition of DATA statement to specify the PTITLE, PVERSN, and PDATE parameters 4) Addition of READ and WRITE statements to read and write module input data, and to specify control parameter ICONON 5) Change of the /CINFO/ and /CIOCON/ names to /CINFR/ and /CICOON/ respectively, because of confliction with the names of the same common blocks of the INITIALIZATION module 6) Initialization of all the control variables included in the /CICOON/ labeled common block 7) Reformation of the dummy variables in the CALL statement of the subroutines CTLEAD, CTFLLW, CTLIFE, and CLFSUM; (dummy variables are contained in the /GLOBCM/ common block
CSTERR	1) Change of the /CIOCON/ common block name to /CICOON/
CSTMPL	1) Removal of the /CMPL/ labeled common block 2) Addition of the /GLOBCM/ labeled common block
CSTMSG	1) Change of the /CIOCON/ common block name to /CICOON/
CTFLLW	1) Addition of the /GLOBCM/ labeled common block 2) Change of the dummy variables of the SUBROUTINE argument, (dummy variables are contained in the /GLOBCM/ common block
CTLEAD	1) Addition of the /GLOBCM/ labeled common block 2) Change of the SUBROUTINE statement, as in CTFLLW subroutine
CTLIFE	1) Addition of the /GLOBCM/ labeled common block 2) Change of the SUBROUTINE statement, as in CTFLLW subroutine
CLFSUM	1) Addition of the /GLOBCM/ labeled common block 2) Change of the SUBROUTINE statement, as in CTFLLW subroutine

Table 7.3. COST Module Modifications for the ASSET/GRAPHICS Model.

SPACE MODULE

SUBROUTINE	MODIFICATION
SPACE	1) Removal of the /CMPL/ labeled common block
	2) Addition of the /GLOBCM/ labeled common block
	3) Addition of DATA statement to specify the PTITLE, PVERSN, and PDATE parameters
	4) Addition of REAL) and WRITE statement to read and write module input data, and output control parameter INOCON
	5) Addition of the argument: OPTOTVL=(VSSCS(1)/TOTVOL)*1.0
	6) Change of the /CINFO/ and /CIOCON/ names to /CINFI/ and /CIOOCN/ respectively, because of confliction with the names of the same common blocks in the INITIALIZATION, and COST modules
	7) Change in the DIMENSION statement because the variables are contained in the /GLOBCM/ labeled common block
	8) Reformation of the dummy variables in the CALL statement of the subroutines SPCS1, SPCS2, SPCS3; (dummy variables are contained in the /GLOBCM/ labeled common block)
	9) Initialization of all the control parameters included in the /CIOOCN/ labeled common block
SPCERR	1) Change of the /CIOCON/ name to /CIOOCN/ common block name
SPCMPL	1) Removal of the /CMPL/ labeled common block
	2) Addition of the /GLOBCM/ labeled common block
SPCMSC	1) Change of the /CIOCON/ name to /CIOOCN/ common block name
SPCS1	1) Addition of the /GLOBCM/ labeled common block
	2) Removal of DIMENSION statement, (variables are contained in the /GLOBCM/ labeled common block)
	3) Change of the dummy variables in the SUBROUTINE argument, (variables are contained in the /GLOBCM/ common block)
SPCS2	1) Addition of the /GLOBCM/ labeled common block
	2) Removal of DIMENSION statement, as in SPCS1 subroutine
	3) Change of the SUBROUTINE statement, as in SPCS1 subroutine
SPCS3	1) Addition of the /GLOBCM/ labeled common block
	2) Removal of DIMENSION statement, as in SPCS1 subroutine
	3) Change of the SUBROUTINE statement, as in SPCS1 subroutine

Table 7.4. SPACE Module for the ASSET/GRAPHICS Model.

7.3 Input

The program begins by reading the following data:

a) variable name to control module execution

b) variable names to control graphical representation of the design variables

c) values of the design variables to create the graphical grid

d) values of the remaining variables after the selection of design variables

e) numbers of menus to be printed in the output

f) data to specify unit-system

g) data necessary for each module execution.

The first four sets of data are read by the **PICTUR** program and the subroutines **PICT01** to **PICT18**, depending upon the case. The remaining sets of data are read by each module separately, also depending on the user's selection. The first four sets of data must be put in a data file (i.e. TAPE8 for the example in this study), and the remaining three sets of data in another data file (i.e. DATA5 for the example in this study). This was necessary because the model calls and executes the selected module as many times as required to create the graphical grid without changing the values of the main portion of the module data (data sets E to G), but the values of the design variables. Input data for the execution of the INITIALIZATION module are always required, no matter which graphics will be produced because of the model structure. Table 7.5 shows the necessary module data (data sets E to G) for production of each graphical output. A simple example of a complete set of input data is shown in Table 7.6 for the execution of the INITIALIZATION module. The input data, examples, formats and other specifications are included in the User's

- GRAPHICAL AND PRINTED OUTPUT FOR INITIALIZATION MODULE
  - Data for INITIALIZATION Module
    1. Input data for unit-system
    2. Input data for printed output menu numbers
    3. Input data for module specifications
- GRAPHICAL AND PRINTED OUTPUT FOR COST MODULE
  - Data for INITIALIZATION Module
    1. Input data for unit-system
    2. Input data for printed output menu numbers
    3. Input data for module specifications
  - Data for COST Module
    1. Input data for unit-system
    2. Input data for printed output menu numbers
    3. Input data for module specifications
- GRAPHICAL AND PRINTED OUTPUT FOR SPACE MODULE
  - Data for INITIALIZATION Module
    1. Input data for unit-system
    2. Input data for printed output menu numbers
    3. Input data for module specifications
  - Data for SPACE Module
    1. Input data for unit-system
    2. Input data for printed output menu numbres
    3. Input data for module specifications

Table 7.5. Input Data to Produce the Graphical and Printed Output for each Module of the ASSET/GRAPHICS Model.

```
$DATA SET A AND B
INTLZN              LBP                 LBPB
$ DATA SET C
300.0               400.0               500.0               600.0               700.0
7.0                 8.0                 9.0                 10.0                11.0                12.0
$ DATA SET D
0.52                0.70
$ DATA SET E AND F
  1222222 2210
$DATA SET G
GIVEN               FUEL WT             AL 5086             ITTC                MS
GIVEN               GIVEN               GIVEN               CP                  GIVEN
GT                  GIVEN               DIESEL              NONE                GIVEN
GIVEN               MECH
75642.1             52756.3                  0.5E-030.513                       0.6178
13.00               15.00               165.00              165.888             0.416231
30.00               0.0                 1.0                 0.978700            0.955
13.60               1.120               19.040              0.478333            489.024
16.50               20500.0             1.16916             2.0                 1.0
0.435               5577.54                  0.1E+37             0.1E+37               0.1E+37
      0.1E+374000.0                        0.76E-0129.0                         2  0 0
    0.76E-011.08                        1.12   --           1.20                1.26
1.29                1.27                1.20                1.11                1.01
0.93                0.88                0.85                0.84                0.83
0.84                0.84                0.85                0.85                0.86
0.86                0.87                0.87                0.87                0.87
0.87                0.87                0.87                0.87                0.87
0.87                0.87                0.0                 0.0                 0.0
0.0                 0.0                 0.0                 0.0                 60.62
40.21               9.3                 21.98               0.0                 32.07
35.12               22.30               5.58                1.70                0.0
0.0                 0.0                 0.0                 0.0                 33.98
6.65                19.55               19.60               18.71               569.601
0.0                 4000.00
```

Table **7.6.** Input Data to Execute the INITIALIZATION Module of the ASSET/GRAPHICS Model.

Guide (graphics mode), Appendix E.

Input data for the creation of the 3-D plots by PICTURE execution are included in the subroutines **PICT01** to PICT18 and thus are fixed for any case. More information on this may be found in the previous section or Appendix B. Input data (parameter specifications, scales, etc.) for the 2-D plots by SIMPLOT execution are included in the SIMPLOT data file (i.e. SIMPDAT for the example in this study) and may be seen in the previous section or in Appendix C.

7.4. output

The ASSET/GRAPHICS model produces two kinds of output~~s~~, printed and graphical. Even though graphical output is the main purpose of the model, it was necessary to add a printed output with the leading particulars and other information for each point (ship) of the design grid. The user controls the amount of printed output by specifying the number of the menu which he wants to print. This is easily done by specification of the variable, i.e. **IONCON** for the INITIALIZATION module; see ASSET/COPES/GRAPHICS User's Guide in Appendix **E.**

Upon execution of the INITIALIZATION module, only printed and graphical output related to this module may be obtained. Upon execution of the COST module, printed output related to the INITIALIZATION and COST modules may be obtained for the same leading particulars, but graphical output may be obtained related only to the COST module. Upon execution of the SPACE module, printed output related to the INITIALIZATION and SPACE modules may be obtained for the same leading particulars, but graphical output may be obtained related to the SPACE module.

The user controls all of the types of output. A complete set of all the outputs for each module is shown in Table 7.7, with all the

INITIALIZATION MODULE

PRINTED OUTPUT

MENU		DESCRIPTION
#1	-	Summary
#2	-	Hull Geometry
#3	-	Hull Structure
#4	-	Resistance
#5	-	Propeller
#6	-	Machinery
#7	-	Weight
#8	-	Hydrostatics and Seakeeping

GRAPHICAL OUTPUT

PLOT NUMBER		DESCRIPTION
#1		WTFULL = f(LBP,LBP/B)
#2		WTFULL = f(LBP,CP)
#3		WTFULL = f(LBP,CX)
#4		WTFULL = f(CP,LBP/B)
#5		WTFULL = f(CX,LBP/B)
#6	-	WTFULL = f(CP,CX)

Table 7.7. Printed and Graphical Output for each Module of the ASSET/GRAPHICS Model.

COST MODULE

PRINTED OUTPUT

MENU	DESCRIPTION
#1	summary
#2	Unit Acquisition Costs
83	Life-Cycle costs

GRAPHICAL OUTPUT

PLOT NUMBER	DESCRIPTION
#1	SYSCPS = f(LBP,LBP/B)
#2	SYSCPS = f(LBP,CP)
#3	SYSCPS = f(LBP,CX)
#4	SYSCPS = f(CP,LBP/B)
#5	SYSCPS = f(CX,LBP/B)
#6	SYSCPS = f(CP,CX)

SPACE MODULE

PRINTED OUTPUT

MENU	DESCRIPTION
#1	Summary
#2	SSCS1 Space Requirements
#3	SSCS2 Space Requirements
#4	SSCS3 Space Requirements

GRAPHICAL OUTPUT

PLOT NUMBER	DESCRIPTION
#1	VPAYL/TOTVOL = f(LBP,LBP/B)
12	VPAYL/TOTVOL = f(LBP,CP)
#3	VPAYL/TOTVOL = f(LBP,CX)
#4	VPAYL/TOTVOL = f(CP,LBP/B)
#5	VPAYL/TOTVOL = f(CX,LBP/B)
#6	VPAYL/TOTVOL = f(CP,CX)

Table 7.7. Continued. Printed and Graphical Output for Each Module of the ASSET/GRAPHICS Model

menu items included. An example of graphical output and the related printed output, menu numbers 1 and 2, from the execution of the INITIALIZATION module may be seen in Figures 7.2 and 7.3.

