SHIP DESIGN OPTIMIZATION USING ASSET

By

Swaroop N. Neti

Thesis submitted to the faculty of the Virginia Polytechnic Institute and State
University in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

in

Ocean Engineering

Committee:
Dr. Wayne Neu, Chairman,
Dr. Alan J. Brown
Dr. Rakesh K. Kapania

February 10, 2005

Blacksburg, Virginia

Keywords: Design optimization, ASSET, Model Center, Process integration.

SHIP DESIGN OPTIMIZATION USING ASSET

By

Swaroop N. Neti

ABSTRACT

This thesis describes the design optimization of two different types of vessels. They are LHA(R), a replacement for the US Navy amphibious assault ship and DDG51, a destroyer class vessel. The overall measure of effectiveness (OMOE) and the lead ship acquisition cost (LCA) are considered to be the objective functions. The evaluation of feasibility of the designs and various ship parameter calculations are performed using the US Navy ship design evaluation software ASSET. ASSET is integrated with the design optimization software DARWIN to obtain results representing the best designs over a range of LCA. Model Center software is used to integrate the processes ASSET and Darwin.

The results generated will provide the owner with the best designs possible (designs with high OMOE) over a range of LCA. This thesis is mainly of academic interest. The results generated could help the owners to look at various design options available for the amount of money they are willing to spend.

Acknowledgements

I would like to express my sincere gratitude to my advisor, Dr. Wayne Neu for his guidance and advice. Completing this thesis would not have been possible without Dr. Neu's support.

I would like to thank Dr. Brown, for allowing me to attend his ocean design class and to use his programs and also for spending time to answer all my questions.

I would like to thank my committee member Dr.Kapania for his support and for his suggestions.

I would like to thank Mr. Todd Heidenreich for his prompt response to all my questions related to ASSET.

I would like to thank the people at Phoenix integration for their support.

It is my parent's love that gives me the strength to perform any task. It is because of their encouragement, their confidence and their prayers I am able to get my second masters degree today.

TABLE OF CONTENTS

APPENDIX

LIST OF FIGURES

LIST OF TABLES

Chapter 1

INTRODUCTION

When designing a ship the naval architect uses, the owner's requirements, the information available for similar type of vessels built earlier and his ability to see the requirements in the future as guide lines for the design process. A ship concept design produced in this manner may be feasible enough to satisfy the owner's requirements but may not be the best possible design for the amount of money the owner is going to spend.

Naval Sea Systems Command (NAVSEA) initiated the design, acquisition and construction (DAC) project to apply rigorous process analysis to naval acquisition process by optimizing ship performance, cutting the acquisition cost and reducing the design cycle time. This formed the basis for a systematic approach to naval ship design [1]. This structured search of designs for designs of high effectiveness becomes difficult when a large number of designs are to be evaluated in a non-linear, discontinuous, constrained design space [2]. Multi-attribute value theory and analytical hierarchy process were used to synthesize an effectiveness function [2]. The multi-objective optimization methodology for the naval ship concept design was developed using genetic algorithms [3]. This thesis implements this multi-objective optimization methodology for the optimization of two types of naval vessels. They are LHA(R), the replacement for the US Navy amphibious assault ship, and DDG51, the guided missile destroyer ship. In both the problems we tried to find designs that have high overall measure of effectiveness (OMOE) and low lead ship acquisition cost (LCA). Ship design optimization is the process of finding the feasible designs, which will lead to ships that will be more effective in performing their objectives and yet have lowest possible building cost. Ship design optimization involves three major steps. They are – process of generating designs, evaluating the feasibility of the generated design, evaluating the effectiveness and cost of the feasible design.

Advanced Surface Ship Evaluation Tool (ASSET) is the software provided by the Naval Surface Warfare Center, Carderock Division (NSWCCD) and is used for the evaluation of ship designs. It determines whether a particular design is feasible and in that process makes changes to various characteristics of the ship to arrive at a balanced design.

Darwin is an optimization tool developed by Phoenix integration. It is a genetic algorithm (GA) based optimization tool that generates new sets of designs based on the results produced using the previous sets of designs.

Model Center (MC) is the process integration software developed by Phoenix integration and is used to integrate the ship design evaluation process of ASSET with the optimization process of Darwin.

Analysis server software also developed by Phoenix integration is used for creating file wrappers which can be used in MC to get access to parameters produced by analysis modules in ASSET.

ASSET, Darwin and Model Center are put together to form the optimization system. Darwin generates designs and sends them to ASSET using the methods provided in the Model Center process integration environment. ASSET determines the feasibility of the design. The Model Center components determine the OMOE and the LCA of the design using the characteristics of the design calculated by ASSET. These values are sent to Darwin. Darwin uses these values to generate better designs. The process will run until a user specified number of designs were evaluated.

Chapter 2

OPTIMIZATION

Optimization is the process of finding the best alternative from a set of feasible options to maximize or minimize a function called the objective function. The variables that the user can get access to and change the value of the objective function by changing their values are called design variables. The user can change the value of the design variable by selecting different alternatives from the set of available alternatives for that design variable. A combination of design variables formed by selecting one option for each of the design variables from their sets of viable options is called a design. Design optimization of any system can be considered as a combination of design and analysis of the system [4]. Designing a system is the process of producing a new design and analysis is the process of determining the effectiveness of the design. While designing a system we may need to satisfy a set of conditions called constraints.

Depending on the number of objective functions and the number of constraints, optimization problems can be divided into four types. If we have only one objective function and no constraints to satisfy then it is called a single objective unconstrained optimization problem. If we have more than one objective function and no constraints then it is called multi-objective unconstrained optimization problem. If we have constraints in our problem then the problem will be either single or multi-objective constrained optimization problem.

Considering the number of design variables and the number of alternatives available to each of them, the number of designs that can be produced can be a very large number. If we want to arrive at the best set of designs by evaluating the objective functions of all these designs it takes a lot of time and it is not efficient. To solve this problem we need an algorithm that can see what combinations of design variables are giving better objective function values and generate designs that are better compared to the previous designs.

If we represent design variables along axes orthogonal to each other then the space formed by these axes is called design space. Any point selected in the design space is called a design point. We have no idea of the design space and where good designs are going to be. Without any initial idea of the design space we cannot expect an algorithm to find better designs unless it can learn on its own. Genetic algorithms (GA) have the capability to learn on their own and are best suited for the kind of problems where the design variables are discrete [5]. Hence GA is selected for our optimization problem.

GENETIC ALGORITHMS

Genetic algorithms are based on the theory of evolution. They simulate the evolution process in generating new designs. Genetic algorithms learn how to move towards better designs based on the results from the previous design evaluations. Hence genetic algorithms do not need any prior knowledge of the optimization process.

To simulate the evolution process, genetic algorithms use three methods called crossover, mutation and selection. Better designs are arrived at by combining successful features of the existing designs.

In GA terminology, a design point, formed by the combination of design variables in the design space, is called a candidate solution. The set of candidate solutions, that we start the optimization process with, represents the first generation. The set of candidate solutions produced in each generation is called a population and the number of candidate solutions in each generation is called population size. The number of preserved designs tells how many designs we want to carry, from the existing best designs, to the next generation.

The first generation (parent) is produced by selecting the candidate solutions randomly over the design space. The objective function values for the whole population are evaluated and submitted to the genetic algorithm. Learning from these results, GA arrives at the next generation (child) by keeping some of the best designs obtained in the previous generation and the remaining designs are obtained by applying crossover and mutation to the designs obtained in the previous generation [6]. The objective function values are obtained for the designs in both the generations. The ability of a candidate solution to survive and occur in the next generation is called its fitness. Candidate solutions with high fitness exist for many generations. Here the fitness function is the objective function. In the case of a multi-objective optimization problem the fitness of a candidate solution is related to all the objective functions. Once the fitness values are known we sort both parent and child populations according to their fitness values. We can find the designs that will survive for the next generation in two ways. In the first method we replace the worst designs from the child population using the best designs in the parent and this is called elitist selection. In the other method we combine both the parent and child populations and sort them according to the fitness values. The designs with the best fitness values are selected and this is called the multiple elitist method of selection.

Using either the elitist or multiple elitist method of selection we select a number of designs equal to the number of preserved designs. The rest of the population in the new generation is obtained using crossover and mutation processes discussed below. We provide the probability of application for both crossover and mutation at the start of the optimization process. The application of crossover or mutation is determined by comparing their probability with a randomly generated probability value. If the probability of crossover or mutation is less than its randomly generated probability value then that operator (crossover/ mutation) is applied. Let us see the crossover and mutation processes.

<u>Crossover:</u> Let us suppose that there are two designs having six design variables. To arrive at the next generation design from these two designs using crossover we split the two parent designs. The point of splitting is chosen randomly. Let us say that we split them into two halves. New generations are produced by combining one part from each of the designs.

Ex: Let design A be 4, 2, 3, 3, 2, 1 and design B be 3, 4, 2, 2, 2, 1, where the numbers represent the option chosen for each design variable (assuming that various options are available for each design variable).

By splitting each of the designs into two halves we get 4, 2, 3 - 3, 2, 1 and 3, 4, 2 – 2, 2, 1. To arrive at new designs using crossover we combine different parts of the existing designs. Hence we get the new designs 4, 2, 3, 2, 2, 1 and 3, 4, 2, 3, 2, 1.

In this process we are combining the existing designs and this restricts us to the part of the design space where these parent designs are present. Hence if we use only crossover we reach the point of convergence very quickly which is a local convergence point. We reached this point without searching the other parts of the design space. This prevents us from finding better designs existing in those regions and also prevents us from arriving at the global convergence point. To go to a different region on the design space we need to make sure that the design variables in different parts of the design space are selected.

Mutation: Mutation is used to introduce designs in the other parts of the design space. Mutation is applied after crossover and with a lower probability. In mutation the point of application is chosen randomly and at that point the design variable value is selected randomly from the available options.

In crossover we generate new designs by combining different parts of existing designs. In this process we can generate only those designs that are formed by the design variable values that are available in the existing designs. We cannot generate any designs having some other values of design variables. For example, in the discussion of crossover we have seen that 4, 2, 3, 3, 2, 1 and 3, 4, 2, 2, 2, 1 can produce designs 4, 2, 3, 2, 2, 1 and 3, 4, 2, 3, 2, 1. But we cannot generate a design that looks like 4, 2, 3, 2, 5, 1. We can arrive at this design with mutation.

Ex: Let the initial design be 4, 2, 3, 2, 2, 1. If the random point of application of mutation is 5 then we select the fifth design variable value from the available options. If the available options are 1,2,3,4,5,6 and if we select 5 as the random option then the new design will be 4, 2, 3, 2, 5, 1. Thus mutation allows us to move through the design space and thus prevents premature convergence.

Thus we arrive at the new generation of designs using genetic algorithms. This leads us to designs which form a non dominated frontier. The process of calculating the fitness values for the designs, selection and generation of new population of designs is repeated until the GA can improve the non dominated frontier or until an initially set number of generations are reached. The final result will be a set of globally non dominated designs.

The next section describes the genetic algorithm based optimization software Darwin.

DARWIN OPTIMIZER

The optimization tool used in this project is called Darwin and it is based on genetic algorithms [6]. Figure 1 shows the graphical user interface (GUI) of Darwin.

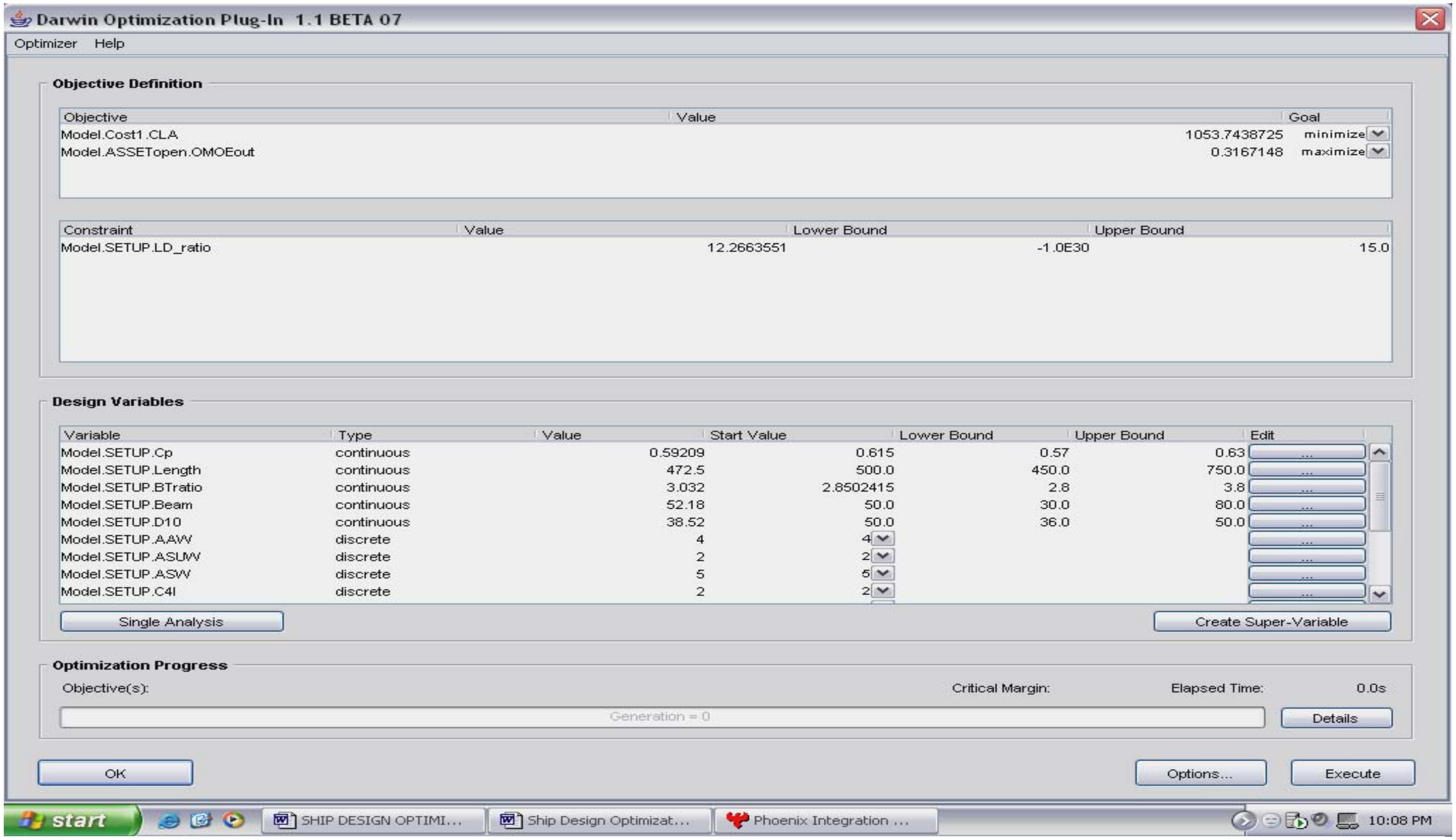
Darwin can perform single as well as multi-objective optimization problems. In Figure 1 we can see the provisions in Darwin to specify Objective functions, Design variables and constraints. We can also see that we can specify whether we want to maximize or minimize the objective function, the range for continuous design variables and the alternatives available for discrete design variables.

The options button of this interface takes us to the genetic algorithm parameter selection GUI of Darwin, shown in Figure 2.

In Figure 2 we can see that Darwin allows the optimization parameters to be selected either manually or automatically. When we select the parameters manually we need to specify the values of population size, selection scheme (elitist / multiple elitist), the seed value for the random number generation, maximum number of generations we want to perform, when to say the process is converged (After a fixed number of generations or after reaching a certain number of generations without improvement), crossover and mutation probabilities. We can also use the automatic selection option which determines the values of all the above parameters based on the number of design variables and the number of options available to each of them. With the initial population being generated randomly and all the optimization parameters being set Darwin arrives at the next generation by applying selection, crossover and mutation on the initial population.

When we use the option *use memory*, Darwin remembers all the designs that are evaluated and whenever it gets a new design it checks whether that design has already been generated. Thus it prevents re-calculation of the values related to the same designs.

For a dual objective optimization problem the results from Darwin will be represented in the form of a Pareto curve. A design point X is said to be a Pareto point if and only if there is no other point in the design space that has an improvement in any of the objective function values without a decrement in some other objective function value. The curve joining all the Pareto points is called a Pareto curve.



Objective	Value	Goal
Model.Cost1.CLA	1053.7438725	minimize
Model.ASSETopen.OMOEout	0.3167148	maximize

Constraint	Value	Lower Bound	Upper Bound
Model.SETUP.LD_ratio	12.2663551	-1.0E30	15.0

Variable	Type	Value	Start Value	Lower Bound	Upper Bound	Edit
Model.SETUP.Cp	continuous	0.59209	0.615	0.57	0.63	...
Model.SETUP.Length	continuous	472.5	500.0	450.0	750.0	...
Model.SETUP.BTratio	continuous	3.032	2.8502415	2.8	3.8	...
Model.SETUP.Beam	continuous	52.18	50.0	30.0	80.0	...
Model.SETUP.D10	continuous	38.52	50.0	36.0	50.0	...
Model.SETUP.AAW	discrete	4	4			...
Model.SETUP.ASUW	discrete	2	2			...
Model.SETUP.ASW	discrete	5	5			...
Model.SETUP.C4I	discrete	2	2			...

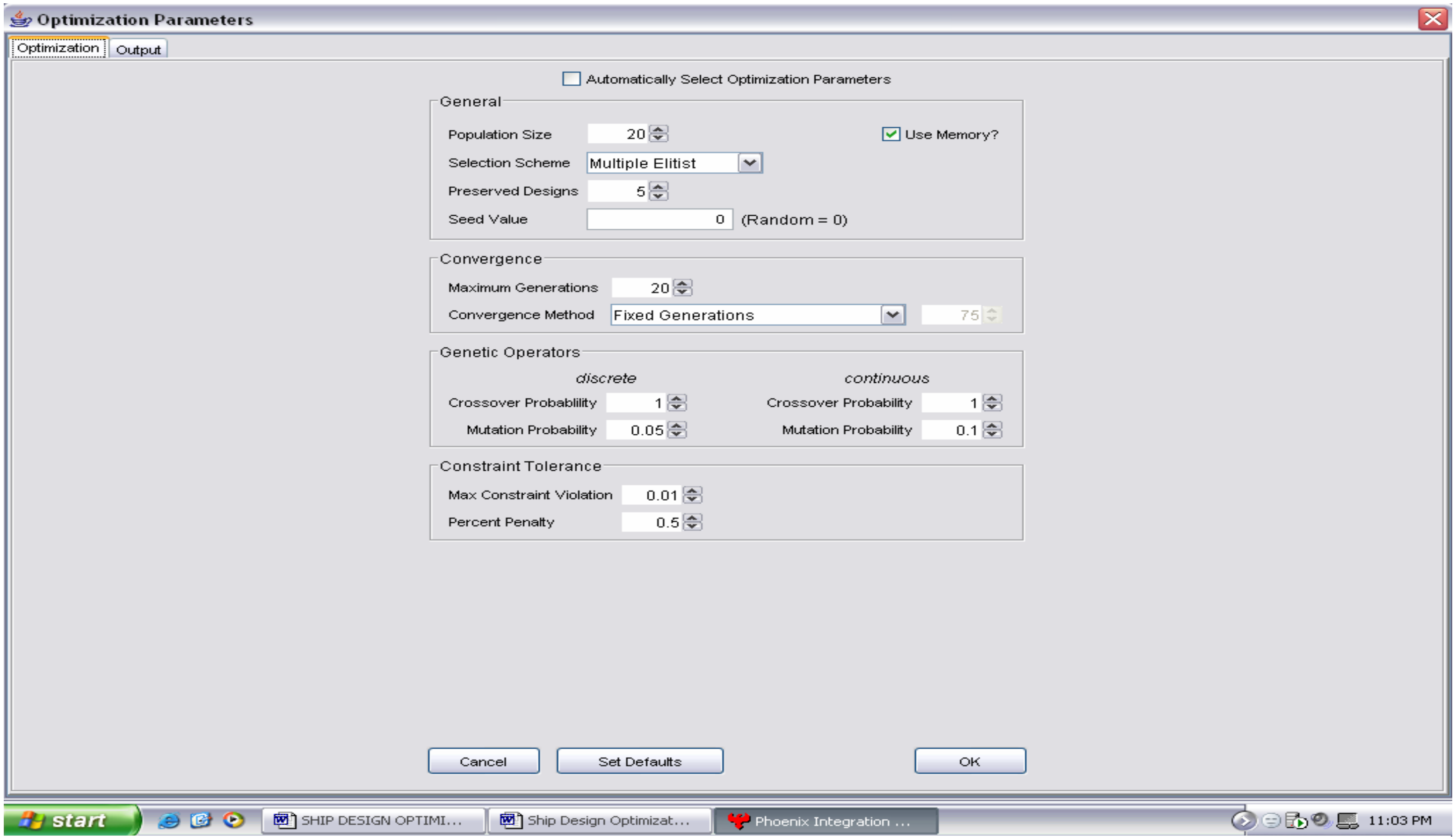
Objective(s):

Critical Margin:

Elapsed Time: 0.0s

Generation = 0

Figure 1. Graphical User Interface (GUI) of Darwin



☐ Automatically Select Optimization Parameters

General

Population Size 20

☑ Use Memory?