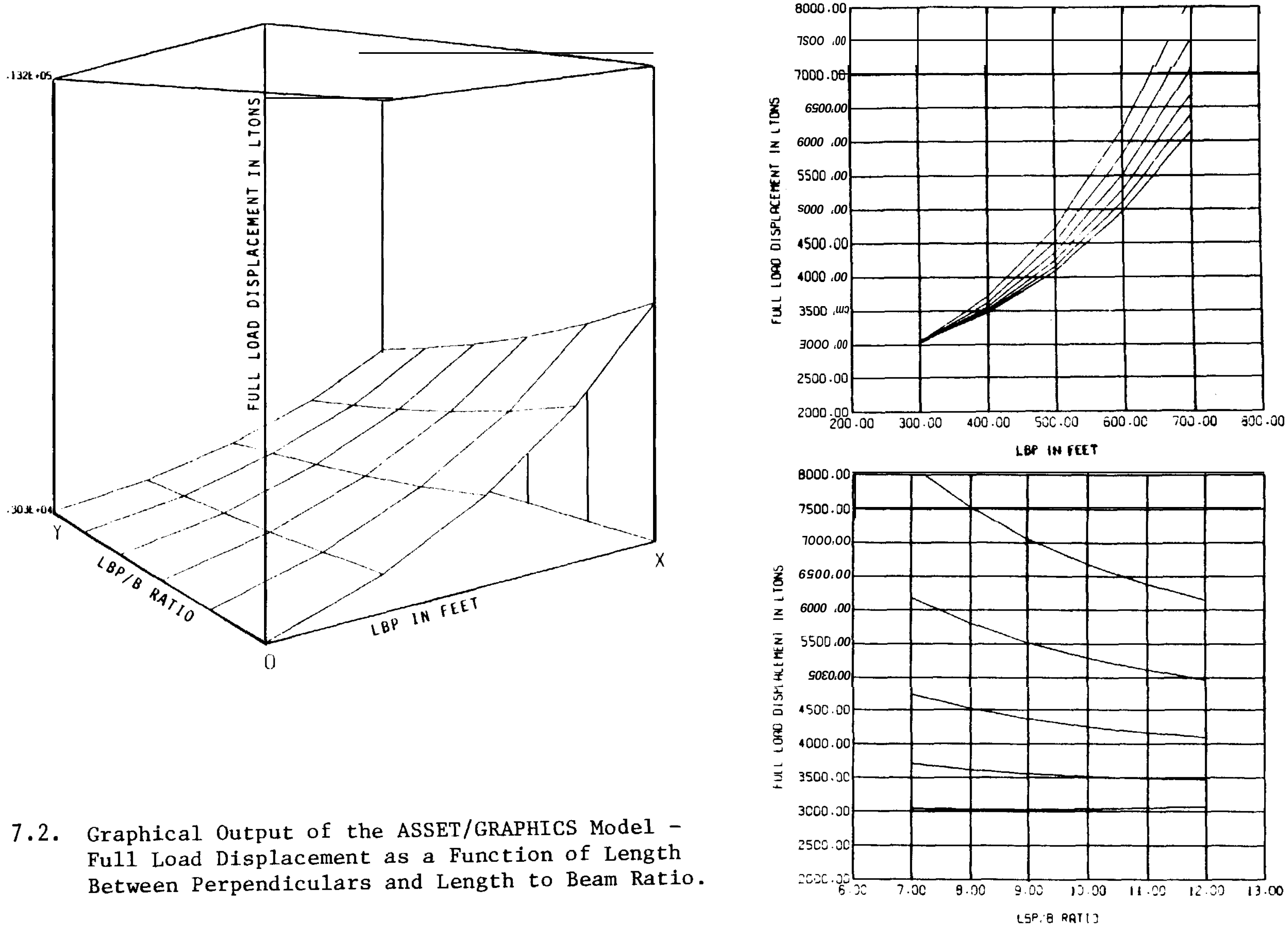


Fig. 7.2. Graphical Output of the ASSET/GRAPHICS Model - Full Load Displacement as a Function of Length Between Perpendiculars and Length to Beam Ratio.

```
      ASSET VERSION 0 - INITIALIZATION MODULE - 83/10/31 15.44.10
MENU ITEM NO. 1 - SUMMARY

MAIN ENG TYPE-GT           HULL SIZE-GIVEN            DESIGN MODE-FUEL WT
MAIN ENG SIZE-GIVEN        HULL LOADS-GIVEN           SHIP CG INPUT-GIVEN
SEC ENG TYPE-NONE          HULL MTRL TYPE-HS          ELECTRIC SYS KW-GIVEN
SEC ENG SIZE-GIVEN         DKHS MTRL TYPE-AL 5086 SSPU ENG TYPE-DIESEL
TRANS TYPE-MECH                                       FRICTION LINE-ITTC
TRANS EFF-GIVEN
PROP TYPE-CP
PROP SIZE-GIVEN

DESIGN SPEED REQ, KT            29.00     LBP, FT                        389.36
RANGE SPEED REQ, KT             20.00     BEAM (ON DWL), FT               44.57
RANGE, NM                     6383.01     DEPTH (MIDSHIP), FT             28.63
MILITARY PAY LOAD, LTON         327.4     DRAFT (DWL), FT                 13.69
NO CREW                          193.     HULL VOLUME, FT3              322926.
MISSION DURATION, DAYS           30.0     TOTAL SHIP VOL, FT3           457330.

ALLOW PRIM STRESS, KSI          17.92     DESIGN SPEED DRAG, LBF        304835.
MIDSHIP MOI, FT2-IN2          135349.     RANGE SPEED DRAG, LBF          63950.

PROPELLER DIA, FT               16.50     LIGHTSHIP WT, LTON             2567.4
NO PROP SHAFTS                     1.     WT MARGIN FACTOR                0.000
DESIGN SPD DEL HP, HP          39481.     USABLE FUEL WT, LTON            569.6
RANGE SPD DEL HP, HP            5336.     FULL LOAD WT, LTON             3375.2

MAIN NO ENG                               NO ENG USED AT RANGE COND          1.
MAX CONT HP AVL/ENG, HP        20500.     MAX PWR MARGIN FAC              1.016
DSGN COND HP REQ/ENG, HP       20170.     RANGE COND HP REQ/ENG, HP       5587.
FUEL CONS, NM/LTON             11.206     ELECTRIC SYS KW                4000.0


MENU ITEM NO. 2 - HULL GEOMETRY

LBP, FT                        389.36     PRISMATIC COEF                   .520
BEAM (ON DWL), FT               44.57     MAX SECTION COEF                 .900
DRAFT (DWL), FT                 13.69     BLOCK COFF                       .468
DEPTH (MIDSHIP), FT             28.63     WATERPLANE COEF                  .710
WETTED SURFACE, FT2           16962.8     DKHS VOLUME FRAC                 .416
DISP ON DWL, LTON              3179.5     DKHS VOLUME, FT3              134412.
MAX SECTION GIRTH, FT           61.62     VOL OF DISP, FT3              111209.
MAX SECTION AREA, FT2           549.3     HULL VOLUME, FT3              322926.
NO BULKHEADS                    14.22     TOTAL SHIP VOL, FT3           457338.
NO LOWER DECKS                   2.58
```

Fig. 7.3. Printed Output of Menus #1 and 2 of the INITIALIZATION Module of the ASSET/GRAPHICS Model.

8.0. ASSET/COPES/GRAPHICS ENSEMBLE

8.1. Model Organization

The final phase of this study was the development of the ASSET/COPES/GRAPHICS ensemble synthesis program. The new synthesis model comprises the following program units:

1. INITIALIZATION Module
2. COST Module
3. SPACE Module
4. COPES/CONMIN Optimizer
5. PICTURE Graphics Program
6. SIMPLOT Graphics Program.

To reduce the amount of storage required and to make efficient use of field length. the synthesis program was divided into overlays. The synthesis model overlay structure is shown in Figure 8.1. All of the graphics possibilities for each module are also shown. The main overlay (0,0) is loaded first and remains in core at all times. Consequently all the utility and control subroutines of both main subprograms are included in the main overlay.

The first primary overlay (1,0) is related to the optimization programs and includes the modified INITIALIZATION. COST and SPACE modules of the ASSET program. The second primary overlay (2,0) is related to the graphics and includes the necessary routines to control the graphics, and the INITIALIZATION. COST and SPACE modules necessary to create the required graphical output. The subroutines for the creation of the graphics are included in the graphics library, which is attached to the program.

The new synthesis model is a versatile tool which can:

a) optimize any function which is included in the GLOBCM

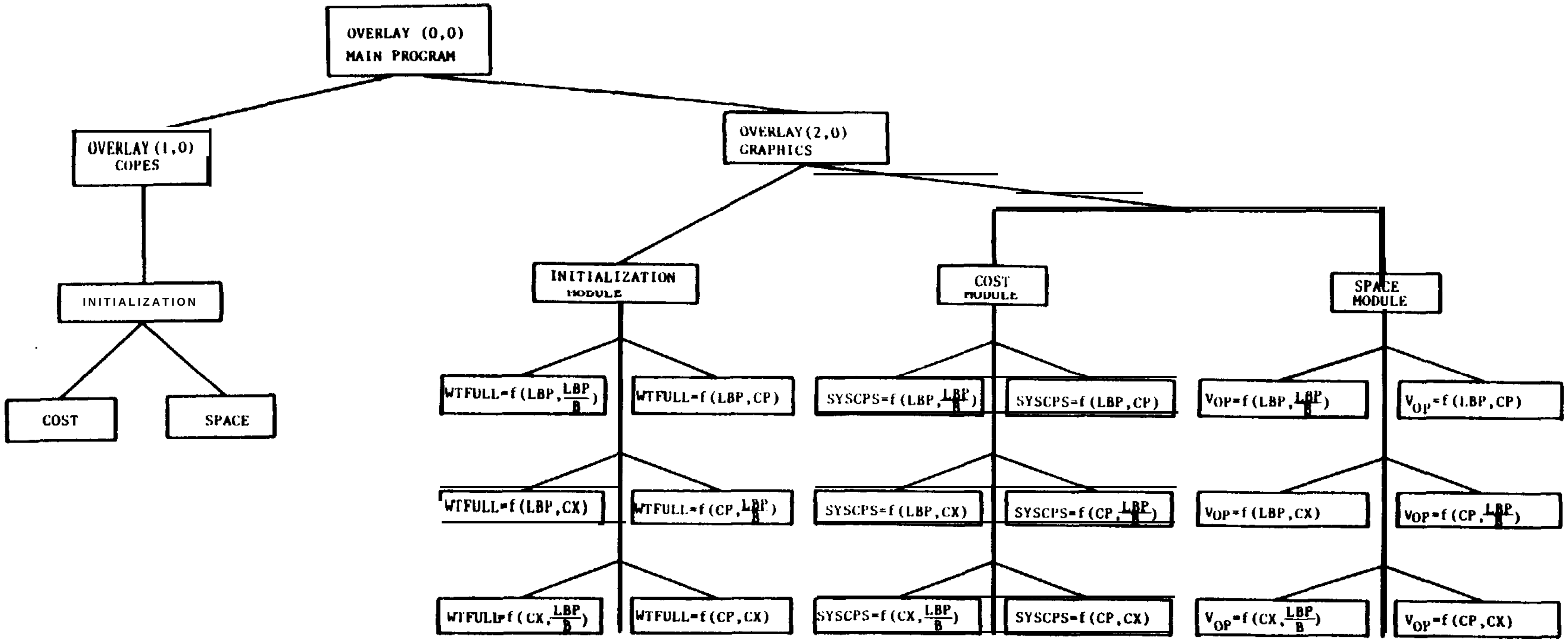


Fig. 8.1. Overlay Structure of the ASSET/COPES/GRAPHICS Synthesis System

common block for each of the INITIALIZATION. COST and SPACE modules

b) control the amount of printed output (for any module and for any menu number)

c) provide 2-D and 3-D graphical output for each one of the INITIALIZATION. COST and SPACE modules. including printed output related to the graphical grid

d) enable the user to control the printed and graphical output of any module for both tasks through an easy process of inputting values to the control parameters of the program.

The proposed process for using the new synthesis model is as follows. Upon decision of a design space. a 3-D graphic protrayal is generated and then a selection of design variables is offered. The optimization mode is then run to determine the optimum which satisfies the imposed constraints. The design space obtained in graphic format is the general design space without the constraints imposed. It is the designer's decision as to which constraints the design must satisfy in order to limit the design space and to obtain an optimum design. A presentation of the above procedure is shown in Figure 8.2.

The design space represents a number of ships which are obtained through a systematic variation of design variables. i.e. length between perpendiculars and prismatic coefficient. The design space may represent:

a) full load displacement - INITIALIZATION module

b) total cost per ship for 30 years of service life - COST module

c) operational to total volume ratio - SPACE module.

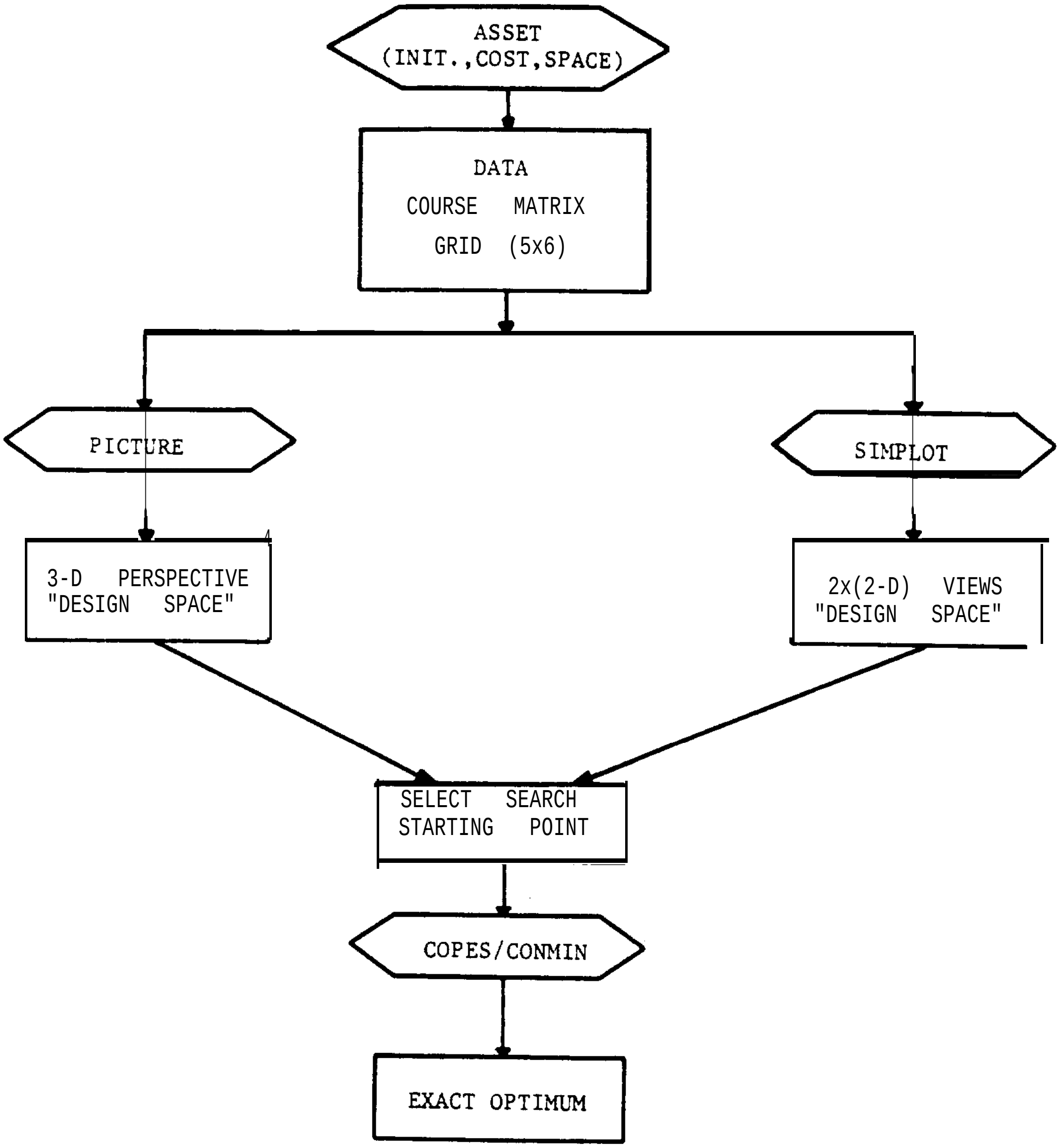


Fig. 8.2. Flow Diagram for the Operation of the ASSET/COPES/GRAPHICS Synthesis System.

The optimum ship in the design space is the one which satisfies all the requirements imposed by the design constraints.

The graphical grid represents the complete design space for specific values of input data. Possible combinations of plotting for each of the modules are:

a. Full Load Displacement = f(LBP. LBP/B, CP. CX)

b. Total Cost/Per Ship = f(LBP. LBP/B, CP, CX)

c. Operational Volume/Total Volume = f(LBP, LBP/B. CP. CX)

The optimization mode then searches for an optimum (minimum or maximum) in the above design space. Any combination of the four design variables. two at a time. can be displayed as 3-D graphical output accompanied by 2 x 2-D plots which represent the projection of the 3-D plot on the two vertical planes.

The overlay structure was preferred for the new synthesis model organization. A schematic illustration of the program structure is shown in Figure 8.1. The main overlay (0,0) controls the entire program. Upon input of the desired mode. the control is transferred to either of the two tasks the program runs, optimization or graphics. The data files and the data tapes are decoded in the main overlay. The control is then transferred to the optimization mode which includes the main program of COPES/CONMIN and the subroutine ANALIZ.

8.2. Input

The data are divided into two types: data necessary to control execution mode, e.g.. output menu number, etc.- and the module-data necessary to run each module. i.e.. ship particulars. machinery, etc. For each execution mode. the data are contained in two data files. Data file TAPE8 was chosen for the control data for both modes. and the data file **DATA5** for the module data of both modes. Another data file is required. as discussed in Section 7, for the data necessary to control and specify the 2-D SIMPLOT graphical **output**. A simple order of arguments. which is included in data file SIMPDAT, is required. For more information, see the SIMPLOT User's Guide. Appendix C.

The user may easily change the values of data files TAPE8 and DATA5 by simply changing the argument of INPUT-OUTPUT at the Program statement of the main overlay **(0,0)**. To execute the OPTIMIZATION mode. the program requires:

a) data to specify this mode (data file: TAPE8)

b) data to specify parameters of the COPES/CONMIN optimizer. desired module execution. output menu number. unit-system, and module-data (data file: DATA5).

To execute the GRAPHICS mode. the program requires:

a) data to specify this mode. the desired module. the design variables for the graphical output and their values (data file: TAPE8)

b) data to specify unit-system. output menu number and module-data for the selected module (data file: **DATA5**)

c) data to control and specify necessary parameters of the output for the two 2-D plots (data file SIMPDAT) .

Because the INITIALIZATION module is the main unit-module, data for this module are required by default. The data chart organization is shown in Figure 8.3. For a complete and detailed reference on input data. see the ASSET/COPES/GRAPICS User's Guide in Appendix E.