Selection Scheme Multiple Elitist

Preserved Designs 5

Seed Value 0 (Random = 0)

Convergence

Maximum Generations 20

Convergence Method Fixed Generations 75

Genetic Operators

discrete		*continuous*	
Crossover Probablility	1	Crossover Probability	1
Mutation Probability	0.05	Mutation Probability	0.1

Constraint Tolerance

Max Constraint Violation 0.01

Percent Penalty 0.5

Figure 2. Genetic Algorithm parameter selection GUI of Darwin

Chapter 3

ADVANCED SURFACE SHIP EVALUATION TOOL (ASSET)

ASSET is a family of computer programs used to evaluate the feasibility of several types of surface ships. It can determine the feasibility of mono hull surface combatants (MONOSC), mono hull carrier vehicles (MONOCV), mono hull amphibious ships (MONOLA). ASSET is an interactive program and prompts us to select the type of ship that we want to analyze [7].

In ASSET the ship designs are stored in databanks. Each databank contains the information about various parts of a ship. To manage this huge amount of data, ASSET uses a multi level, tree type hierarchy. The primary components of a ship like propulsion plant, electric plant, hull, etc. form the top level. Each of these primary components will be divided into secondary components like hull form, hull sub division and hull structure for hull. Each of these secondary components will further be divided into tertiary components. So as we go down the hierarchy we will be going towards greater details of that component. This division will go on until we reach the level where we have parameters that contain data about the physical characteristics of the ship model. The list of all these parameters is called model parameter list (MPL).

We can edit the databank in ASSET using the edit tool, using the command line and using a wizard. Using the edit tool we can go to the particular parameter and change its value manually. Wizards give the option to modify all the necessary parameters of a particular component of the ship manually. Using command line we can send commands to ASSET and change values in the databank. When we have a big, continuous block of data (data that occupies contiguous blocks in the MPL of that ship) that we want to attach to the databank the command line method will be good. This is because ASSET provides the facility by which we can store the continuous block of data as a component. When we want to edit the databank, instead of sending individual commands for each of these parameters, we can send a single command asking ASSET to use the stored component. This reduces the amount of time and work required to enter each parameter. Various parts of a ship are divided into seven weight groups, each having three levels. The sum of weights of all third level components will be the weight of second level group. The sum of all the second level groups will be the weight of the component at the top level. The sum of weights of all the components at the top level is the ship weight. This is called the ship work breakdown structure (SWBS) [7].

The seven top level groups are hull structure, propulsion plant, electric plant, command and surveillance systems, auxiliary systems, outfit and furnishings, and armament. Let us say W100, W200, …, W700 are the weights of these seven top level structures and if the weights of margins and loads are WM00, WF00 then the full load weight of the ship (WFL) is given by WFL=W100+W200+W300+W400+W500+W600+W700+WM00+WF00.

ASSET has two kinds of modules called computational and I/O (Input/Output) support. Computational modules can be divided into Synthesis modules and Analysis modules. The feasibility of a design is evaluated by checking its feasibility for the synthesis modules. Following is the list of synthesis modules in ASSET.

1. Hull Geometry module
2. Hull Subdivision module
3. Aviation Support module (for MONOCV only)

4. Deck house module
5. Hull Structure module
6. Appendage module
7. Resistance module
8. Propeller module
9. Machinery module
10. Auxiliary Systems module (for MONOSC only)
11. Weight module
12. Space module
13. Design summary module

Synthesis modules were always run in the above sequence. While running each ASSET module lot of calculations are performed and many parameters in ASSET will be calculated. When all the modules in ASSET run without producing any error and when all the parameters in ASSET converge we get a feasible design. Convergence checking is necessary because the first time ASSET executes its synthesis modules it uses default values to parameters, that are required in the calculations but does not have a value assigned to them and when the synthesis modules in ASSET are run again it uses the results generated in the previous iterations and estimates the values of parameters provided with default values in the previous generation. Thus the parameters in ASSET are modified in every iteration through the synthesis modules. When the difference between the parameter values from two consecutive iterations is less than the allowed difference, called tolerance, we say that the parameter has converged.

ASSET involves thousands of parameters and convergence of all these parameters takes a lot of time. To avoid this we need to find the parameters whose change affects the objective function values and check the convergence of those parameters.

Analysis modules in ASSET are used to calculate the performance characteristics of a feasible design. ASSET shows its results in the form of printed and graphics reports. The parameters calculated by synthesis modules will be stored in the databank while the results generated by analysis modules will be displayed in the form of printed reports only. Hence the optimization process cannot get direct access to the parameters calculated by analysis modules.

I/O modules are used for input and output of data in ASSET. The Hull Generation module gives the user more control in generating the molded hull form. The Export module gives the user the power to convert the ship data in ASSET to the format wanted by other programs.

ASSET is a highly interactive program. It requires input from the user during many of its calculations. It is possible for the user to provide this information only if the number of designs that are evaluated is small. But in our optimization problem we are going to evaluate hundreds of designs and with user interaction this process will take a very long time. We need to find a way to automate the process. In the next chapter, we discuss Model Center which solves the problem.

Chapter 4

MODEL CENTER

Model Center is a process integration environment. It provides the functionality to link different processes together and to schedule the process runs so that the linked processes can follow a definite path of execution. These linked processes with a scheduled way of execution form the model of the system that we want to generate.

Figure 3 shows the model of the system that we are going to generate for the ship design optimization problem. The arrows represent the sequence of execution.

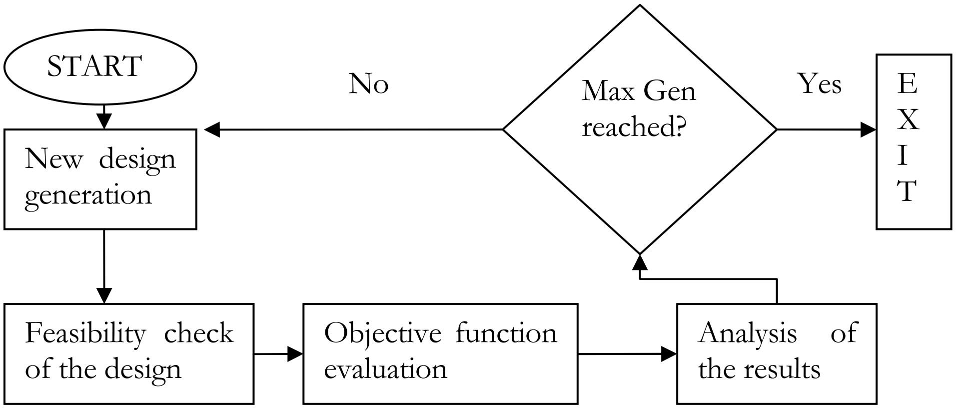


Figure 3. Ship design optimization model

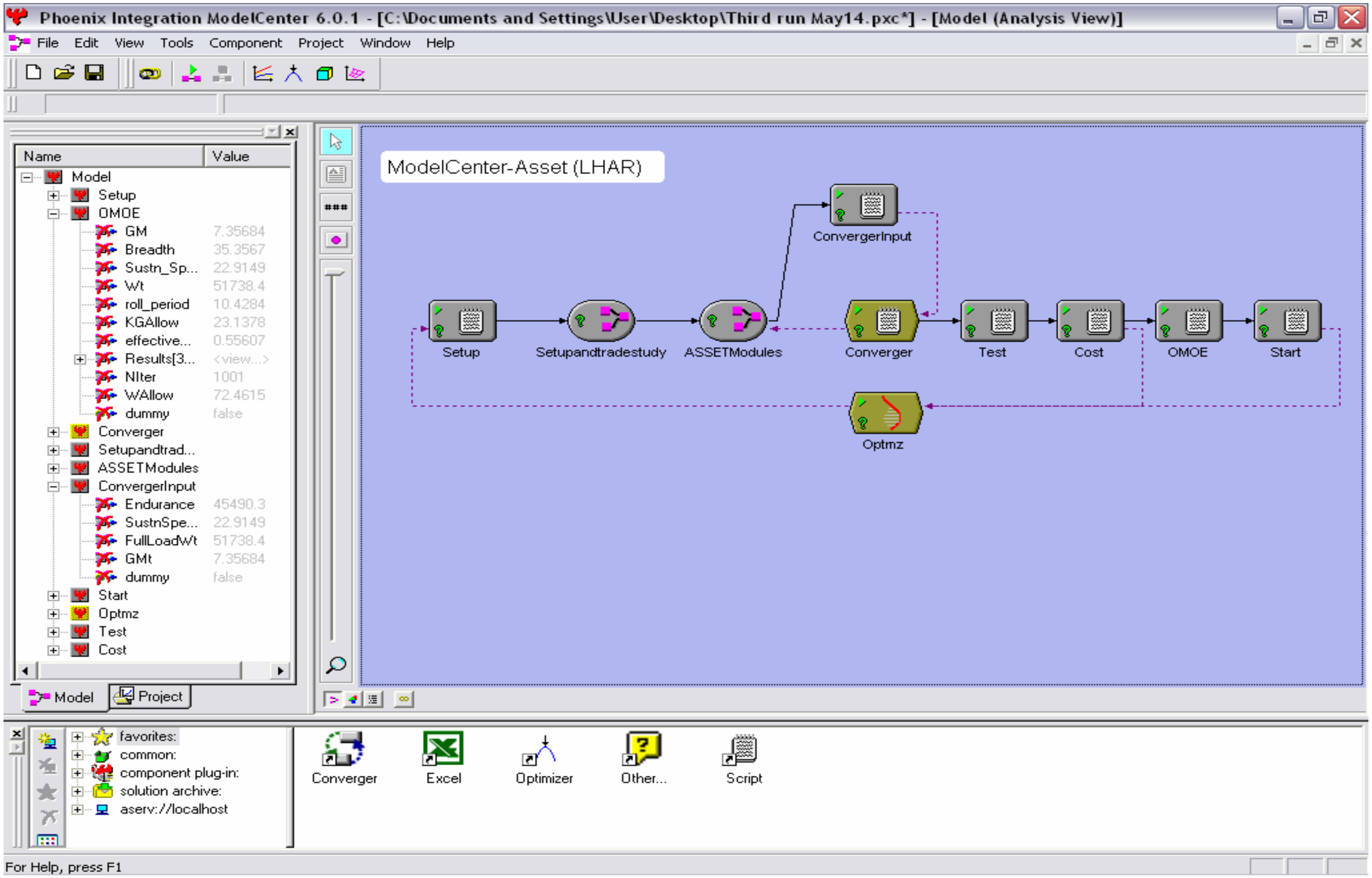
In Model center the processes are represented by components. A component can be thought of as a black box that takes input and, following the scheduling instructions – when to run and how to run, generates the output.

Different kinds of components in MC are analysis components, assembly components, geometry components, and driver components [8]. All these components can be created using the script component editor. The script component editor in MC has two segments. One for defining input and output variables and the other for writing the script that does all the required calculations. While defining a variable we can define its properties such as units, type (Integer, Real, Array, etc.), description, lower bound, upper bound, etc. The scripting language used in this project is VBScript.

Once we have the components representing all the processes of the system we need to connect them so that they can talk to each other (share the data). The link editor is activated by dragging a line from one component to another. The link editor in Model Center provides a means to link the components. A link is formed between two components when the output from one component is linked to the input of another component. Links can be created, suspended and broken using link editor. Once all the components in the model are linked in the required fashion the model is ready to run. The order of running the components is determined by the scheduler.

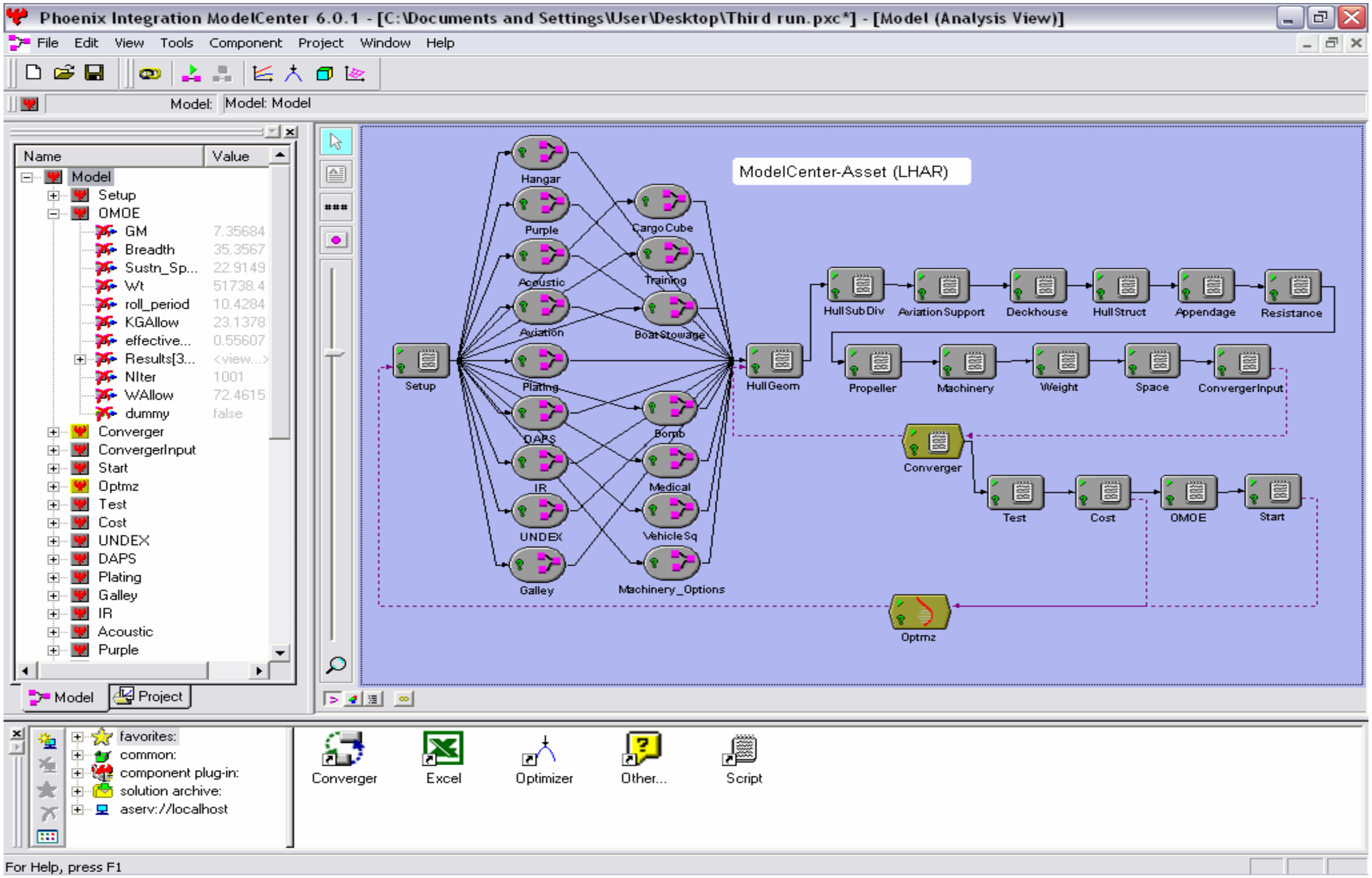
The scheduler is the part of Model Center that is responsible for knowing which components need to be run and when. Model Center supports different schedulers. They are backward scheduler, forward scheduler, mixed mode scheduler, script scheduler. We used forward scheduler in our models. In forward scheduling as soon as an input value of a component is changed, all downstream components are run in the sequence in which they are linked.

The Model Center model of LHA(R) is shown in Figure 4 and Figure 5. The model is formed using script components and assembly components of Model Center and the DARWIN optimizer component.



Name	Value
Model	
Setup	
OMOE	
GM	7.35684
Breadth	35.3567
Sustn_Sp...	22.9149
Wt	51738.4
roll_period	10.4284
KGAllow	23.1378
effective...	0.55607
Results[3...	<view...>
NIter	1001
WAllow	72.4615
dummy	false
Converger	
Setupandtrad...	
ASSETModules	
ConvergerInput	
Endurance	45490.3
SustnSpe...	22.9149
FullLoadWt	51738.4
GMt	7.35684
dummy	false
Start	
Optmz	
Test	
Cost	

Figure 4. Screen shot showing the LHAR model in Model Center.



Name	Value
Model	
Setup	
OMOE	
GM	7.35684
Breadth	35.3567
Sustn_Sp...	22.9149
Wt	51738.4
roll_period	10.4284
KGAllow	23.1378
effective...	0.55607
Results[3...	<view...>
NIter	1001
WAllow	72.4615
dummy	false
Converger	
ConvergerInput	
Start	
Optmz	
Test	
Cost	
UNDEX	
DAPS	
Plating	
Galley	
IR	
Acoustic	
Purple	

Figure 5. Screen shot showing all the trade study options and Asset modules of LHAR model in Model Center.

The components “Setup”, “ConvergerInput”, “Converger”, “Test”, “Cost”, “OMOE”, “Start” are script components. The components “Setupandtradestudy”, “ASSETModules” represent the assembly components. A group of script components together form these assembly components. The script components “HullGeom” to “Space” in Figure 5 are the contents of “ASSETModules” assembly component of Figure 4. The arrows in Figure 4 and Figure 5 represent the links between the components and the direction of the arrows represents the direction of the data flow between those components. The “Optmz” component represents the DARWIN optimizer component which generates new design parameters and analyses the objective function values of those designs.

These new design parameters are to be sent to ASSET, an external program to Model Center, to form the new design. Instructions should be sent to ASSET to analyze the new design. These instructions are written in the script components of Model Center. Thus ASSET process is wrapped using the script components in Model Center. The scripts or instructions written to interact with ASSET are discussed in the next section. The scripts for various ASSET related functions are discussed in APPENDIX section SCRIPTS - ASSET. The LHA(R) and the DDG51 models are discussed in chapters 6 and 7 respectively.

INTEGRATING ASSET WITH MODEL CENTER

ASSET is invoked by running the executable “assetwui.exe”. Hence to access ASSET we need to get access to “assetwui.exe”. For this we need to establish a connection to ASSET through an object that allows us to use its services and interact with ASSET. Here is how it is done.

```
Dim assetExecutive
Set assetExecutive = CreateObject ("Asset.Executive")
```

The order of execution and the set of modules associated with the ship vary with ship type in ASSET. We can get access to the command line through the ship type object only. Hence we need to establish a connection to the ship type object. Here is how it is done and how we get access to the command line.

```
Dim assetShipType
Set assetShipType    = assetExecutive.GetShipType
Dim assetCommands
Set assetCommands = assetShipType.GetCommands
```

Now that we have access to the command line of ASSET we can manipulate any parameters in ASSET using the SendCommand method supported by assetCommands object. In that command we should specify the ASSET variable name or array name that we want to modify and the row and column of the databank into which we want these values to go. Here is how it is done.

```
assetCommands.SendCommand "SET, P+A SWBS KEY TBL"
assetCommands.SendCommand "C(167,1)W241"
assetCommands.SendCommand "C(169,1)W244"
assetCommands.SendCommand "Q"
```

As we discussed in Chapter 3 if we want to use a component we can do it with just one command.

```
assetCommands.SendCommand "USE, B COMP, MACHY TRUNK X LOC ARRAY, THRU, MACHY TRUNK HORZ LOC TBL"
```

In the above command we are telling ASSET to use the component called BCOMP and use the portion of its data between MACHY TRUNK X LOC ARRAY and MACHY TRUNK HORZ LOC TBL.

When all the parameters representing the new design are applied to the baseline ship the synthesis modules in ASSET are run to evaluate the feasibility of the design. A script component to run each module in ASSET was created separately.

```
assetCommands.SendCommand "RUN, HULL GEOM MODULE"
```

The above command runs hull geometry module. Similarly all the modules in ASSET will be run. If the design is not feasible it results in a fatal error in some module. Hence we should check for the fatal error generation after the execution of every module.

```
iError = assetShipType.getError
if (iError =0) then
success = true
end if
```

If there is no fatal error ASSET returns zero for iError otherwise ASSET returns one. Before running the next module we check to see if all the modules prior to it ran properly. If there was any fatal error in any of the modules we skip executing all the other modules. We penalize the design, by setting OMOE equal to a very low value (zero) and LCA to a very high value (10000000) so that the optimizer learns that it is not a good design, and proceed to the next design. If there are no fatal errors we go to check for the convergence of variables in ASSET. As we already discussed it takes a lot of time to converge on all the parameters in ASSET hence we check the convergence of only those parameters that have a direct influence on the objective function values.

Convergence checking: A parameter is said to be converged if the amount of variation in the value of the parameter, in two successive runs of the synthesis modules of ASSET, is less than the specified value of tolerance. The tolerance value is taken to be 0.1%. Let us see the convergence check on the parameter “Endurance”. The first time we complete running all the synthesis modules in ASSET we get the value of the parameter “Endurance” from ASSET and store it.

```
EndurIndex = assetMPL.GetParameterIndex ("ENDURANCE")
set assetParameter = assetMPL.GetParameter (EndurIndex)
Endurance   = assetParameter.value
```

For the same design we run all the ASSET modules once again and obtain the new value of this parameter. If the ratio, of the difference between the parameter values in the current and previous runs to the current value of the parameter, is less than the tolerance value then that variable is converged. If the parameters are not converged, we run the ASSET modules again. This process will be repeated until all the required parameters are converged. Then we proceed with the calculation of the objective function values.