8.3. output

Two types of output may be obtained through the new synthesis

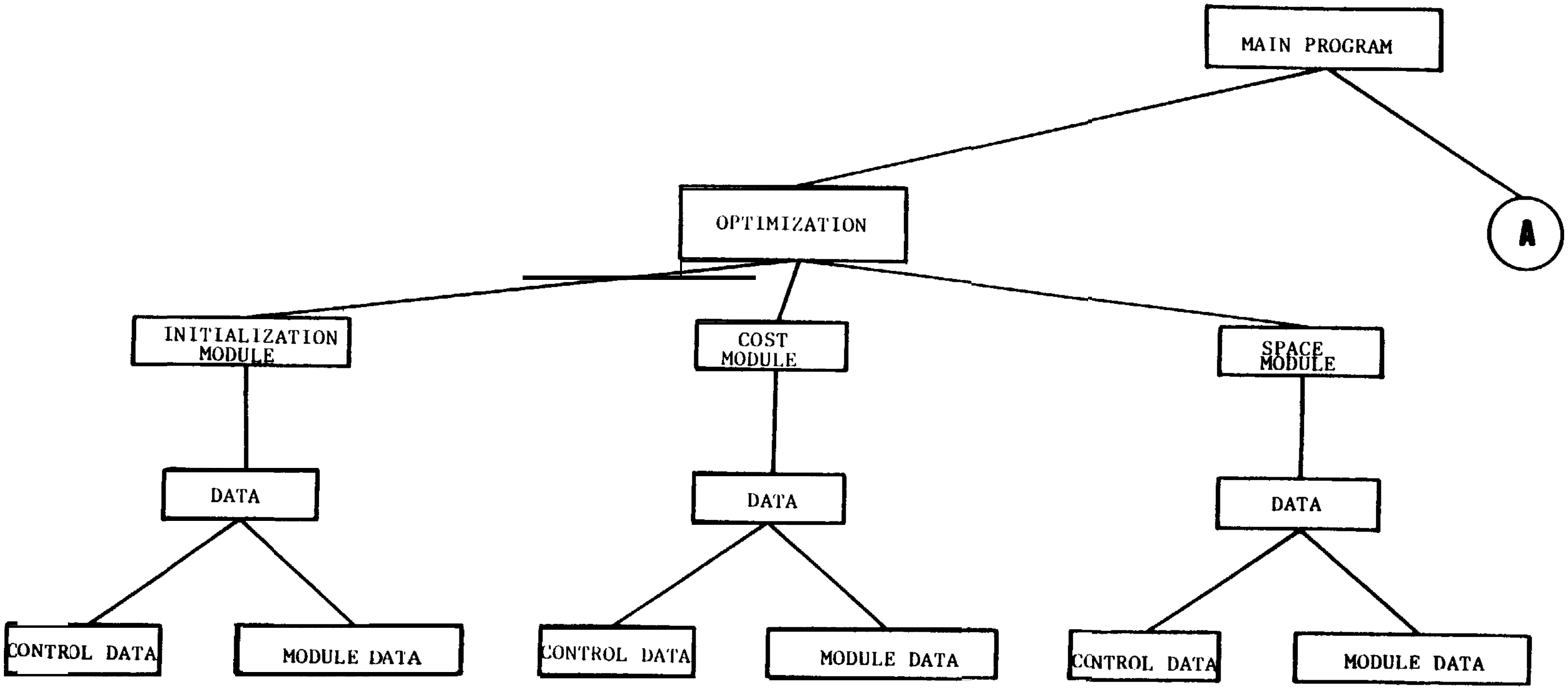


Fig 8.3. Data Chart Organization of the ASSET/COPES/GRAPHICS Synthesis System

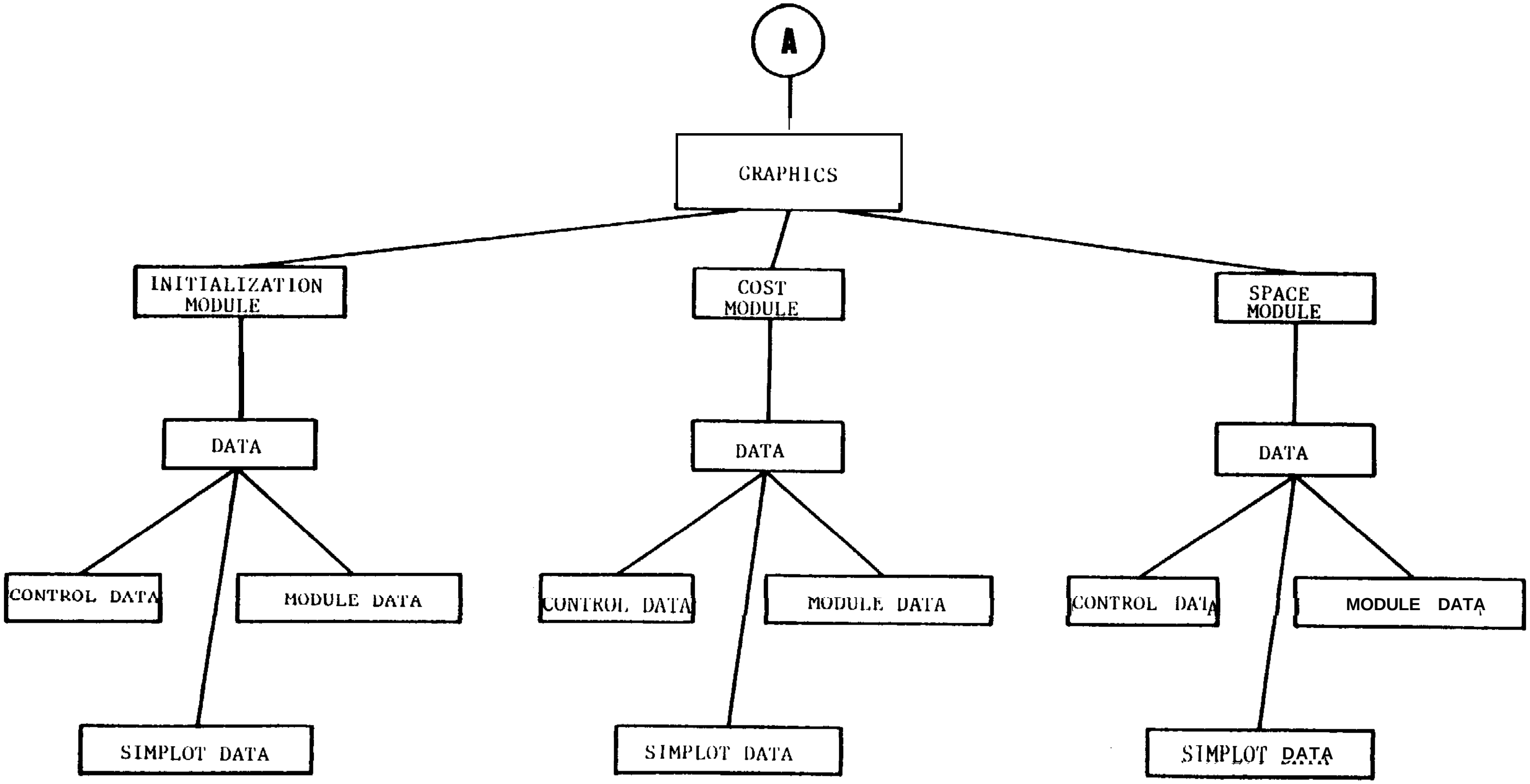


Fig. 8.3. Continued. Data Chart Organization of the ASSET/COPES/GRAPHICS Synthesis System

model: printed or graphical. Execution of the **OPTIMIZATION** mode concludes with a detailed output with information on the optimization process and on the optimum design for each one of the three modules. INITIALIZATION, COST or SPACE. upon user request. Execution of the GRAPHICS mode concludes with both types of output: a 3-D perspective plot and two 2-D plots which represent the design space for a specified set of one objective function and two design variables. In addition. printed output may be obtained which represents the whole design space and gives detailed information for every point of the design space grid. The user controls all the outputs and also the amount of printed information needed.

A summarized presentation of the new synthesis model output for both execution modes is given in Tables 8.1 and 8.2 respectively.

INITIALIZATION MODULE

Menu	Description
#1	Summary
#2	Hull Geometry
#3	Hull Structure
#4	Resistance
#5	Propeller
#6	Machinery
#7	Weight
#8	Hydrostatics and Seakeeping

Optimization Information

COST MODULE

#1	Summary
#2	Unit Acquisition Costs
#3	Life-Cycle Costs

Optimization Information

SPACE MODULE

#1	Summary
#2	SSCS1 Space Requirements
#3	SSCS2 Space Requirements
#4	SSCS3 Space Requirements

Optimization Information

Table 8.1. Printed Output of the Optimization Mode of the ASSET/COPES/GRAPHICS Synthesis System

INITIALIZATION MODULE

PRINTED OUTPUT

MENU	DESCRIPTION
#1	Summary
#2	Hull Geometry
#3	Hull Structure
#4	Resistance
#5	Propeller
#6	Machinery
#7	Weight
#8	Hydrostatics and Seakeeping

GRAPHICAL OUTPUT

PLOT NUMBER	DESCRIPTION
#1	WTFULL = f(LBP,LBP/B)
#2	WTFULL = f(LBP,CP)
#3	WTFULL = f(LBP,CX)
#4	WTFULL = f(CP,LBP/B)
#5	WTFULL = f(CX,LBP/B)
#6	WTFULL = f(CP,CX)

PLOT DESCRIPTION

PLOT NUMBER	AXIS DESCRIPTION
#1	OX: LBP
	OY: LBP/B
	OZ: WTFULL
#2	OX: LBP
	OY: CP
	OZ: WTFULL
#3	OX: LBP
	OY: CX
	OZ: WTFULL
#4	OX: CP
	OY: LBP/B
	OZ: WTFULL
#5	OX: CX
	OY: LBP/B
	OZ: WTFULL
#6	OX: CP
	OY: CX
	OZ: WTFULL

Table 8.2. Printed and Graphical Output of the Graphics Mode of the ASSET/COPES/GRAPHICS Synthesis System

COST MODULE

PRINTED OUTPUT

MENU	DESCRIPTION
#1	**Summary**
#2	Unit Acquisition Costs
#3	Life-Cycle costs

GRAPHICAL OUTPUT

PLOT NUMBER	DESCRIPTION
#1	SYSCPS = f(LBP,LBP/B)
#2	SYSCPS = f(LBP,CP)
#3	SYSCPS = f(LBP,CX)
#4	SYSCPS = f(CP,LBP/B)
#5	SYSCPS = f(CX,LBP/B)
#6	SYSCPS = f(CP,CX)

PLOT DESCRIPTION

PLOT NUMBER	AXIS DESCRIPTION
#1	OX: LBP OY: LBP/B OZ: SYSCPS
#2	OX: LBP OY: CP OZ: SYSCPS
#3	OX: LBP OY: CX OZ: SYSCPS
#4	OX: CP OY: LBP/B OZ: SYSCPS
#5	OX: CX OY: LBP/B OZ: SYSCPS
#6	OX: CP OY: CX OZ: SYSCPS

Table 8.2. Continued. Printed and Graphical Output of the Graphics Mode of the ASSET/COPES/GRAPHICS Synthesis System

SPACE MODULE

PRINTED OUTPUT

MENU		DESCRIPTION
#1	-	summary
#2	-	SSCS1 Space Requirements
#3	-	SSCS2 Space Requirements
#4	-	SSCS3 Space Requirements

GRAPHICAL OUTPUT

PLOT NUMBER	DESCRIPTION
#1	VPAYL/TOTVOL = f(LBP,LBP/B)
#2	VPAYL/TOTVOL = f(LBP,CP)
#3	VPAYL/TOTVOL = f(LBP,CX)
#4	VPAYL/TOTVOL = f(CP,LBP/B)
#5	VPAYL/TOTVOL = f(CX,LBP/B)
#6	VPAYL/TOTVOL = f(CP,CX)

PLOT DESCRIPTION

PLOT NUMBER	AXIS DESCRIPTION
#1	OX: LBP OY: LBP/B OZ: VPAYL/TOTVOL
#2	OX: LBP OY: CP OZ: VPAYL/TOTVOL
#3	OX: LBP OY: CX OZ: VPAYL/TOTVOL
#4	OX: CP OY: LBP/B OZ: VPAYL/TOTVOL
#5	OX: CX OY: LBP/B OZ: VPAYL/TOTVOL
#6	OX: CP OY: CX OZ: VPAYL/TOTVOL

Table 8.2. Continued. Printed and Graphical Output of the Graphics Mode of the ASSET/COPES/GRAPHICS Synthesis System

9.0. DESIGN STUDIES

This section will present the results of studies performed using the ASSET/COPES/GRAPHICS synthesis system.

9.1. ASSET/COPES/GRAPHICS Example

9.1.1. Problem Statement

The prime purpose of a naval ship is the transportation of weapons and the equipment to direct them. Thus the ultimate goal of the design process is to develop a ship which represents a reasonable balance between cost and military performance. The starting point of the design process is the mission requirements. In the case of a naval combatant, this includes the military payload and mission to be accomplished over the lifetime of the ship. More explicitly, these requirements would include definition of payload (weapons and sensors), maximum sustained speed, endurance speed and range, habitability standards, future growth margins and design life of the ship. These requirements are mutually dependent. For example, mission requirements will generally dictate the type of equipment, the required operating crew, and additional materials to support them. A partial list of payload items for an FFG7 naval frigate ship is given in Table 9.1, taken from References [26 and 27].

After the mission requirements have been specified, the designer proceeds to the next step of ship design. Here the parameters which define the ship's form are selected. This is done through sensitivity studies which compare different hull materials, propulsion plants, geometric parameters such as length and coefficients of form, electric plant parameters, number of crew members, habitability standards, etc. As a result of this **step,** the designer must develop relatively accurate values for the ship specifications to use as input for the

FF(ASW command and control) Radio communications
SPS-49 Radar w/IFF
SPS-55 Radar w/IFF
SQS-56 Sonar
FF/FFG Basic ECM suite
ASWC and C-FF-2C,7D Electronic Tactical Data Systems
Vulcan/Phalanx on 01Lv
76mm/62-Caliber Oto Melara Gun
Mk-92 CIWS/STIR
800 3"/50 rounds
10000 20mm rounds
Mk-13 Tartar Missile Launcher w/40 Missiles
Mk-32 Triple Torpedo Tubes P/S w/24 Torpedos
Harpoon FCS
2 SH-3 Helos with supports

Table 9.1. Frigate Payload Items.

synthesis model. The ASSET/COPES/GRAPHICS synthesis system allows any starting value that is within the specified limits of the design variables. Thus it is possible to start with an entirely unacceptable design and yet end up with an acceptable one after optimization.

The user of the ASSET/COPES/GRAPHICS system needs only to select the design variables, an objective function and the constraints from the list of ship specifications, identify them, and run the synthesis system. Table 9.2 is a list of the variables which must be input for this example. The user may select the parameters used for optimization from the same table.

9-1.2. Parameter Selection

In previous sections, the selection of the necessary parameters to apply optimization was discussed extensively. The objective function of this example is the minimization of the full load displacement (WTFULL) WTFULL is listed in global location 126 in the GLOBCM statement and identified as the objective function to be minimized in data block E of the COPES input data..

Four design variables were selected:

1. length between perpendiculars (LBP) - global location 28 in the GLOBCM common block
2. length to beam ratio (LBP/B) - global location 26
3. prismatic coefficient (CP) - global location 24
4. midsip section coefficient (CX) - global location 25.