Chapter 5

OBJECTIVE FUNCTIONS

The two objective functions in this project are overall measure of effectiveness (OMOE) and lead ship acquisition cost (LCA).

OMOE

Effectiveness can be defined as the extent to which a system meets its mission requirements. OMOE is a number between 0 and 1 describing the ship effectiveness in specified missions.

OMOE calculation involves four major steps. They are hierarchical arrangement of the required capabilities of the ship, pair-wise comparison of the elements of hierarchy, calculation of the weights and values of performance of the elements of hierarchy and calculating OMOE using these weights and values of performance [9].

Naval vessels perform different types of missions such as Mine Counter Measures (MCM), Marine Amphibian (MARG), and Littoral Surface Warfare (LSUW) [10].

The required capabilities of the ship for different types of missions are arranged in a hierarchical manner. The various mission types form the top level of this hierarchy. Different categories of capabilities necessary for each type of mission form the next level. The actual capabilities that represent each category form the lowest level of hierarchy. These actual capabilities are called the measures of performance (MOP).

Ship design experts were asked to do a pair-wise comparison of the elements of this hierarchy. In pair-wise comparison the expert takes two elements of equal hierarchy and on a relative scale he expresses his opinion as which of the two elements is important and how important it is compared to the other capability. The pair-wise comparison process is applied in a bottom up fashion starting with the elements of the lowest level and moving towards the top level in the hierarchy. Analytical hierarchy process (AHP) theory (Saaty, 1996) was used to analyze the results of the pair-wise comparison process to generate the relative weights of measures of performance (WMOP) for all the MOPs. These are normalized weights and the sum of these weights is equal to 1.

Each of these MOPs may have a number of options to select from. These different options available for each MOP are assigned values of performance (VOP), depending on its range if the MOP is a continuous variable or depending on the option if it is a discrete variable. These VOPs are assigned based on a scale of 0 - 1.

The OMOE is defined as the sum of the products of WMOP and VOP values of all MOPs. The maximum value of OMOE is 1.

The specific details of the OMOE functions of LHA(R) and DDG51 are discussed in chapters 6 and 7.

LEAD SHIP ACQUISITION COST (LCA)

The cost model used to calculate the lead ship acquisition cost of the ship in this project is weight-based. It takes the weights of the seven SWBS groups, the internal communication systems weight, the weapons loads weight, the weight of all the aircrafts it needs to support and the required power of propulsion systems as input [11].

The lead ship acquisition cost has three portions. They are ship builder portion, government portion and post delivery cost. While calculating all these costs an annual inflation rate of 10% from 1981 is applied to the base year.

The inflation factor can be obtained by FI = (1+RI/100) ^ n where n is difference between base year and 1981.

The lead ship construction cost can be obtained by adding the cost of all the SWBS groups, the integration cost, margin cost and the ship assembly and support cost.

Costs of SWBS groups:

Cost of SWBS group 1: CL1 =0.03395 * FI * 0.85 * (W1^0.772)
Cost of SWBS group 2: CL2 =0.00186 * FI * 1.6*(PBPENGTOT^0.808)
Cost of SWBS group 3: CL3 =0.07505 * FI * (W3^0.91)
Cost of SWBS group 4: CL4 =0.10857 * FI * 2.3 * (W4^0.617)
Cost of SWBS group 5: CL5 =0.09487 * FI * 1.3 * (W5^0.782)
Cost of SWBS group 6: CL6 =0.09859 * FI * (W6^0.784)
Cost of SWBS group 7: CL7 =0.00838 * FI * (W7^0.987)
sigmaCLi = CL1+CL2+CL3+CL4+CL5+CL6+CL7

Margin Costs:

CLM = (WM24/ (WLS - WM24)) * sigmaCLi

Integration Costs:

CL8 = 0.034 * 15 * ((sigmaCLi + CLM) ^1.099)

Assembly and support costs:

CL9 = 0.135 * 2.5 * ((sigmaCLi + CLM) ^0.839)

Lead ship construction cost = sigmaCLi+CL8+CL9+CLM

The lead ship price is 110% of the lead ship construction cost, which includes 10% of lead ship construction cost as profit.

The ship builder portion of the LCA can be obtained by adding the lead ship price to the margin provided for the increase in expenses due to changes in orders, which is about 12% of the lead ship price. Hence the ship builder portion of LCA is 112% of the lead ship price.

The government portion of the LCA consists of the costs for military payloads, boats, outfitting cost and margin provided for the increase in cost during production etc, this portion will be about 20% of the lead ship price.

The post delivery cost is 5% of the lead ship price.

LCA = Ship building cost + Government supplies cost + post delivery cost.

This weight based cost model does not provide a good reflection of the change in cost with the change of machinery or some other equipment whose value does not depend on its weight but on the properties of the equipment. If the equipment used is of lower weight and of higher cost then the approximation provided by this model will be less than the actual cost.

The cost model described here is used in the optimization of both LHA(R) and DDG51 ships.

Chapter 6

DESIGN OPTIMIZATION OF LHA(R) SHIPS

INTRODUCTION

This chapter describes the design optimization of LHA(R), the replacement for the US Navy amphibious assault ship. The overall measure of effectiveness (OMOE) and lead ship acquisition cost (LCA) are the objective functions. We want to arrive at designs with high OMOE and low LCA. Trade studies were conducted by a panel of US Navy ship design experts identifying the possible areas of improvement in LHA(R) ships [12]. These areas of improvement form the design variables. There are no constraints applied to this problem. Hence this is a multi-objective unconstrained design optimization problem.

The baseline ship, the trade study options and the adjustments that are to be made to the baseline ship to apply each trade study option were provided by the NSWCCD. New designs are generated by applying various trade study option combinations to the baseline ship. The feasibility of each of these new designs is evaluated using ASSET. The OMOE and LCA of all the feasible designs are used by Darwin to select design variable options (trade study options) for the next generation. The process of generating new designs, their evaluation and OMOE, LCA calculation is repeated until either both OMOE and LCA are converged or the maximum number of generations is reached. Model Center will provide the environment to hold ASSET, Darwin and the other components together and to allow for the data exchange between them.

DESIGN VARIABLES

The possible areas of improvement in LHA(R) ships were identified during the US Navy trade studies. Table 1 lists these areas and options available for each area. These areas of improvement form the design variables for the current study and the options form the alternatives available for each design variable.

Table 1. Description of design variables – LHA(R)

Design variable	Option	Option description
Hangar length and high hats	Option 1	Consume "Composite Shop" and space above into Hangar.
	Option 2	Increase High Hat length by 2 frames on aft end.
	Option 3	Consume "Composite Shop" and space above into Hangar + Additional 5-frame High Hat starting at frame 122.
	Option 4	Combination of 2 + 3
	Option 5	Baseline
Aviation Maintenance and stowage	Option 1	Find alternative location - new stores - for "Supply Mountain" equipment and spares.
	Option 2	New spaces for projected requirement for 11,556 cubic feet and 40 lt for new ACE (over legacy ACE) plus 2,700 cubic feet shortfall from legacy ACE.
Increased cargo cube	140 k Sq. ft.	Baseline
	150 k Sq. ft.	
	160 k Sq. ft.	
	170 k Sq. ft.	Additional stowage space located FWD of well under ramp on 1st plat for small arms/inert cargo - no ballistic protection.

Increased vehicle square	25400 Sq. ft.	Baseline
	26000 Sq. ft.	Relocated Gas Turbine exhaust high hat impinging on 1st platform vehicle stowage. Contingent on results of 22 know machinery study. Approx. 800 sq. ft. gross returned to vehicle square.
	26500 Sq. ft.	Optimize arrangements in upper vehicle deck. Relocate ICE, etc.
	27000 Sq. ft.	Both Changes Made
Galley	Option 0	Distributed Galleys
	Option 1	Consolidated WR/CPO/SSNCO Galley
	Option 2	Consolidated WR/Crew/Troop Galley
Dedicated troop training and muster spaces	Option 0	LHD-8 Baseline
	Option 1	Relocate exercise equipment and utilize existing troop training and muster space as such.
	Option 2	Design dedicated space on 01 Level or 02 level, potentially use existing space, and relocate exercise equipment - distributed exercise rooms.
	Option 3	Redesign current ship training area and include other adjacent shops/STRMs
Boat stowage and handling	External	Install additional fixed overhanging davit for 11M, retain existing LCPL davit
	Internal	Internal stowage for both 11m RIB with fixed launching/recovering system.
Medical	Baseline	PD-2 Medical capabilitiess - 6 OR's, 2 dental OR's, 23 ICU beds, 65 ward beds (Lvl II)
	Option 1	Reduced medical capability.
Damage tolerance 1: Plating	Option 2b	TSS Preferred Option - Just less than best performance
	Option 3	Best Performance

Damage tolerance 2:DAPS	Option 0	Remove DAPS.
	Option 1	TSS Preferred Option - best performance
	Option 2	
	Option 3	Negotiated Option
	Option 4	
	Option 5	
	Option 6	Worst Performance
Damage tolerance 3: UNDEX	Option 1(HSLA 65)	TSS Preferred Option, HSLA 65
	Option 2 (HSS)	HSS Option, ~same performance
IR Signatures	IR 3	LHD-8 Baseline
	IR 2	Midline Option
	IR 1	ESS and EMS, fore and aft
Acoustic signatures	Tech 2	Remove Tech 1
	Tech 1	Remove Tech 2
	Tech 1+2	Technology 1 & Technology 2,Included in baseline PD-2
Purple mission spaces	Option 1	Minimum compliance with MEB/JTF ref. docs.
	Option 2	Moderate compliance with MEB/JTF ref. docs.
	Option 2.1	Flexible spaces with moderate compliance with MEB/JTF ref. docs.
	Option 3	Maximum compliance with MEB/JTF ref. docs.
Bomb Farm alternatives	External	Bomb Farm on Flight Deck
	Internal	Internal Bomb Farm on Main Deck adjacent to Elev 1 & 3. Min. protection.
Machinery	Option A	LHD 8 mechanical drive system
	Option B	"Flipped Shaft" new mechanical drive system

	Option C	New mechanical drive system
	Option E	Combined GT & APS drive system
	Option F	Mechanical drive system w/de-rated MT-30

In Table 1 we see only five options for Machinery. The information about design variable option - Option D (Integrated power systems) was not made available and hence not considered.

Studies were conducted by a panel of ship design experts to assess the impact of these improvements on the available area in the ship, weight of the structure and change in KG. Table 2 provides the impact of these options on the available area in the ship, weight of the structure and change in KG.

Table 2. The impact of design variable options on Area, Wt and KG.

Design variable	Option	Impact on			
		Area, sq.ft.	Weight, LT	KG, feet	
Hangar length and high hats	Option 1	3,800	0		
	Option 2	1,120	0		
	Option 3	6,500	7	-0.01	
	Option 4	7,620	7	-0.01	
	Option 5	0	0	0	
Aviation Maintenance and stowage	Option 1	400	0	0	
	Option 2	2,100	40	0	
Increased cargo cube	140 k	0	0	0	
	150 k	800			
	160 k	1600			

	170 k	2,400			
Increased vehicle square	25400	0			
	26000	600			
	26500	1100			
	27000	1600			
Galley	Option 0	0			
	Option 1	-540			
	Option 2	-540			
Dedicated troop training and muster spaces	Option 0	0			
	Option 1	3400			
	Option 2	3000			
	Option 3	3670			
Boat stowage and handling	External	0	15		
	Internal	3000			
Medical	Baseline	0			
	Option 1	-1400			
Damage tolerance 1: Plating	Option 2b		108		
	Option 3		428		
Damage tolerance 2: DAPS	Option 0	0	0		
	Option 1	5,880	261		
	Option 2	4,900	225		
	Option 3	3,920	189		
	Option 4	2,940	151		
	Option 5	1960	117		
	Option 6	980	83		

Damage tolerance 3: UNDEX	Option 1 (HSLA 65)		550		
	Option 2 (HSS)		1000		
IR Signatures	IR 3		0		
	IR 2		-2		
	IR 1		-4		
Acoustic signatures	Tech 2		-20		
	Tech 1		100		
	Tech 1+2		0		
Purple mission spaces	Option 1	545			
	Option 2	3,413			
	Option 2.1	2,400			
	Option 3	6,272			
Bomb Farm alternatives	External	0	0	0	
	Internal	2,000	117	0.05	
Machinery	Option A	0	0	0.00	
	Option B		-74	0.07	
	Option C		-74	0.07	
	Option E	0	-35		
	Option F	0	-20		

OBJECTIVE FUNCTIONS

OMOE: To compare the OMOE of various ship designs we need a quantitative way of measuring the OMOE. LHA(R) ships perform different types of missions. The parameters influencing these missions can be organized in a hierarchical manner.

Thus all these parameters fall into four categories. They are mission, mobility, survivability and own-ability. The Mission category has two sub categories. They are operational and capacity. The Survivability category also has two sub categories vulnerability and susceptibility. The lowest level in the hierarchy will be occupied by the actual MOPs. Purple mission spaces, training and muster spaces, medical, hangar length and high hats come under the operational category. Vehicle square, cargo cube, aviation stowage come under the capacity category. Sustained speed, seakeeping come under mobility category. Plating, DAPS, UNDEX come under vulnerability category. IR signature, acoustic signature, RCS come under susceptibility category. KG service life allowance, weight service life allowance, boat stowage and galley come under own-ability category. The Bomb farm is divided into two parts. The first part ordnance flow comes under operational category and the second part weapons vulnerability comes under vulnerability category.

A panel of experts was asked to do a pair-wise comparison of elements in the hierarchy. Table 3 shows the results of pair-wise comparison - the MOPs and their WMOP and VOP values. To evaluate the effectiveness of a ship, the contribution of each MOP, to the effectiveness of the ship, is obtained by multiplying WMOP with that of the VOP for the type of alternative chosen. The OMOE of the ship is obtained by adding the contributions of all these individual components.

In Table 3 we can see that the sum of the WMOP values of all the MOPs is 1. The VOP value of each MOP is also normalized to 1 [13]. Hence the maximum possible value of OMOE is 1.

Table 3. Measures of Performance and their WMOP, VOP values.

	MOP ↓ MOP alternatives →	WMOP	MOP alternatives and their VOPs						
			1	2	3	4	5	6	7
1	Hangar	0.197	0.241	0.224	0.484	1	0.091	---	-
2	Aviation maintenance stowage	0.066	0.2	1	--	--	---	---	-
3	Cargo Cube	0.020	0.8	0.9	0.95	1	---	---	-
4	Vehicle Square	0.036	0.827	0.827	0.904	1	---	---	-
5	Galley Arrangement	0.018	0.5	0.5	1	---	---	---	-
6	Training and Muster spaces	0.055	0.5	1	0.707	0.707	---	---	-
7	Boat Stowage	0.018	0.9	1	---	---	---	---	-
8	Medical	0.028	1	0.8	---	---	---	---	-
9	Plating	0.110	0.143	1	---	---	---	---	-
10	DAPS	0.020	1	0.655	0.417	0.263	0.168	0.112	0
11	UNDEX	0.046	1	0.143	---	---	---	---	-
12	IR	0.005	1	0.395	0.094	---	---	---	-
13	Acoustic	0.037	0.237	0.094	1	---	---	---	-
14	Purple spaces	0.065	0.284	0.333	0.403	1	---	---	-
15	Ordnance Flow	0.080	1	1	---	---	---	---	-
16	Weapons Vulnerability	0.032	1	1	---	---	---	---	-
17	RCS	0.022	1	0	---	---	---	---	-
18	KG service life allowance	0.049	VOP calculation discussed below.						
19	Seakeeping	0.045	VOP calculation discussed below.						
20	Weight service life allowance	0.040	VOP calculation discussed below.						
21	Sustained speed	0.011	VOP calculation discussed below.						

VOPs of KG service life allowance:

```
if (KGAllow < 1 ft) then VOP18 = 0
else if (KGAllow>=1ft and KGAllow < 1.5 ft) then
    VOP18 = (KGAllow - 1) * 0.294 + 0.166
else if (KGAllow>=1.5 ft and KGAllow < 2 ft) then
    VOP18 = (KGAllow - 1.5) * 0.466 + 0.313
else if (KGAllow>= 2 ft and KGAllow < 2.5 ft) then
    VOP18 = (KGAllow - 2) * 0.908 +0.546
else VOP18 = 1
```

VOPs of Seakeeping: Depends on the roll period.

```
if (roll period < 13 sec) then VOP19 = 0
else if (roll period >= 13 sec and roll period < 14 sec) then
    VOP19 = (roll period - 13)* 0.121 + 0.146
else if ( roll period >= 14 sec and roll period < 15 sec) then
    VOP19 = (roll period - 14)* 0.263 + 0.267
else if ( roll period >= 15 and roll period <16) then
    VOP19 = (roll period - 15)* 0.47 + 0.53
else VOP19 = 1
```

VOPs of Weight service life allowance:

```
WAllow = (Full load wt - 30000 MT)/300

if (WAllow <5) thenVOP20 = 0
else if (WAllow >=5 and WAllow <6) then
    VOP20 = (WAllow - 5) * 0.150 +0.153
else if ( WAllow >=6 and WAllow <7) then
    VOP20 = (WAllow - 6) * 0.27 +0.303
else if (WAllow >=7 and WAllow < 7.5) then
```

```
        VOP20= (WAllow - 7) * 0.427 +0.573
    else VOP20 = 1
```

VOPs of Sustained speed: Sustained speed in knots.

```
    if ( Sustn_Speed < 21.5) thenVOP21 = 0
    else if (Sustn_Speed >=21.5 and Sustn_Speed < 22) then
        VOP21 = (Sustn_Speed - 21.5)*0.6 + 0.5
    else if(Sustn_Speed >=22 and Sustn_Speed < 22.5) then
        VOP21 = (Sustn_Speed -22)*0.2 +0.8
    else if(Sustn_Speed >=22.5 and Sustn_Speed < 23) then
        VOP21 = (Sustn_Speed -22.5)*0.2 +0.9
    else VOP21 = 1
```

If we consider a design 2,1,4,4,1,2,1,1,1,7,1,1,3,4,1,2,2, 1.25,13.5,6,22.75 where the numbers represent the option chosen for each of the first 17 MOPs and the value of the required parameters for the last 4 MOPs. The OMOE of the design can be obtained by $\sum$MOP*VOP. Hence the OMOE of this design would be (0.197*0.224) + (0.066*0.2) + (0.020*1) + (0.036*1) + (0.018*0.5) + (0.055*1) + (0.018*0.9) + (0.028*1) + (0.110*0.143) + (0.020*0) + (0.046*1) + (0.005*1) + (0.037*1) + (0.065*1) + (0.080*1) + (0.032*1) + (0.022*0) + (0.049* 0.2395)+ (0.045 * 0.2065) +(0.040 * 0.303) + (0.011 * 0.95)= 0.545856.

Lead Ship Acquisition Cost (LCA): LCA is determined as described in the chapter objective functions.

PROBLEM DEFINITION

Now that we know the objective functions and design variables we can formally define the LHA(R) design optimization problem. There are no constraints. So the problem can be defined as

Maximize OMOE and Minimize LCA subject to the condition that the US Navy design requirements are satisfied.

All the design variables in this problem are discrete variables. We can combine the various alternatives available for the design variables in **77,414,400** ways and thus can obtain **77,414,400** designs. Solving the problem by evaluating the feasibility, OMOE and Cost of all these designs will take a very long time even with the use of computers. So we use the GA based optimizer Darwin for this purpose.

MODEL DESCRIPTION

Figure 4 shows the Model Center version of the LHA(R) design optimization model. Figure 5 represents the same model by exposing the contents of the assembly components. A group of components are combined together to form an assembly component.

In Figure 4 the "Optimizer" represents the GA based optimizer Darwin. We specify our design variables and objective function using the GUI of "Optimizer". The "Optimizer" generates a new design and sends the values to the component "Setup". "Setup" gets access to ASSET using the COM operability and loads the baseline ship then "Setup" checks whether the generated values are within their range and if they are, it will send the design values to the corresponding trade study option components. Each trade study option component is represented by a separate assembly component. When this trade study component receives the design value, the corresponding trade study option component will run. It will get access to ASSET using the COM operability and apply the corresponding changes to the baseline ship. Then we go to the "ASSETmodules" assembly component where we run the synthesis modules corresponding to the MONOCV ship type. If we get a feasible design we go to the “converger input” component which gets the values of the convergence parameters - Endurance, Full load wt, Sustained speed and GMt from ASSET and inputs these values to the converger component. The converger component checks for the convergence of all these four parameters. If they are converged we go to calculate the values of objective functions. The “Test” component gets the ASSET parameters required in calculating the cost and sends them to “Cost” component. “Cost” component determines the value of the objective function LCA and activates the “OMOE” component. The “OMOE” component determines the value of the second objective function “OMOE”. The results from both Cost and OMOE components were input to the “Optimizer”. This process is repeated until the maximum number of generations is reached.

RESULTS – LHA(R) DESIGN OPTIMIZATION

We began the optimization process with a population size of 20 and the values of the GA parameters – Crossover probability, mutation probability are chosen to be 1 and 0.03 respectively. Multiple Elitist method of selection is used. With this combination of parameters the optimizer seemed to produce a converged Pareto front in fifty generations.