After the selection of the design variables, it is necessary to specify their global location in the GLOBCM statement and to specify any side constraints that may be imposed on them. The data block F of the COPES optimizer is used to specify side constraints and design variable number, and block G specifies global location. In Table 9.3,

- MISSION
  - Design Speed Required
  - Range Speed Required
  - Range
  - Mission Duration
- PAYLOAD ITEMS
- HULL GEOMETRY
  - LBP
  - LBP/B
  - LBP/D
  - T/D
  - Prismatic Coefficient
  - Maximum Section Coefficient
- HULL MATERIAL ITEMS
- HULL LOADS
  - Hogging Bending Moment
  - Sagging Bending Moment
- DECKHOUSE GEOMETRY
- DECKHOUSE MATERIAL ITEMS
- PROPULSION
  - Number of Main Engines
  - Main Engine Cont. HP Avail.
  - Number of Second. Engines
  - Second. Eng. Cont. HP Avail.
- ELECTRIC PLANT ITEMS
- TRANSMISSION
  - Design Transmission Efficiency
  - Range Transmission Efficiency
- PROPELLER
  - No. Propeller Shafts
  - Propeller Diameter
- RESISTANCE
  - Correlation Allowance
  - Friction Line
- WEIGHTS
  - Full Load Weight
  - Usable Fuel Weight
- ACCOMODATION ITEMS
- DESIGN MARGINS
- ECONOMIC FACTORS
- PAYLOAD COST FACTORS
- SHIP COST FACTORS

Table 9.2. ASSET/COPES/GRAPHICS Input Variables.

PARAMETER	LOWER BOUND	UPPER BOUND
LBP	300.00	700.00
LBP/B	7.00	12.00
CP	0.52	0.68
CX	0.70	0.90

Table 9.3. Side Constraints on Design Variables

-1.0E+06 ≤ GMRREQ = (GMMIN/GM) - 1.0 ≤ 0.0

-1.0E+06 ≤ FBRREQ = (FBDMIN/FBDACT) - 1.0 ≤ 0.0

-1.0E+06 ≤ PHPRAT = (PHPREQ/PHPAVL) - 1.0 ≤ 0.0

-1.0E+06 ≤ RATIO1 = (0.85/RATIO) - 1.0 ≤ 0.0

-1.0E+06 ≤ RATIO2 = (RATIO/1.15) - 1.0 ≤ 0.0

-1.0E+06 ≤ BT1 = (2.25/BT) - 1.0 ≤ 0.0

-1.0E+06 ≤ BT2 = (BT/3.75) - 1.0 ≤ 0.0

-1.0E+06 ≤ CP1 = (0.52/CP) - 1.0 ≤ 0.0

-1.0E+06 ≤ CP2 = (CP/0.68) - 1.0 ≤ 0.0

-1.0E+06 ≤ CV1 = (0.001/CV) - 1.0 ≤ 0.0

-1.0E+06 ≤ cv2 = (CV/0.002) - 1.0 ≤ 0.0

-1.0E+06 ≤ SL1 = (-1.0) * SL ≤ 0.0

-1.0E+06 ≤ SL2 = (SL/2.0) - 1.0 ≤ 0.0

-1.0E+06 ≤ WTFUEL1 = (-1.0) * WTFUEL ≤ 0.0

-1.0E+06 ≤ PCHDIA1 = (0.68/PCHDIA) - 1.0 ≤ 0.0

-1.0E+06 ≤ PCHDIA2 = (PCHDIA/3.4) - 1.0 ≤ 0.0

Table 9.4. Design Constraints.

the **upper** and lower bounds of the design variables are illustrated.

The final task is to specify the constraints which will eliminate unacceptable designs. Table 9.4 is a list of all the imposed constraints in this study. The data blocks H and I of the COPES input data are related to the specification of these constraints. Data block H specifies the number of constraints, while data block I identifies the global location of each one of them and specifies the upper and lower bounds and the scaling factor desired for them. For this example, the upper bound is zero, the lower bound is numerically minus infinity and the scaling factor is defaulted to 0.1. Table 9.5 shows the global location in the GLOBCM statement of the objective function, the design.variables and the constraints of this example. Table 9.6 lists the COPES/CONMIN data blocks illustrating the input data described above.

9.1.3. Input and Output

Both options of the ASSET/COPES/GRAPHICS synthesis system, the optimization and the graphics, have been used in this example. Thus the system requires two different types of input. Input for the optimization mode is shown in Table 9.7, and input data for the graphics mode are presented in Table 9.8.

The printed output consists of the initial design, the final optimum design, and information on the optimization.. Graphical output consists of the full load displacement as a function of the length between perpendiculars, the length to beam ratio, the prismatic coefficient and the midship section coefficient. The results of the optimization mode are presented in tabular form in Table 9.9. The initial and optimum values for the objective function, design variables, and constraints are illustrated in this table.

GLOBAL	FORTRAN	DEFINITION LOCATION NAME
126	**WTFULL**	Full Load Displacement, Tons
24	HCP	Prismatic Coefficient
25	HCX	Midship Section Coefficient
26	HLBPB	Length to **Beam** Ratio
28	HLBP	Length Between Perpendiculars, Feet
150	FBRREQ	FBDMIN/FBDACT - 1.0
151	**GMRREQ**	**GMMIN/GM** - 1.0
152	PHPRAT	PHPREQ/PHPAVL - 1.0
154	RATIO1	**0.85/RATIO** - 1.0
155	RATIO2	RATIO/1.15 - 1.0
156	**BT1**	2.25/BT - 1.0
157	BT2	BT/3.75 - 1.0
158	**CP1**	**0.52/CP** - 1.0
159	**CP2**	**CP/0.68** - 1.0
160	**CV1**	0.001/CV - 1.0
161	cv2	CV/0.002 - 1.0
162	**SL1**	(-1.0) * SL
163	**SL2**	**SL/2.0** - 1.0
164	**WTFUEL1**	(-1.0) * WTFUEL
165	**PCHDIA1**	**0.68/PCHDIA** - 1.0
166	PCHDIA2	**PCHDIA/3.4** - 1.0

Table 9.5. Global Location and Definition of the Objective Function, Design Variables, and Design Constraints

```
$ BLOCK A
MINIMIZATION OF FULL LOAD DISPLACEMENT FOR FFG7 SHIP
$ BLOCK B
2,4
$ BLOCK C
3,,,,,,10
$ BLOCK D
0.0
0.0
$ BLOCK E
0,126,-1.0
$ BLOCK F
300.0,700.0
7.0,12.0
0.52,0.68
0.7,0.9
$ BLOCK G
1,28,1.0
2,26,1.0
3,24,1.0
4,25,1.0
$ BLOCK H
2
$ BLOCK I
150,152
-1.0+6,0.0,0.0
154,156
-1.0+6,0.0,0.0
$ BLOCK V
END
```

Table 9.6. COPES/CONMIN Data Blocks

```
   DATA  FILE:  TAPE8
OPTIMUM

   DATA FILE:  DATA5
MINIMIZATION OF FULL LOAD DISPLACEMENT FOR FFG7 SHIP
2,4
3,,,,,,10
0.0
0.0
0,126,-1.0
300.0,700.0
7.0,12.0
0.52,0.68
0.7,0.9
1,28,1.0
2,26,1.0
3,24,1.0
4,25,1.0
2
150,152
-1.0+6,0.0,0.0
154,166
-1.0+6,0.0,0.0
END
INTLZN
 1987654321000000000000000000
GIVEN     FUEL WT   AL 5086   ITTC      MS
GIVEN     GIVEN     GIVEN     CP        GIVEN
GT        GIVEN     DIESEL    NONE      GIVEN
GIVEN     MECH
75542.1   52756.3      0.5E-030.513     0.6178
13.00     15.00     165.00    165.888   0.416231
30.00     0.0       1.0       0.978700  0.955
0.535600  C.74940   8.906     13.60     380.00
1.12      19.040    0.478333  489.024   16.50
20500.0   1.16916   2.0       1.0       0.435
5574.54      0.1E+37   0.1E+37   0.1E+37   0.1E+37
4000.0      0.76E-0129.0      20.0        0.76E-01
1.08      1.12      1.20      1.26      1.29
1.27      1.20      1.11      1.01      0.93
0.83      0.85      0.84      0.83      0.84
0.84      0.85      0.85      0.86      0.86
0.87      0.87      0.87      0.87      0.87
0.87      0.87      0.87      0.87      0.87
0.87      0.0       0.0       0.0       0.0
0.0       0.0       0.0       60.62     40.21
9.3       21.98     0.0       32.07     35.12
22.3      5.58      1.70      0.0       0.0
0.0       0.0       0.0       33.98     6.65
19.55     19.60     18.71     569.601   0.0
5500.00   250000.0  70000.0   30000.0   6000.0
```

Table 9.7. Input Data to Execute the Optimization Mode for Minimization of the Full Load Displacement.

DATA FILE : TAPE6

```
GRAPHICS
INTLZN      LBP         LBPB
300.0       400.0       500.0       600.0       TOO.0
7.0         8.0         9.0         10.0        11.0        12.0
0.600       0.800
```

DATA FILE: DATA5

```
 12222222210
GIVEN       FUEL WT     A L 5086    ITTC        MS
GIVEN       GIVEN       GIVEN       CP          GIVEN
GT          GIVEN       DIESEL      NONE        GIVEN
GIVEN       MECH
75642.1     52756.3        0.5E-030.513         0.6178
13.00       15.00       165.00      165.888     0.416231
30.00       0.0         1.0         0.978700    0.955
13.60       1.120       19.040      0.478333    419.024
16.50       20500.0     1.16916     2.0         1.0
0.435       5577.54        0.1E+37     0.1E+37     0.1E37
   0.1E+374000.0         0.76E-0129.0          2.0.0
  0.76E-011.08          1.12        1.20        1.26
1.29        1.27        1.20        1.11        1.01
0.93        0.88        0.85        0.84        0.83
0.84        0.84        0.85        0.85        0.86
0.86        0.87        0.87        0.87        0.87
0.87        0.87        0.87        0.87        0.87
0.87        0.87        0.0         0.0         0.0
0.0         0.0         0.0         0.0         60.62
40.21       9.3         21.98       0.0         32.07
35.12       22.30       5.58        1.70        0.0
0.0         0.0         0.0         0.0         33.98
6.65        19.55       19.60       18.71       569.601
0.0         4000.00
```

DATA FILE: SIMPDAT

```
VARIABLE LIST   HLBP,WTFULL1,WTFULL2,WTFULL3,WTFULL4,WTFULL5,WTFULL6
INPUT MEDIUM    TAPE 19
INPUT FORMAT
OUTPUT DEVICE   PLOTTER
MULTIPLOT       LINE,HLBP VERSUS WTFULL1 / LINE,HLBP VERSUS WTFULL2 /
                LINE,HLBP VERSUS UTFULL 3 /LINE,HLBP VERSUS WTFULL4 /
                LINE,HLBP VERSUS UTFULL5 /LINE,HLBP VERSSU WTFULL6
PLOT OPTIONS    HORIZONTAL=GRID200.0 TO 800.0,DIVISIONS=6, DIGITS=2/
                VERTICAL=GRID, $000.0 TO 8000.0, DIVISIONS=12, DIGITS=2
BOTTOM LABELS   LBP IN FEET
SIDE LABELS     FULL LOAD DISPLACEMENT IN LTONS
FINISH
VARIABLE LIST   HLBPB WTFULL1,WTFULL2,WTFULL3,WTFULL4,WTFULL5
INPUT MEDIUM    TAPE16
INPUT FORMAT
OUTPUT DEVICE   PLOTTER
MULTIPLOT       LINE,HLBPB VERSUS WTFULL1/LINE,HLBPB VFRSUS WTFULL2/
                LINE,HLBPB VERSUS WTFULL3 / LINE,HLBPB VERSUS WTFULL4/
                LINE,HLBPB VFRSUS WTFULL5
PLOT OPTIONS    HORIZONTAL=GRID, 6.0TO 13.0, DIVISIONS,7 DIGITS=2 /
                VERTICAL-GRID, 2000.0 TO 8000.0, DIVISIONS=12, DIGITS=2
BOTTOM LABELS   LBP/B RATIO
SIDE LABELS     FULL LOAD DISPLACEMENT IN LTONS
FINISH
```

Table 9.8. Input Data to Execute the Graphics Mode for WTFULL as a Function of LBP and LBP/B.

PARAMETER	INITIAL VALUE	OPTIMUM VALUE
Objective Function		
WTFULL	3425.9000	3375.2000
Design Variables		
LBP	380.0000	389.3600
LBP/B	8.9060	8.7370
CP	0.5356	0.5200[1]
CX	0.7494	0.9000[2]
Constraints		
GMRREQ	-0.2501	-0.0027
FBRREQ	-0.7677	-0.8070
PHPRAT	-0.0496	-0.0160
RATIO1	0.17113	-0.0970
RATIO2	-0.3688	-0.1810
BT1	-0.1464	-0.2800
BT2	-0.2971	-0.1660
CP1	-0.0967	-0.0155
CP2	-0.1534	-0.2230
CV1	-0.5421	-0.5000
CV2	0.0918[3]	0.0000[4]
SL1	-1.0260	-0.1010
SL2	-0.4870	-0.4930
WTFUEL1	-569.6000	-569.6000
PCHDIA1	-0.4490	-0.4440
PCHDIA2	-0.6366	-0.6399

[1] lower bound of design variable
[2] upper bound of design variable
[3] violated constraint
[4] active constraint

Table 9.9. Initial and Optimum Parameter Values

It may be noted that the initial design is infeasible, with constraints RATIO1 and CV2 violated. Physically, this means' that there was not equilibrium between the displacement and the weight. The optimizer increased the displacement by manipulating the design variables until all of the constraints were satisfied. The constraint CV2 in the optimum design is indicated as active, which means that it lies within a specified tolerance value of the constraint zero value boundary.

The results of the graphics mode are illustrated in Figures 9.1 through 9.6. The full load displacement is graphically presented as a function of the four design variables. It may be noted that the length between perpendiculars is the most critical parameter in the estimation of the full load displacement. The prismatic coefficient and the midship section coefficient have little effect on the full load displacement, as may be seen from the graphics. As has been stated, the graphics represent the unconstrained design space, that is the design constraints are not added to graphics output in this version. Imposition of the constraints on the design space will lead to the same final optimum design as determined in the optimization mode.

Finally, Figure 9.7 shows the objective function versus the number of iterations required to reach the optimum. It can be seen that the optimizer is indeed efficient and arrived at a value near the optimum in four iterations. The remaining four iterations were done simply to "fine tune" the value of the objective function. The average run time for this example was approximately 1.377 CPU seconds on a CDC 6600 system. The entire example is contained in Appendix F.