Figure 6 shows the global set of Pareto designs and Figure 7 shows the progress of the Pareto front for the same problem. In Figure 6 a point of interest is one that has high effectiveness for a given range of cost. Before this point are the designs having similar cost but less effectiveness. Beyond this point we cannot find any big increase in OMOE even if we increase cost until we go near another point like this.

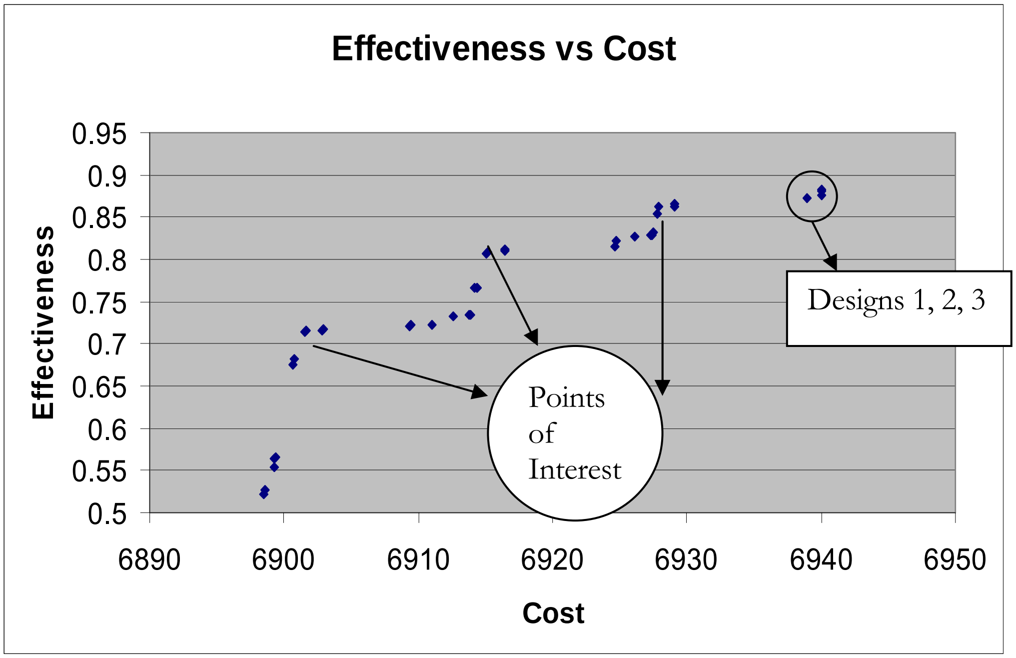


Figure 6. Pareto front obtained using population size of 20 in 50 generations.

Looking at these results the decision maker can decide, which is the best design that he can get for his capital? The results show how a little change in the capital can influence the best design he can get. If a small increase in capital results in a far better design (design with more effectiveness) then he probably could decide to increase his capital. If decreasing the capital by a large amount results in a design whose effectiveness value is only a little lower then he may decide to decrease his capital.

Two designs having almost the same values of effectiveness and cost can be very similar or may also be very different from each other. If there are many designs having similar effectiveness values in the same range of cost then the decision maker can select the design having his preferred design variable options.

Consider designs 1, 2, 3 highlighted in Figure 6. The OMOE and CLA values of the first design are 6939.98 Mdollars, 0.87667 for the second design are

6938.90 Mdollars, 0.8726625 and for the third design are 6939.98 Mdollars, 0.8806. The design variable option values of these three designs are shown in Table 4.

Table 4. Design variable options of the designs highlighted in Figure 6.

Design Variable	Design 1	Design 2	Design 3
Machinery	4	3	4
Hangar	4	4	4
Aviation maintenance	2	2	2
Cargo Cube	1	4	4
Vehicle Square	4	4	4
Galley	3	1	3
Training and Muster spaces	2	2	2
Boat Stowage	1	1	1
Medical	1	1	1
Plating	2	2	2
DAPS	1	1	1
UNDEX	1	1	1
IR	3	2	3
Acoustic	3	3	3
Purple spaces	4	4	4
Bomb farm	2	2	2

If we consider designs 1 and 3 from Table 4 we can see that there is not much difference between them except in the volume provided for the cargo cube. On the other hand if we consider designs 1 and 2 from Table 4 we can see that these two designs have different values for the design variable options 1, 4, 6, 13. The values of these design variable options are shown in Table 5.

Table 5. Differences in the design variable options of design 1 and design 2.

Design variable	Option	Represents
1 – Machinery for design 1	4	Combined GT&APS system
for design 2	3	New Mechanical Drive System
4 - Cargo Cube for design 1	1	160K
for design 2	4	140K
6 – Galley for design 1	3	Distributed Galley
for design 2	1	Consolidated Galley
13 – IR for design 1	3	LHD 8 Baseline
for design 2	2	Midline Option

From these results we could see that to arrive at a design having desired values of cost and effectiveness, we may have the option to choose from a group of designs which are very different from each other or we can select from designs that are very much similar to each other.

In Figure 7 we can see that the designs from generation 30 to generation 50 falling almost on top of each other and thus forming a converged Pareto front. We generated two more sets of results using population size 10 for 100 generations and population size 50 for 20 generations. Figure 8 is produced using a population size of 50 for 20 generations. It shows the Pareto front movement with number of generations. The results obtained using a population size of 10 for 100 generations is shown in Figure 9. Observations similar to the ones we made using a population size of 20 can be made with these new sets of results.

Figure 10 shows the three converged Pareto fronts produced by varying the population size. The population sizes for the three cases are 10, 20 and 50. The total number of designs produced in all the three cases is 1000. In Figure 10 we can see that the results obtained by generating equal number of designs with different population sizes to be almost identical. The Pareto front generated in all the three cases is stepped in nature. These steps indicate regions where we can get designs with good improvement in OMOE while there is not much increase in the cost. Designs at the top and bottom of these steps were observed and in most cases it is found that varying only a few of the design variables caused the OMOE of the designs to improve with the corresponding increase in cost being very small.

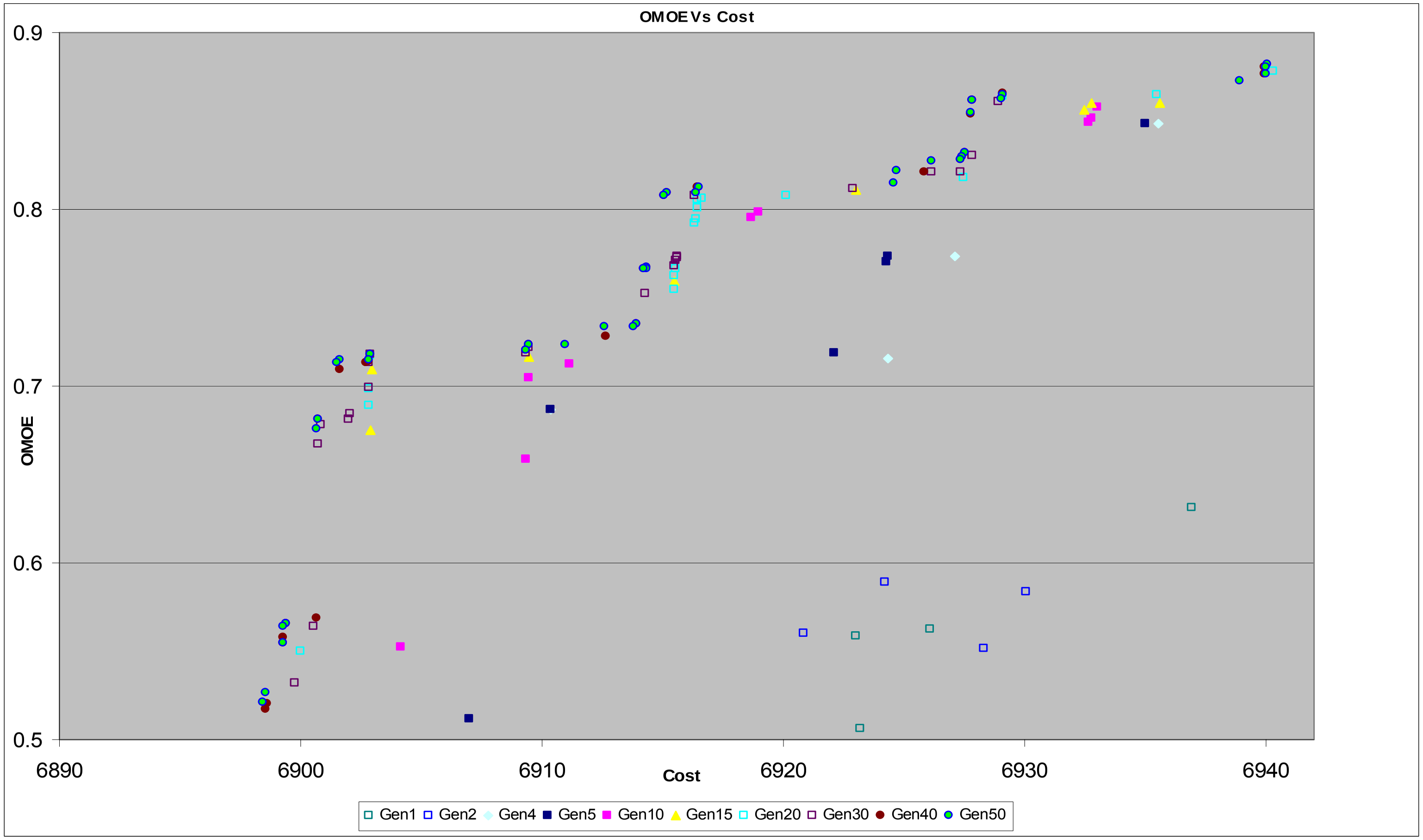


Figure 7. Non dominated frontier at various generations, generated with a population size of 20 for 50 generations.

Figure 8. Pareto front development obtained using population size of 50 in 20 generations.