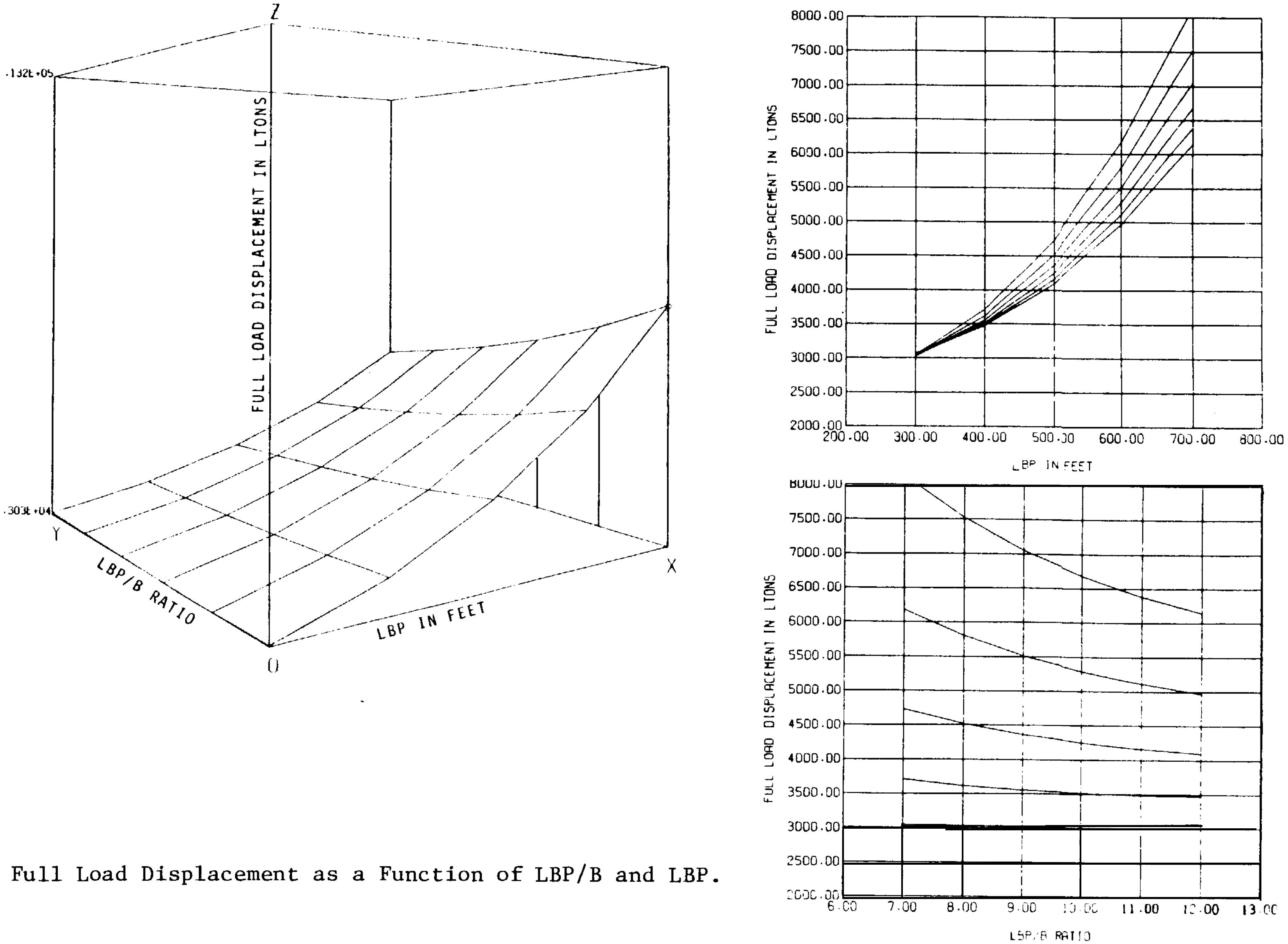
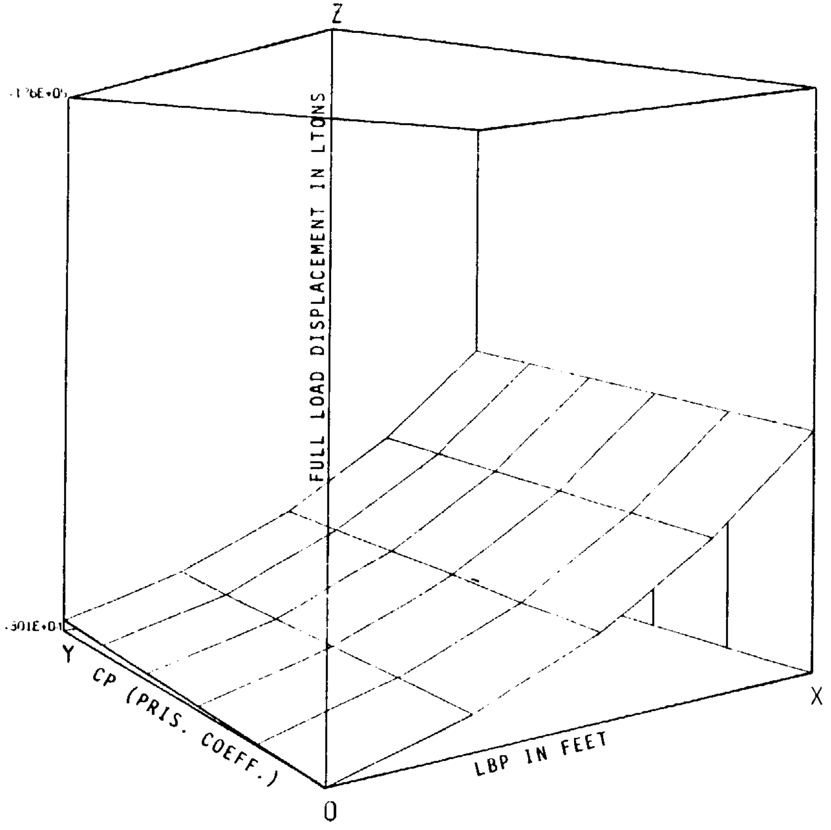


Fig. 9.1. Full Load Displacement as a Function of LBP/B and LBP.



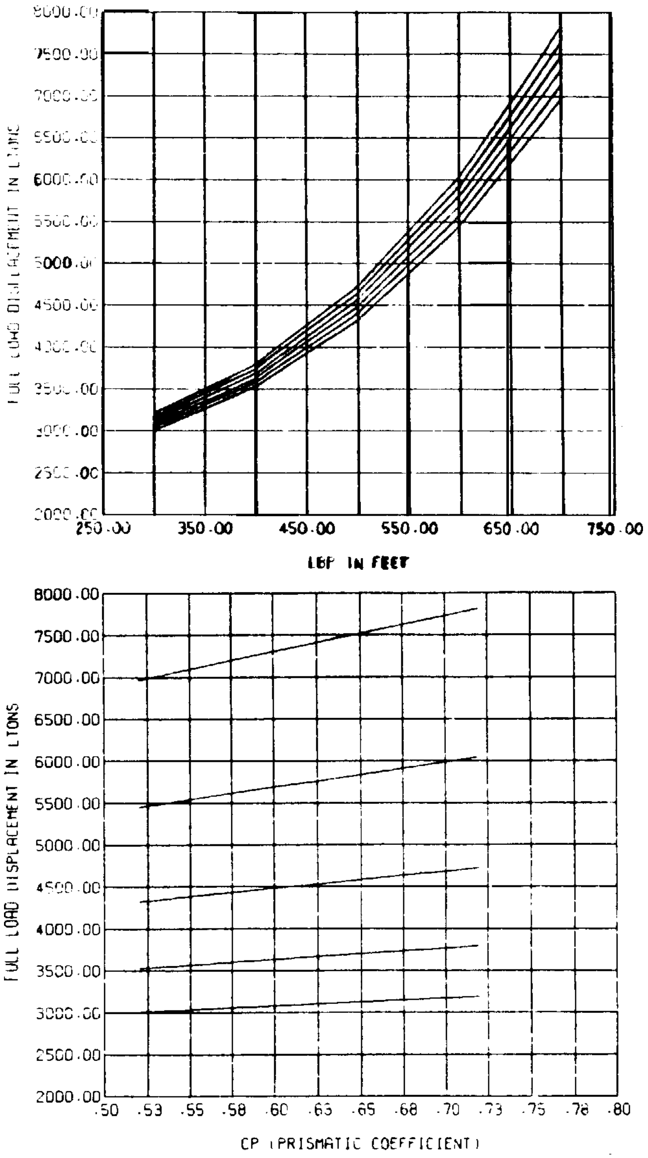
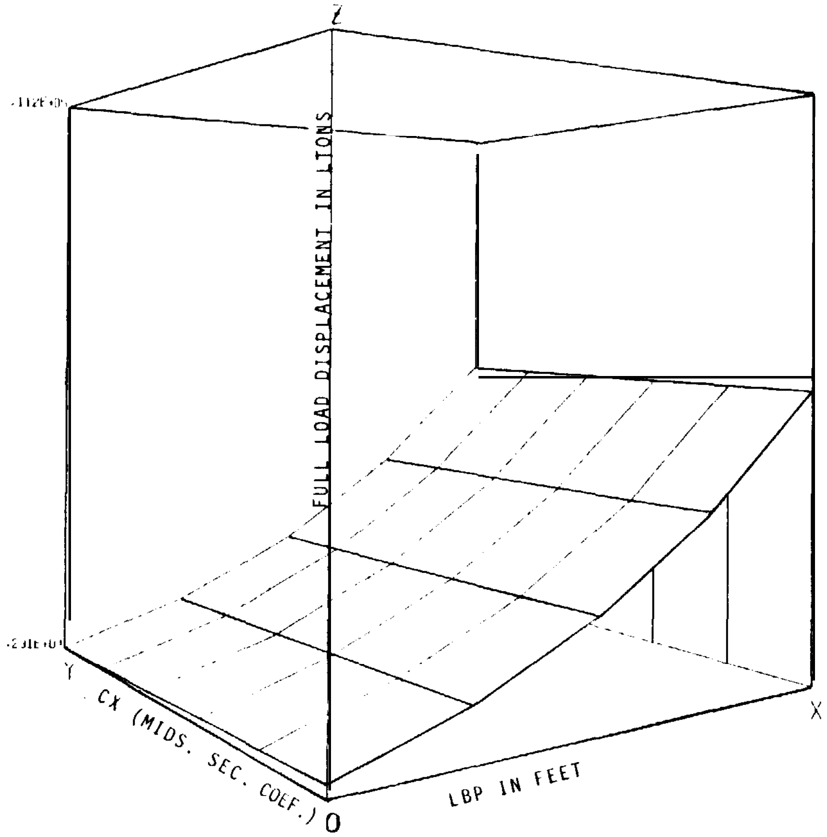
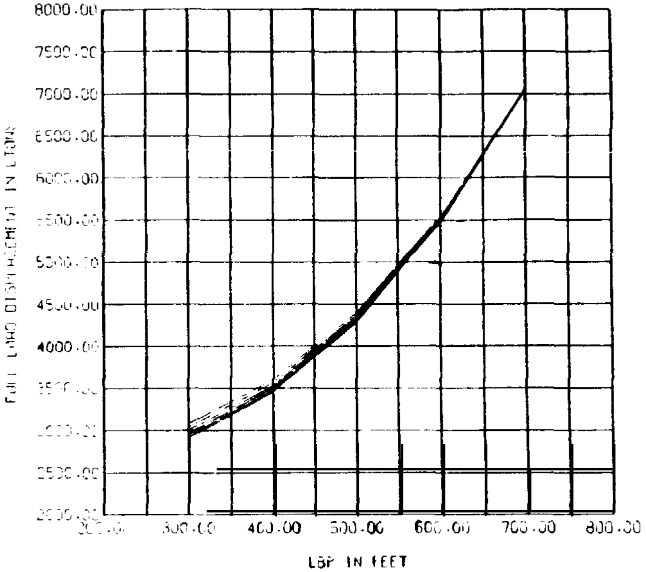


Fig. 9.2 Full Load Displacement as a Function of C_p and LBP.





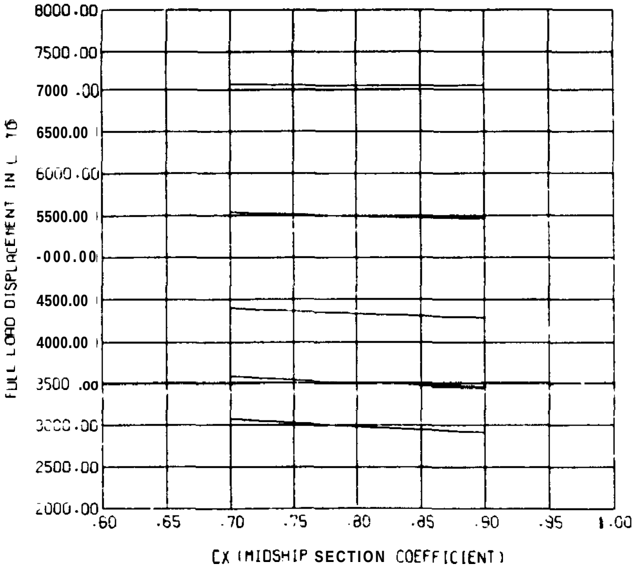
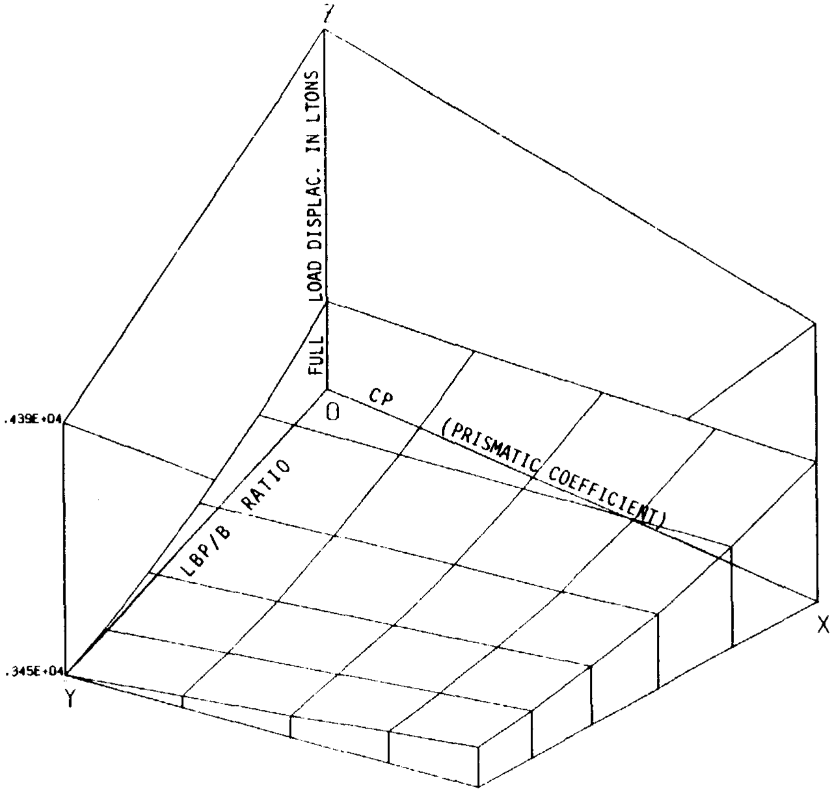
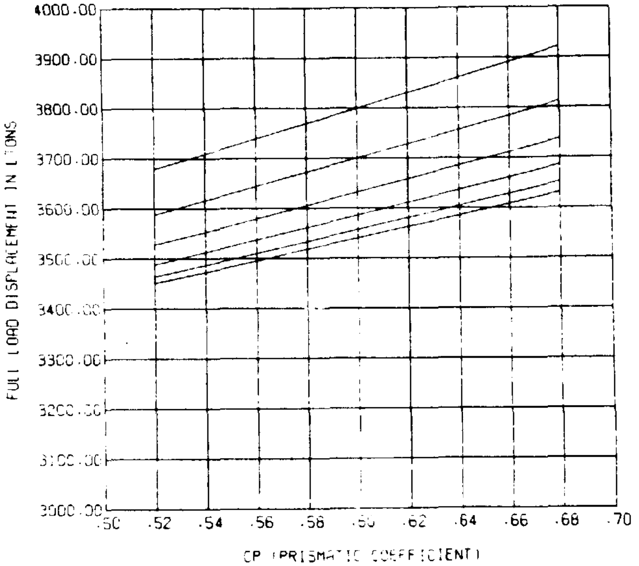


Fig. 9.3. Full Load Displacement as a Function of C_x and LBP.





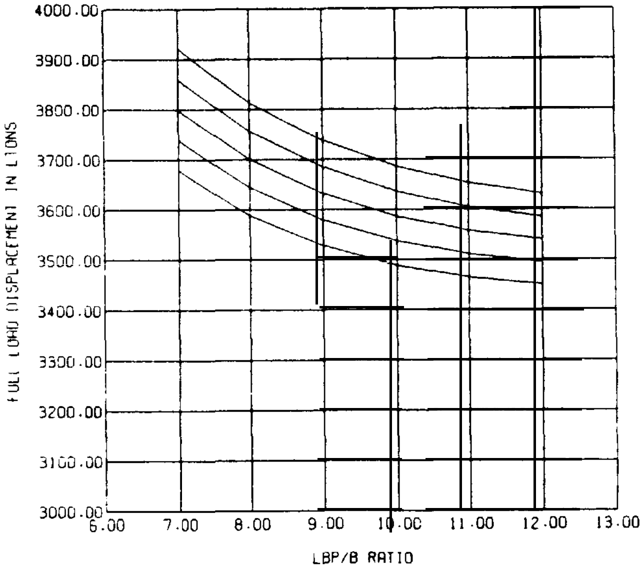
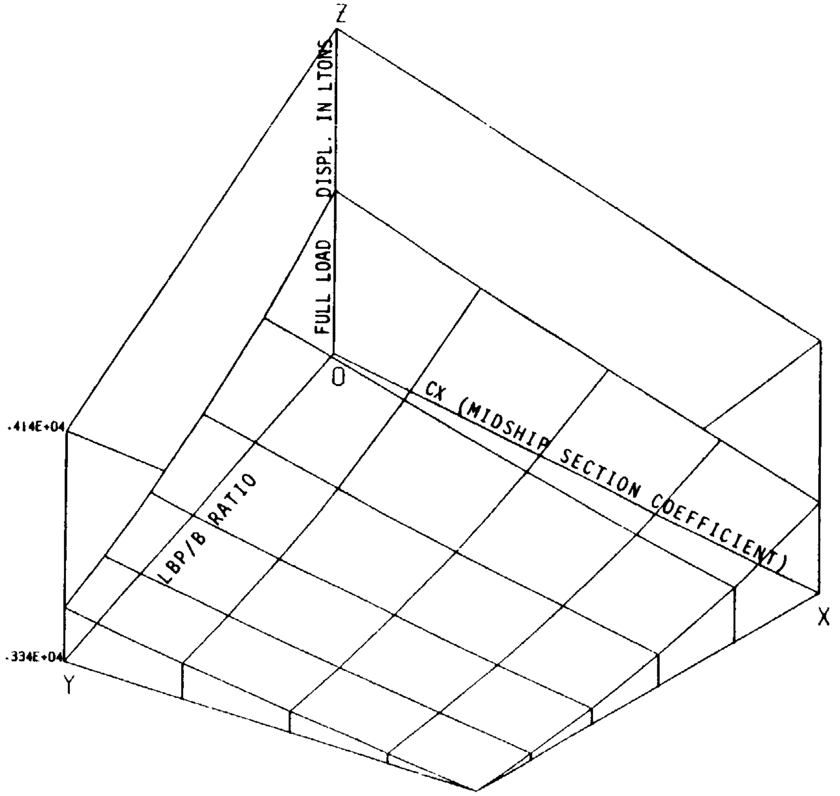


Fig. 9.c. Full Load Displacement as a Function of LBP/B and C_p.



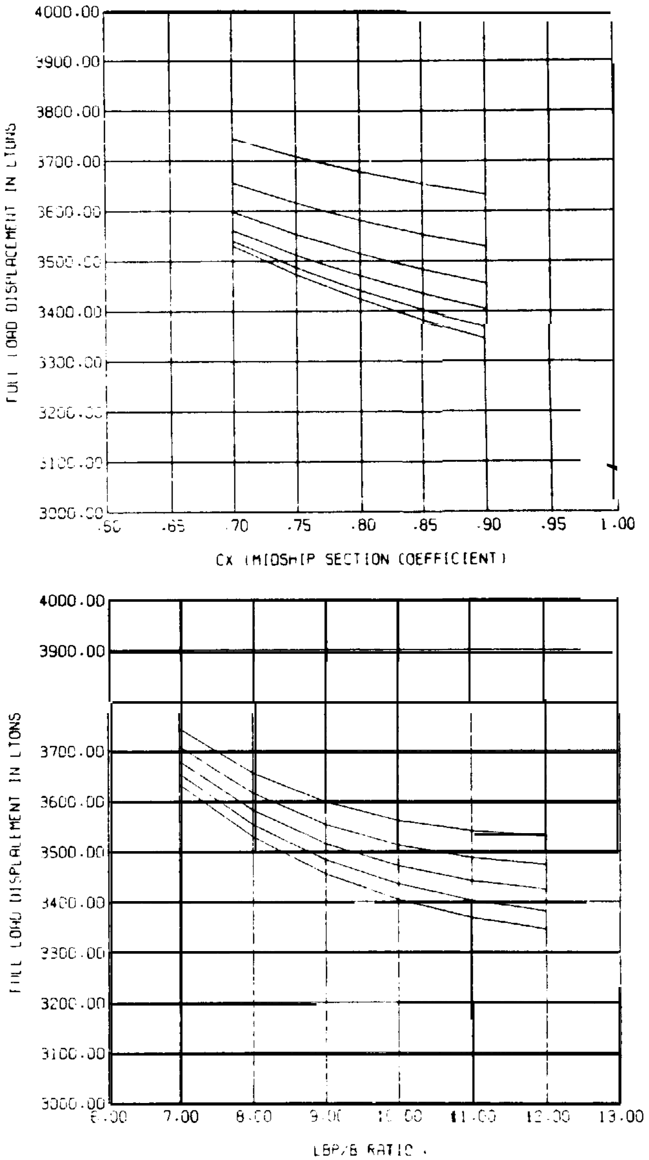


Fig. 9.5 Full Load Displacement as a Function of LBP/B and C_x

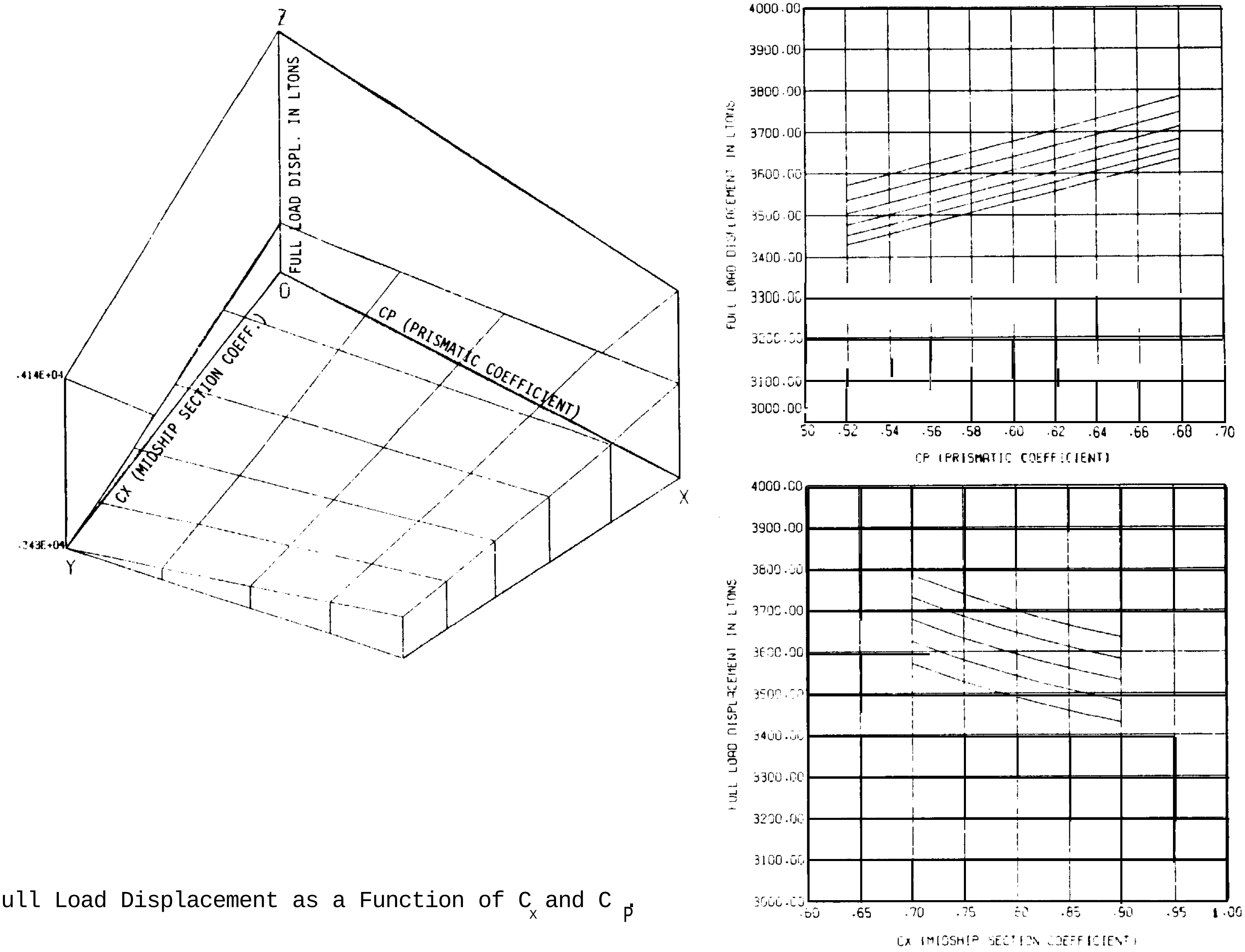


Fig. 9.6. Full Load Displacement as a Function of C_x and C_p

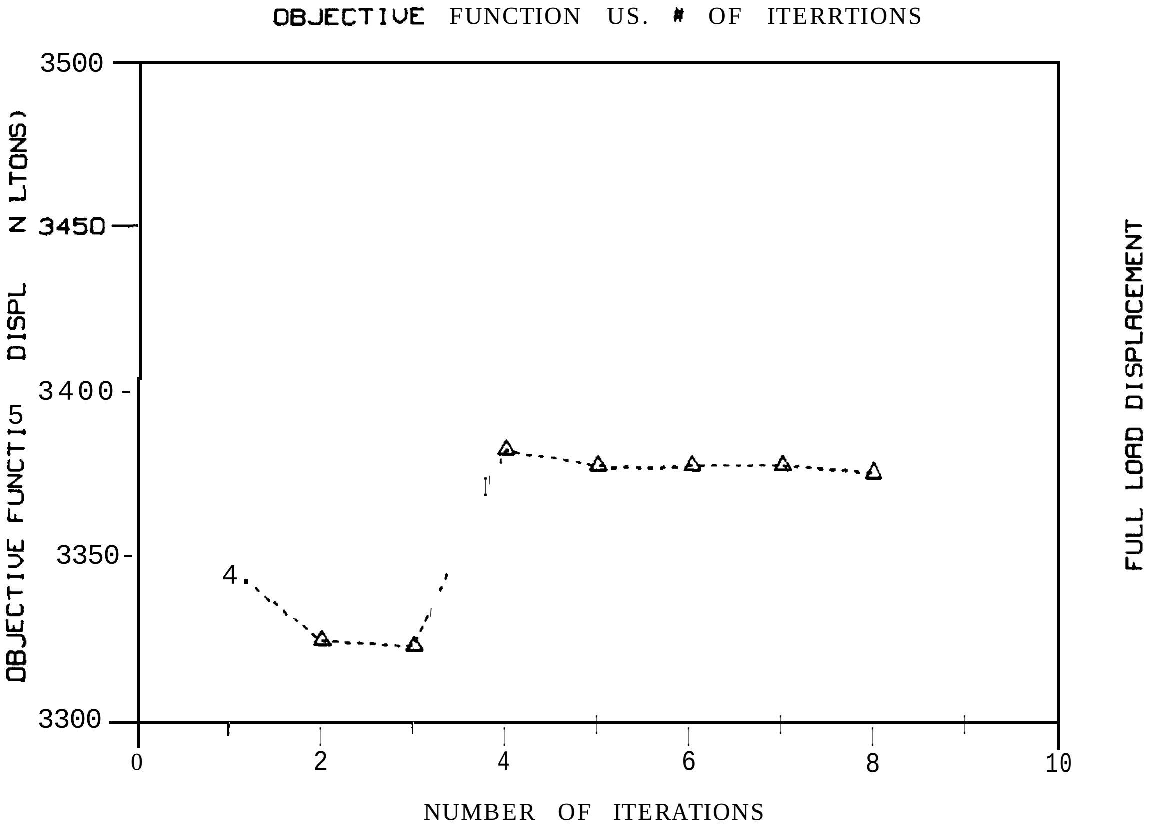


Fig. 9.7. Objective Function Versus Number of Iterations for the Optimization Process.

9.2. Multiple Objective Functions

A number of different objective functions are used here for comparison of the final products. These were:

1. minimize full load displacement
2. minimize length between perpendiculars
3. minimize resistance at design speed
4. minimize life cycle cost per ship
5. maximize military mission volume to total volume.

Length between perpendiculars, which is also a design variable, was chosen as an objective function in this case because of its relationship to the displacement. Minimum resistance at design speed, which is an indication of fuel efficiency and minimum life cycle cost per ship, which are desirable from a mission standpoint.. Maximization of the military mission volume to total volume ratio is an indication of an effective volume usage. All of the designer's requirements, payload, fuel available weight, etc., remained constant for all designs. The design variables, side constraints and behavioral constraints in the optimization process were the same as those in Tables 9.3 and 9.4.

Table 9.10 tabulates the results of this study. It can be seen that the main ship particulars remain almost the same regardless of the objective function. This indicates that:

a) The final product is a well defined optimum ship, characterized as the global optimum in the design space.

b) The results for minimum full load displacement, length between perpendiculars, life cycle cost per ship and maximum military mission volume to total volume ratio are essentially the same, indicating the indirect

PARAMETER	MINIMUM WTFULL	MINIMUM LBP	MINIMUM DRAGD	MINIMUM SYSCPS	MAXIMUM VPAYL/TOTVOL
VDESGN	29.000	29.000	29.000	29.000	29.000
VRANGE	20.000	20.000	20.000	20.000	20.000
DMISSN	30.000	30.000	30.000	30.000	30.000
WTMILP	327.400	327.400	327.400	327.400	327.400
LBP	389.360	389.390	398.050	391.280	390.720
LBP/B	8.736	8.740	8.649	8.998	8.998
CP	0.520	0.520	0.529	0.520	0.520
CX	0.900	0.900	0.782	0.839	0.835
B	44.570	44.550	46.020	43.490	43.420
T	13.690	13.700	14.000	13.760	13.740
LBP/D	13.599	13.600	13.599	13.600	13.599
RANGE	6383.010	6383.190	6264.130	6412.350	6415.320
WTFULL	3375.200	3375.100	3521.200	3407.900	3406.600
WTFUEL	569.600	569.600	569.600	569.600	569.600
PHPREQ	40340.000	40326.000	37710.000	38270.000	38246.000
VOLTOT	457338.000	457250.000	476001.000	440145.000	437625.000
FUEL CONS.	11.206	11.206	10.997	11.258	11.263
GM/GMMIN	1.003	1.003	1.460	1.001	1.003
SYSCPS	807.200	809.000	811.900	808.660	806.500
VPAYL/TOTVOL	0.229	0.229	0.228	0.231	0.231

Table 9.10. Results from the Optimization Mode for Different Objectives Functions

relation of all the above functions.

c) Life cycle cost per ship for 30 years of service life differs very little for ships having slightly different principal dimensions.

A time and cost summary of each module for the optimization and graphics mode is presented in Table 9.11 and Figure 9.8. This illustrates another important feature of the ASSET/COPES/GRAPHICS synthesis system.

9.3. Testability and Reliability

At this point it is important to discuss two other properties of the system, testability and reliability. Testability is the ability of the system to produce a viable version of an existing ship design, and to compare the system output to it.

9.3.1. Alternative Studies Comparison

Results for an **FFG7** frigate ship were obtained from several different sources and compared with the results of the ASSET/COPES/GRAPHICS system for three objective functions: minimization of full load displacement, life cycle cost per ship and maximization of military mission volume to total volume ratio. The results are compared with those taken from different references in Table 9.12. The first three columns present the results of ASSET/COPES/GRAPHICS system, the fourth column shows results from the ASSET synthesis model provided by the Boeing Company [10], the fifth column shows results from the REED/COPES model by Jenkins [16], and the last column presents data obtained from the FFG7 ship design from Garzke and Kerr [28].

The results of the ASSET/COPES/GRAPHICS system correlate well

	CPU Secs	Cost in $
OPTIMIZATION MODE (Results from COPES Only)		
Initialization Module	1.377	1.140
Cost Module	1.600	1.250
Space Module	1.858	1.360
GRAPHICS MODE (Graphical Output, 3-D PICTURE and 2 x 2-D SIMPLOT)		
Initialization Module	5.806	3.220
Cost Module	6.318	3.510
Space Module	5.694	

Table 9.11. Computer Time and Related Execution Cost for the ASSET/COPES/GRAPHICS Synthesis System.

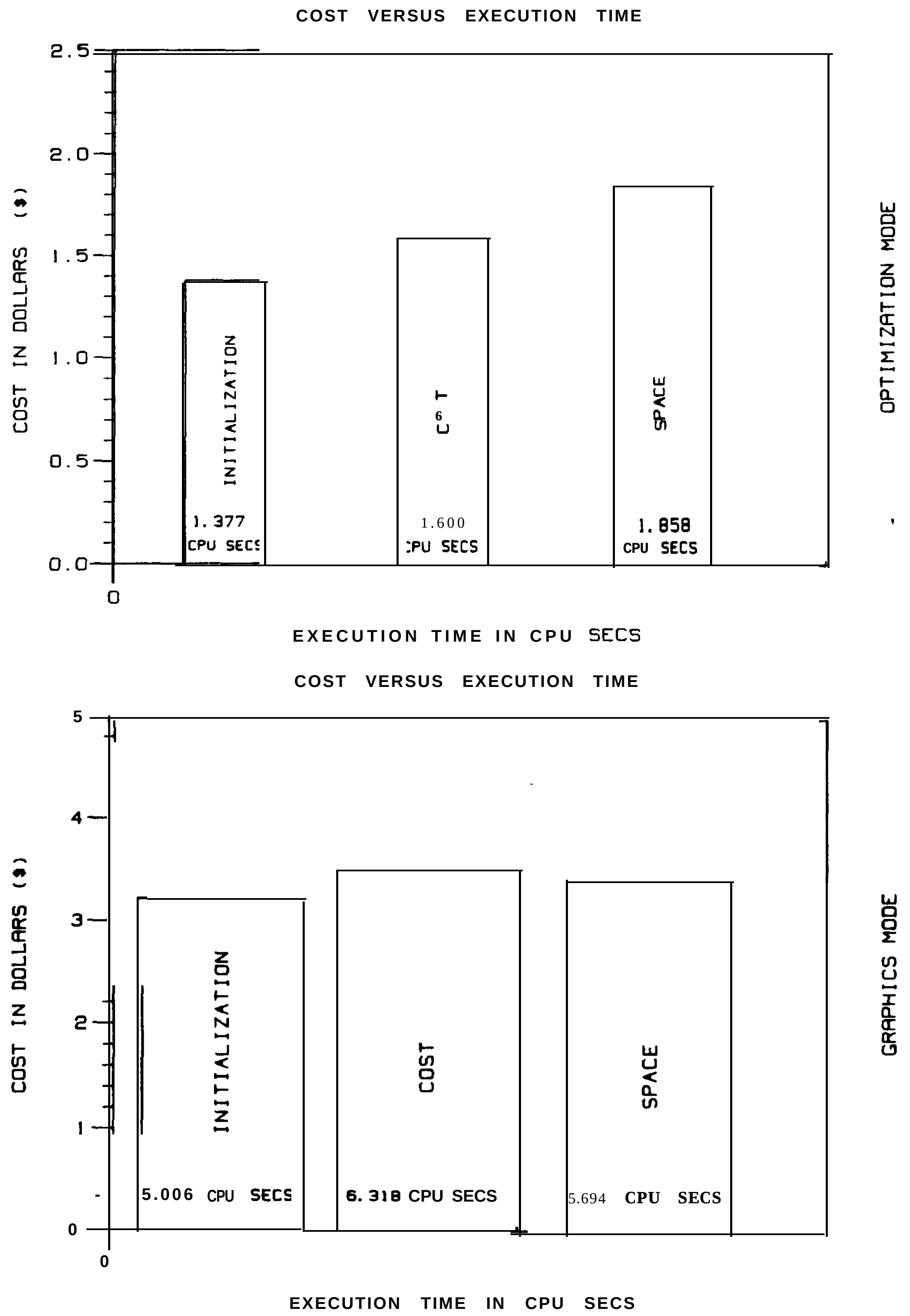


Fig. 9.8. CPU Execution Time and Cost for Each Mode and Module of the ASSET/COPES/GRAPHICS Synthesis System.

PARAMETER	ASSET/COPES MIN. WTFULL	ASSET/COPES MIN. SYSCPS	ASSET/COPES MAX. $V_{OPER.}$	REF. ~~7~~ 12	REF. ~~26~~ 16	REF. ~~26~~ 16	REF. ~~42~~ 28
VDESGN	29.000	29.000	29.000	29.000	28.560	30.830	29.000
VRANGE	20.000	20.000	20.000	20.000	20.000	20.000	20.000
WTMILP	327.400	327.400	327.400	327.400	327.400	327.400	327.000
LBP	389.360	391.280	390.720	408.000	394.390	418.000	408.000
B	44.570	43.490	43.420	44.815	45.030	43.060	45.200
T	13.690	13.760	13.740	14.350	13.540	15.450	14.350
D	28.630	28.770	28.730	30.000			30.000
LBP/B	8.736	8.998	8.998	9.104	8.760	9.710	9.000
LBP/D	13.599	13.600	13.599	13.600			13.600
B/T	3.255	3.160	3.160	3.123	3.330	2.790	3.150
CP	0.520	0.520	0.520	0.596	0.500	0.590	0.620
CX	0.900	0.839	0.835	0.749	0.770	0.750	0.750
WTFULL	3375.200	3407.900	3406.600	3718.260	3511.000	3575.000	3672.000
RANGE	6383.010	6412.350	6415.320	5577.540			
PHPREQ	40340.000	38270.000	38246.000	35068.000	39990.000	40004.000	40000.000
VOLTOT	457338.000	440145.000	437625.000	515447.000			531980.000
SYSCPS	807.200	808.660	806.500	819.500			
VPAYL/TOTVOL	0.229	0.231	0.231	0.225	0.200	0.200	0.199

Table 9.12. Comparison of Results from Alternative Studies.

with the real ship results within a tolerance of plus or minus five percent. This tolerance would be much less if more design variables had been chosen and more constraints had been imposed on the design space. It should also be emphasized that the results of this study represent results of the conceptual design phase and not the final design phase.

9.3.2. Optimizer Reliability

An example with the objective function being the minimization of the full load displacement was chosen to show the reliability of the system. Three different examples, with the same design variables and constraints, Tables 9.3 and 9.4, were run. All three start from a different initial point in the design space. The main purpose was to check to see if all three determine the same global optimum design point in the design space. This process is proposed as a general check of the final optimum design.

Table 9.13 shows the results of this study. The first column represents the results of the initial starting point in the design space with LBP equal to 300 feet, while the second column represents the final optimum results of this design. The following columns represent similar results but with different starting points, LBP = 400, 600 ft, in the design space. As can be seen, there is good convergence of the first two designs. Excellent repeatability of the optimum ship and a well defined ship are demonstrated.

The third design, with the initial the LBP equal to 600 feet, however, led to a different optimum design than the previous two. This design is a local optimum in the design space. Comparison with the previous two designs shows that they represent the global optimum, as the full load displacement is less than that of the third design.

PARAMETER	INITIAL 300'	OPTIMUM 300'	INITIAL 400'	OPTIMUM 400'	INITIAL 600'	OPTIMUM 600'	OPTIMUM 389.36'
VDESCN	29.000	29.000	29.000	29.000	29.000	29.000	29.000
VRANGE	20.000	20.000	20.000	20.000	20.000	20.000	20.000
DMISSN	30.000	30.000	30.000	30.000	30.000	30.000	30.000
WTMILP	327.400	327.400	327.400	327.400	327.400	327.400	327.400
LBP	300.000	389.360	400.000	390.890	600.000	446.980	390.740
LBP/B	9.000	8.737	8.700	8.436	9.000	9.292	8.439
CP	0.540	0.520	0.520	0.520	0.555	0.527	0.520
CX	0.900	0.900	0.900	0.887	0.900	0.788	0.887
T	10.550	13.690	14.070	13.750	21.100	15.720	13.740
LBP/D	13.599	13.599	13.600	13.600	13.600	13.598	13.600
RANGE	5490.530	6383.010	6378.650	6361.720	4632.760	6141.590	6361.530
WTFULL	2912.500	3375.200	3448.800	3410.400	5513.800	3823.600	3409.400
WTFUEL	569.600	569.600	569.600	569.600	569.600	569.600	569.600
PHPREQ	53244.000	40340.000	39724.000	41000.000	33886.000	32672.000	41000.000
VOLTOT	207702.000	457338.000	497932.000	476820.000	1689547.000	627772.000	476183.000
SYSCPS	796.700	807.200	810.100	807.900	900.300	841.700	808.600
VPAYL/TOTVOL	0.254	0.229	0.226	0.227	0.161	0.216	0.227

Table 9.13. Comparison of Results of the Optimization Mode to Indicate System Reliability

However, the results of the third design may be perfectly justified. The **optimizer** in this case followed a path which led it to a local optimum for two reasons:

- **a)** The four selected design variables represent a relatively low number of variables for the conceptual ship design.
- **b)** The imposed constraints should be increased in number to narrow more of the design space.

Additional design variables in this case could be the length to depth ratio, the draft to depth ratio, etc. Rigorous proof that the optimizer always converges to a global optimum is impossible.. For the reasonably well-behaved functions that have been used to model the ship design process and a sufficient number of design variables and constraints, a global optimum will usually be obtained.

All of the above designs started in the infeasible region and proceeded to a feasible one. This feature is of great importance to the designer, as numerous designs that a synthesis model would have rejected are simply tried and identified as infeasible on the path to a feasible design. The system maintains a record of all the designs tried which the designer **may** examine and use for future design decisions. The system also identifies those constraints and design variables that are active or violated, thus providing the designer with information on what is most critical in the design.

9.4. System Trade-Off Studies

One of the useful applications of the ASSET/COPES GRAPHICS system is the ability to conduct trade-off studies using the optimizing mode. The system allows these studies to be conducted with consistency and confidence as the program makes consistent routine decisions and calculations.

In this example, the influence of different propulsion plants on the ship leading particulars and cost were investigated. Three objective functions were used:

1. minimize of full load displacement
2. minimize of life cycle cost per ship
3. maximization of the military mission volume to total volume ratio.

The types of engines used were:

a) gas turbine, GT
b) combined gas turbine and steam turbine, COGAS
c) diesel engine, DIESEL.

Results for all three designs are given in Table 9.14. The GT engine gives, as expected, the minimum ship dimensions to carry the payload, while the COGAS engine results in a design very comparable to that of the GT engine. As expected, using a DIESEL engine requires a larger ship to support the payload. All three optimizations based on different objective functions gave very comparable results for each design case. Surprisingly enough, all three designs result in a very similar life cycle cost per ship. The fact that the the increase of the ship dimensions in the case of the diesel engine may balance the more expensive gas turbine engine is the probable explanation for this.

However, the military payload volume to total volume ratio is much less in the case of the diesel engine than in the other two cases. At this point,. the art of design is introduced. The designer would have to decide if a larger ship with lower operational volume ratio, or a smaller ship with higher operational volume, for the same life cycle cost,is desired..

PARAMETER	GT ENGINE	COGAS ENGINE	DIESEL ENGINE
VDEGSN	29.000	29.000	29.000
VRANGE	20.000	20.000	20.000
DMISSN	30.000	30.000	30.000
WTMILP	327.400	327.400	327.400
LBP	389.360	392.340	424.920
LBP/B	8.736	8.469	8.808
CP	0.520	0.520	0.536
CX	0.900	0.876	0.793
B	44.570	46.320	48.240
T	13.690	13.800	14.950
LBP/D	13.599	13.599	13.600
RANGE	6383.010	6336.080	7946.700
WTFULL	3375.200	3453.400	4349.900
WTFUEL	569.600	569.600	569.600
PHPREQ	40340.000	40940.000	41000.000
VOLTOT	457338.000	478312.000	575791.000
FUEL CONS.	11.206	11.124	13.951
GM/GMMIN	1.000	1.003	1.009
SYSCPS	807.200	810.600	807.000
VPAYL/TOTVOL	0.229	0.227	0.218

Table 9.14A. Minimization of the Pull Load Displacement for Different Propulsion Plants

PARAMETER	GT ENGINE	COGAS ENGINE	DIESEL ENGINE
VDESGN	29.000	29.000	29.000
VRANGE	20.000	20.000	20.000
DMISSN	30.000	30.000	30.000
WTMILP	327.400	327.400	327.400
LBP	391.280	393.720	424.180
LBP/B	8.998	8.713	8.846
CP	0.520	0.520	0.546
CX	0.839	0.809	0.787
B	43.490	45.190	47.950
T	13.760	13.850	14.920
LBP/D	13.600	13.600	13.599
RANGE	6412.350	6333.820	7795.120
WTFULL	3407.900	3487.300	4359.300
WTFUEL	569.600	569.600	569.600
PHPREQ	38270.000	38816.000	40750.000
VOLTOT	440145.000	457751.000	575278.000
FUEL COMS.	11.258	11.120	13.685
GM/GMMIN	1.000	1.000	1.000
SYSCPS	808.660	809.700	805.400
VPAYL/TOTVOL	0.231	0.229	0.217

Table 9.14B. Minimization of the Life Cycle Cost per Ship for Different Propulsion Plants

PARAMETER	GT ENGINE	COGAS ENGINE	DIESEL ENGINE
VDESGN	29.000	29.000	29.000
VRANGE	20.000	20.000	20.000
DMISSN	30.000	30.000	30.000
WTMILP	327.400	327.400	327.400
LBP	390.720	394.230	424.270
LBP/B	8.998	8.720	8.843
CP	0.520	0.520	0.544
CX	0.835	0.808	0.784
B	43.420	45.210	47.980
T	13.740	13.870	14.920
LBP/D	13.599	13.598	13.598
RANGE	6415.320	6333.470	7840.880
WTFULL	3406.600	3490.800	4358.300
WTFUEL	569.600	569.600	569.600
PHPREQ	38246.000	38748.000	40736.000
VOLTOT	437625.000	459040.000	573449.000
FUEL CONS.	11.263	11.119	13.766
GM/GMMIN	1.007	1.000	1.009
SYSCPS	806.500	808.900	806.800
VPAYL/TOTVOL	0.231	0.229	0.217

Table 9.14C. Minimization of the Military Mission Volume to Total Volume Ratio for Different Propulsion Plants

10. CONCLUSIONS

Automated design optimization in combination with computer graphics has been introduced into the conceptual ship design process. The output of **ASSET/COPES/GRAPHICS** synthesis system provides the naval architect with an efficient design tool with the capability of reducing the time required to perform feasibility and conceptual design studies. The ASSET/COPES/GRAPHICS system provides a rapid way of exploring all reasonable boundaries of dimensions and hull form in the initial stage of the design process. Its characteristics, which include:

a) flexibility

b) information obtained in the neighborhood of the optimum

c) easy selection of free variables and constraints without change of the design model and

d) visualization of the optimum situation through computer graphics,

make it a useful, successful, and essential tool in the designer's hands. It can assist the designer in selecting the most appropriate baseline ship for the design process, in estimating dimension and form changes needed to meet the operational requirements with minimum penalties, and in evaluating the results using the same standards of comparison among the alternative designs. Moreover, a greater number of design alternatives can be processed and compared as a result of the system's ability to start with an infeasible design and proceed to a feasible design. The designer is made aware of the design variables and constraints which are critical in each case and is able to alter those to achieve the goals.

Additionally, the traditional point design method has been combined with a graphical presentation of the constraints, design variables and objective functions in a specified design space.

The flexibility of the ASSET synthesis model permits the designer to update and review the data bank items or computational program algorithms as they become available, without affecting the organization of the system. The real value of the system lies in design situations where the designer faces radically new operational requirements and is uncertain of the choice of a basis ship to start with.

The combined system may be considered as a technique whereby a computer may be used in a flexible manner in a realistic iterative design process in which human judgment can continue to play a decision-making role. The latter means that the system must be used with intelligent caution. Optimization cannot be blindly applied to a problem and the results cannot be accepted at face value. Automated optimization analysis can carry out results which are as accurate as the analysis code on which the design is based.

11. RECOMMENDATIONS

The system development is far **from** being finalized. All of' the results obtained to date by this effort indicate that the new **synthesis system** Will be highly useful in conceptual design. A further development of this work would be the incorporation of the proposed technique into all the stages of the design process. Ship design can gain greatly from optimization techniques at every step of the design sequence. The incorporation of the **COPES/CONMIN** optimizer in an iterative fashion with the entire ASSET **synthesis** model is the logical extension of this study. Figure 11.1 illustrates the process by which the designer would be able to use each one of the computational programs coupled with the optimizer to perform detailed optimization at every step of the design process.

In conclusion, the roles the ASSET/COPES GRAPHICS system can play have been clarified and illustrated, and the importance of applying judgement in its use has been emphasized. As the need for rapid design of economical **systems grows** and as further development in computer technology and graphics capabilities continues, the techniques presented here will gain more potentialities in all disciplines of engineering. In the near future, the designer will be able to proceed through the entire synthesis, analysis, and optimization process from the initialization to the final level of design while sitting in front of a computer terminal, Figure 11.2.

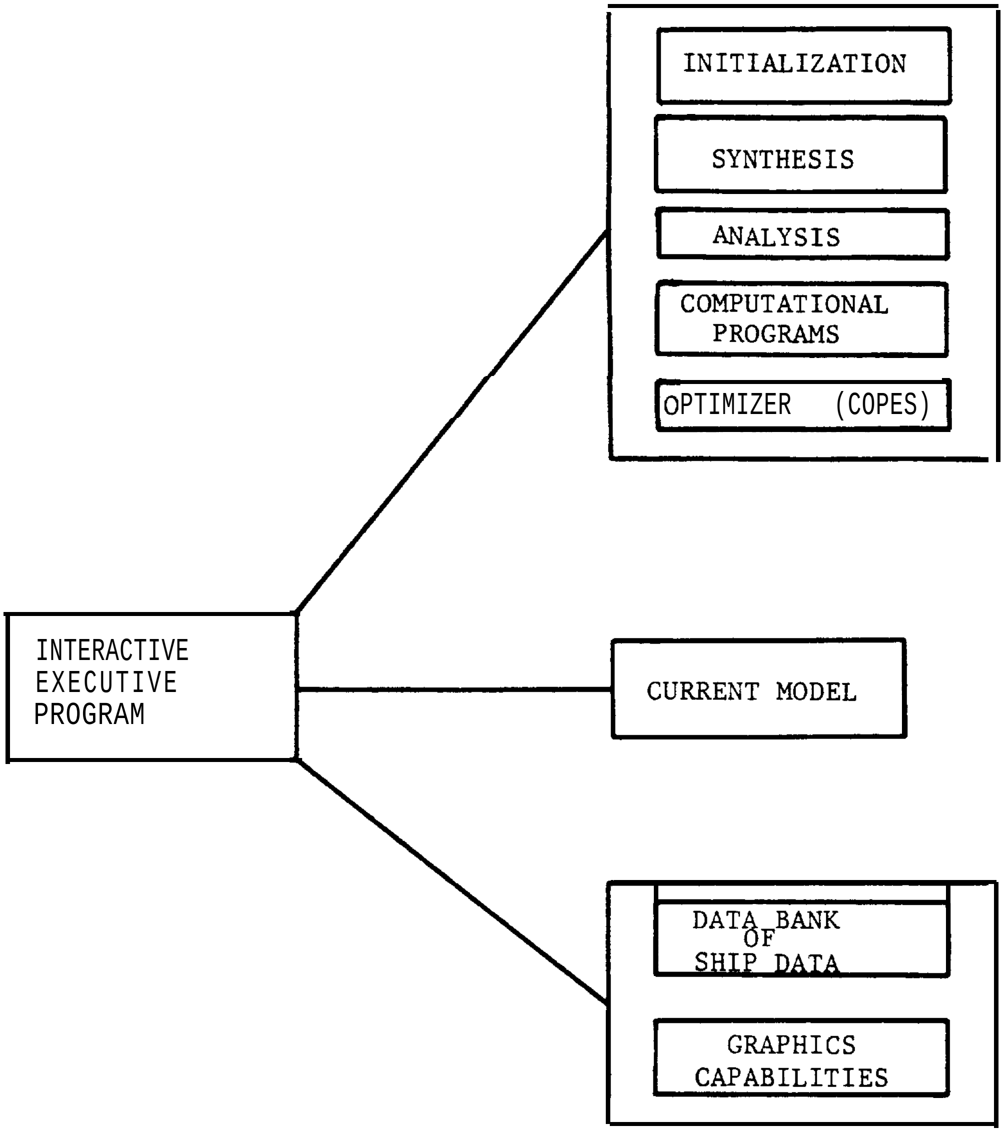


Fig. 11.1. Proposal for Future Extension of the Present Study

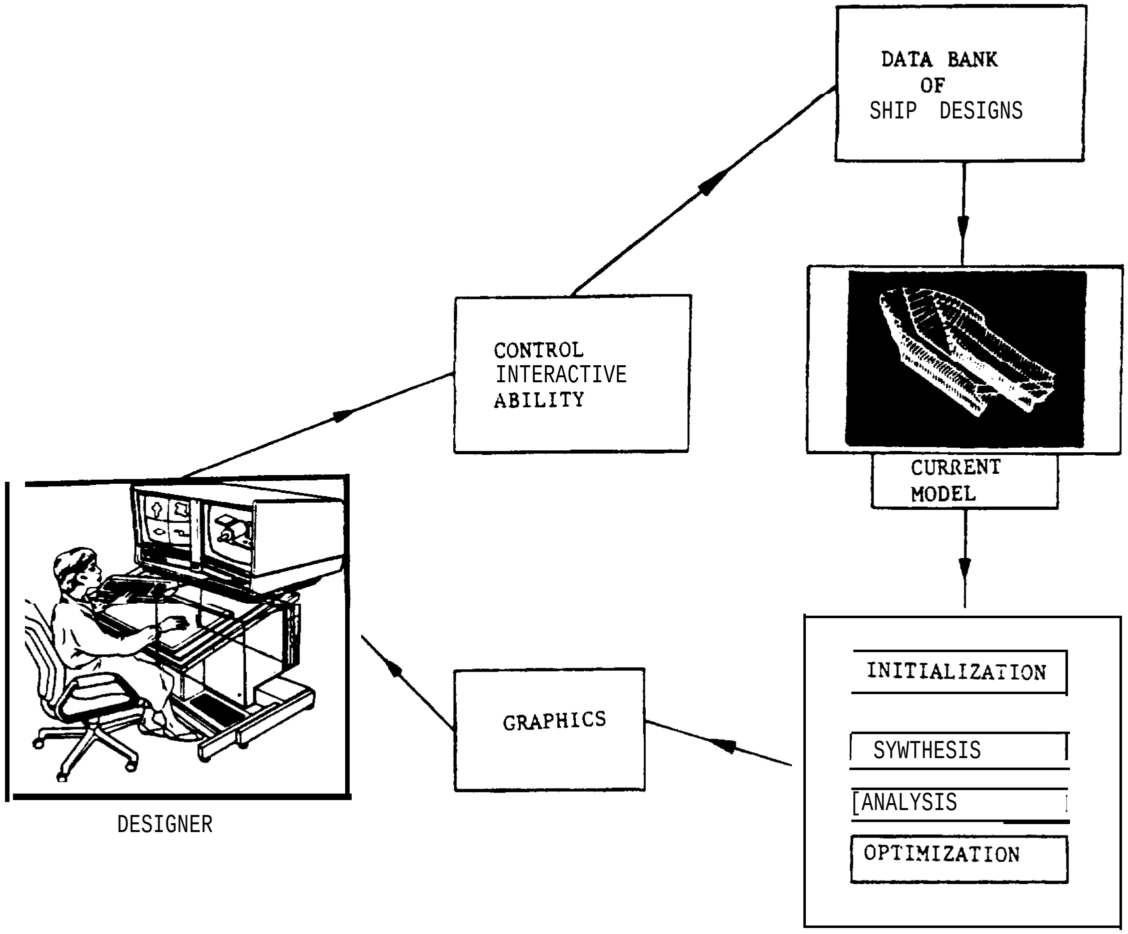


Fig. 11.2. Ship Design Using Optimization, and Interactive Computer Graphics

REFERENCES

1. Johnson, R. S., Fuller, A. C., Claffey, J. A., Bjorklund, F. R., and Thomson, B. M., Wan-Computer Graphics in Preliminary Ship Design," AIAA Paper No. 72-587, AIAA/SNAME/USN Advanced Marine Vehicles Meeting, July 17-19, 1972.
2. Rogers, D. F., Rodriguez, F., and Saterfield, S. G., "Computer Aided Ship Design and Numerically Controlled Production of Towing Tank Models," 16th Design Automation Conference, June 25-27, 1979.
3. Galloway, T. L., and Smith, M. R., "General Aviation Design Synthesis Utilizing Interactive Computer Graphics," SAE 760476, Business Aircraft Meeting, April 6-9, 1976.
4. Raymer D. P., "Developing an Aircraft Configuration Using a Minicomputer,' Astronautics and Aeronautics, November 1979.
5. Calkins, D. E., "A study in Computer-Aided Aerospace Vehicle Design," Master's Thesis, San Diego State University, 1969.
6. NAVSEA T0800-AA-RPT-010/CASDAC, "CASDAC, Abstracts of Computer Programs," Commander Naval Sea Systems Command, 1981.
7. Goodwin, Michael J., "Ship Synthesis Model for Coast Guard Cutters," Master's Thesis, Massachusetts Institute of Technology, June 1975.
8. Reed, Michael Robert, "Ship Synthesis Model for Naval Surface Ships," Master's Thesis, Massachusetts Institute of Technology, May 1976.
9. King, J. H., Devine- M. D., "HANDE - A Computer-Aided Design Approach for Hydrofoil Ships," Naval Engineers Journal, April 1981.
10. Hubble, E. N., "Program PHMOPT Planing Hull Feasibility Model User's Manual," DTNSRDC/SPD-0840-01, December 1978, revised

January 1981.

11. Aubble, E. N., "Performance Predictions for Planing Craft in a Seaway,' DTNSRDC/SPD-0840-02. September 1980.

12. Beyer C. F., Devine, M. D., Tsao, S. K., "ASSET-Advanced Surface Shin Evaluation Tool," Department of the Navy, DTNSRDC, Vol. 1, 2, 3A, 3B, 4, 5A, 5B, May 1982.

13. Murphy, R., Sabat, D. J , and Taylor R. J-, "Least Cost Ship Characteristics by Computer Techniques," Marine Technology, 1965.

14. Mandel. P., and Leopold, R., 'Optimization Methods Applied to Ship Design," Trans. SNAME, Vol. 74. 1966, pp. 477-521.

15 Eames. M. C. and Drummond, T. G., "Concept Expoloration--an Approach to Small Warship Design," Meeting of Royal Institution of Naval Architects, London, April 7, 1976.

16. Jenkins- James T., "Application of Optimization Techniques to Naval Surface Combatant Ship Synthesis," Master's Thesis, Naval Postgraduate School, Monterey, CA, March 1982.

17. Madsen, Leroy E., and Vanderplaats, Garret N., COPES - A FORTRAN Control Program for Engineering Synthesis, NPS69-81-003, Naval Postgraduate School, March 1982.

18. Vanderplatts, Garret N., "CONMIN - A FORTRAN Program for Constrained Function Minimization," TM X-62,282- 1973, NASA Ames Research Center, Moffett Field, CA.

19. PICTURE, Subroutine for Three-Dimensional Displays, User's Guide, Document # N358, Academic Computer Center, University of Washington, Seattle, June 1983.

20. SIMPLOT User's Manual, Document # N319, Academic Computer Center, University of Washington, Seattle, December 1982.

21. Calkins, D. E., "HYCAT: Hybrid Hydrofoil Catamaran Concept," AIAA 6th Marine Systems Conference, Seattle, September 1981.

22. Pantazopoulos M. S., "Ship Design Synthesis Utilizing Automated Optimization Techniques and Interactive Computer Graphics," 3rd Award in SNAME Pacific Northwest Section Student Paper Contest, Seattle, February 1983.

23. Saunders, Harold E., Hydrodynamics in Ship Design, Vol. I, The Society of Naval Architects and Marine Engineers, New York, 1957.

24. Nowacki, H., Brusis, F., and Swift, P. M., 'Tanker Preliminary Design--An Optimization Problem with Constraints,' Trans. SNAME, Vol. 78, 1970, pp. 357-380.

25. Leopold R., "Mathematical Methods of Optimization Applied to Ship Design," Massachussets Institute of Technology Report No. 65-15, 1965.

26. "Naval Gun Mounts and Guided Missile Launching Systems,. FMC Corp., Northern Ordnance Division, Minneapolis, MN.

27. Jane's Weapon Systems, 9th Ed., Franklin Watts, Inc., New York, N.Y., 1978.

28. Garzke, W. A., and Kerr, G., "Major Factors in Frigate Design," Trans. SNAME, Vol. 89, 1981, pp. 179-210.