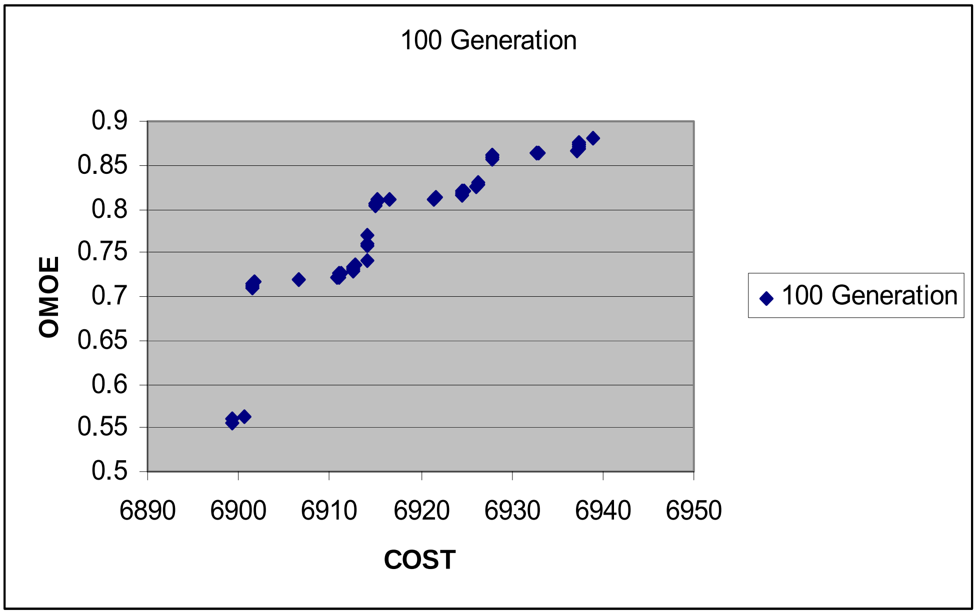


Figure 9. Results generated with population size10 in 100 generations.

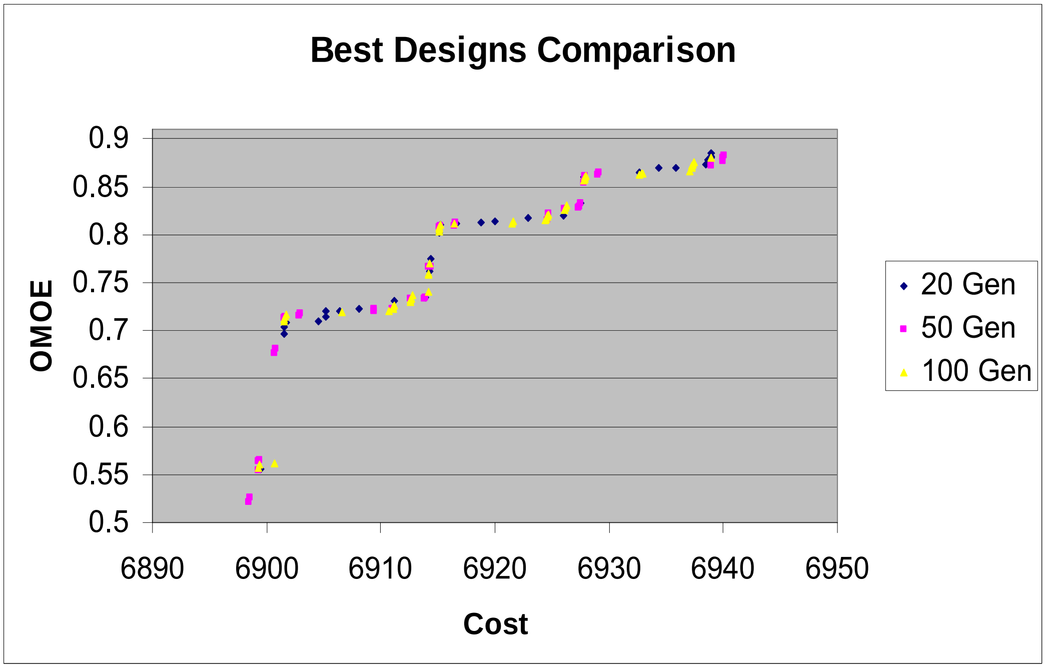


Figure 10. Final sets of designs generated using population sizes 50, 20 and 10.

Table 6 shows the design variable options of designs at the top and bottom of steps of the Pareto fronts generated with 50 and 20 population sizes. By observing values in Table 6 two design variables were found to be of major influence in forming steps in the Pareto front. They are Aviation maintenance stowage and Plating (Damage tolerance 1). Each of these two design variables has two options. The designs with option 2 for these two variables were at the top with high OMOE values while the designs with option 1 were at the bottom of the step with low OMOE values.

Table 6. Design variable options of designs forming steps in the Pareto Front.

Design variable→ Design ↓	MACHINERY	HANGER	AVIATION	CARGOCUBE	VEHICLESQ	GALLEY	TRAINING	BOAT STWG	MEDICAL	PLATING	DAPS	UNDEX	IR	ACOUSTIC	PURPLE	BOMB FARM	Cost	OMOE
Step 1 of the Pareto front generated with population size 20.																		
Design 11	4	4	1	4	4	3	2	1	1	1	1	1	2	3	4	2	6913.90	0.7351
Design 15	3	4	1	2	4	3	2	1	1	2	7	1	1	3	4	2	6915.26	0.8101
Step 2 of the Pareto front generated with population size 20.																		
Design 20	4	4	1	4	4	3	2	1	2	2	2	1	1	3	4	2	6925.99	0.8199
Design 23	3	4	2	4	4	3	2	1	1	2	7	1	3	3	4	2	6927.77	0.8602
Step 2 generated by a different set of designs of the Pareto front generated with population size 20.																		
Design 22	4	4	1	4	4	3	2	1	1	2	1	1	1	3	4	2	6927.51	0.8324
Design 23	3	4	2	4	4	3	2	1	1	2	7	1	3	3	4	2	6927.77	0.8602
Step 1 of the Pareto front generated with population size 50.																		
Design 16	4	4	1	4	4	3	2	1	1	1	1	1	3	3	4	2	6913.81	0.7336
Design 21	3	4	1	4	4	3	2	1	1	2	7	1	3	3	4	2	6915.09	0.8075
Step 2: of the Pareto front generated with population size 50.																		
Design 28	4	4	1	4	4	3	2	1	1	2	1	1	3	3	4	2	6927.35	0.8278
Design 32	3	4	2	4	4	3	2	1	1	2	7	1	2	3	4	2	6927.86	0.8617

PROBLEMS DURING THE OPTIMIZATION OF LHA(R): It was observed that ASSET cannot remain open for more than 218 iterations of the optimization process. Discussing with ASSET developers, the possible reasons for this crash were thought to be (1). Memory leak in FORTRAN. (ASSET was programmed in FORTRAN) (2). Variables in ASSET were updated too many times. (3). ASSET may have a limitation on the maximum run time.

To avoid this problem we thought that it might be a good idea to close the ASSET session and open it again after every 50 or 100 iterations. But we can not do this manually as we want the process to have no human intervention. After 50 iterations we closed ASSET by sending the command "Exit". Until this point we used only the capabilities of COM. Now to open the application ASSET we need to get access to the operating system. We used Windows Script Host (WSH) to develop the method that can access the operating system and invoke ASSET automatically.

Windows Script Host supports 14 different objects. We used the WshShell object which has methods that enable one to activate an application, to send a command to that application, etc. Once ASSET is invoked we should stop the process of sending commands to ASSET for a few hundred milliseconds to make sure that the commands that we are sending are going to the pop up window we get to select the ship type. But the Model Center environment does not support the sleep method that we use in VBScript. Hence we need to load the dynamic link library "PHXSleep.dll" before using the sleep command [14]. This file is provided by Phoenix Integration. Once the WSH script sleep command runs, we can send commands to select the ship type and the databank that we need to use. The component "Start" shown in Figure 4 contains the instructions to invoke ASSET. This component will be executed once every 50 iterations through the optimization process.

Chapter 7

DESIGN OPTIMIZATION OF DDG51

INTRODUCTION

This chapter describes the design optimization of DDG51, a guided missile destroyer ship. The overall measure of effectiveness (OMOE) and lead ship acquisition cost (LCA) are the objective functions. We want to arrive at designs with high OMOE and low LCA. Trade studies were conducted by a panel of US Navy ship design experts identifying the possible areas of improvement in DDG51 class naval vessels. These areas of improvement form the first set of design variables. Design variables based on the principal dimensions of the ship form the other set. There is a constraint in ASSET on the length to depth ratio. The length to depth ratio should be less than 15. Hence this is a multi-objective constrained design optimization problem.

The baseline ship is generated by modifying the Flight I design which is provided in the MONOSC version of ASSET supplied by the NSWCCD. New designs are generated by applying various trade study option combinations to the baseline ship. The feasibility of each of these new designs is evaluated using ASSET. The OMOE and LCA of all the feasible designs are used by Darwin to select design variables for the next generation. The process optimization process is repeated until either OMOE, LCA both are converged or the maximum number of generations is reached. Model Center will provide the environment to hold ASSET, Darwin and the other components together and to allow for the data exchange between them. Analysis Server's file wrapper capability was used to transfer the McCreight Index, generated by the seakeeping analysis module in

ASSET, to Model Center. This McCreight index is used in the calculation of OMOE.

BASELINE SHIP GENERATION

To compare the OMOE and cost of different designs we should have those designs generated from the same baseline ship. The Flight I model provided in the surface combatant version of ASSET is used as the basis to generate the baseline ship. The input variables to all the modules of ASSET were observed to see if they are to be modified to make ASSET run without any human intervention. Table 7 provides the changes made to Flight I model to get the baseline ship.

Table 7. Changes made to Flight I to obtain the base line ship

Variable Name	Value of Variable in Flight I	Value of Variable in Baseline
Hull Subdiv Ind	Given	Calculate
Trans Bhd Spacing	--	0.075
Deckhouse size Ind	MAX	AUTO X
Deckhouse geometry Ind	Given	GENERATE
Deckhouse material type Ind	Other	STEEL
Endur displacement Ind	Full load	Average Displacement
Prop dia Ind	Given	Calculate
Prop series Ind	Given	TROOST
Prop Area Ind	Given	Calculate
Prop Location Ind	Given	Calculate
Pitch ratio Ind	--	Calculate
Rudder size Ind	Given	Calculate
Thrust Brg location Ind	Given	Calculate

Machy KG Ind	Given	Calculate

The variables having the option "Given" in Flight I will be expecting the user to provide that information before executing the corresponding module. If we change them to "Calculate" or "Generate" it allows that ASSET module to set those values. When Hull Subdiv Ind is set to "Calculate" the transverse bulkhead spacing should be provided so that the Hull Subdiv module can determine the position of the transverse bulkheads. Deck house is constructed of steel hence the material type for deck house should be steel. Deck house size indicator is an input to the deck house module. We want to have a deck house that provides the required amount of arrangeable area hence we chose option "AUTO-X". This option allows ASSET to auto expand the deck until the available deck house area becomes equal to the required deckhouse area. The Endurance Displacement Ind set to "Average Displacement" generates a design which will be stable until a fraction of the fuel is remaining. Compensating ballasting does not begin until only that fraction of the usable fuel load is remaining. The propeller series indicator set to TROOST determines the hydrodynamic characteristics of the propeller based on the Wageningen B-Screw series data that is provided within ASSET. The model obtained by making these changes to the Flight I is used as the base line ship.

DESIGN VARIABLES

DDG51 class ships are guided missile destroyers designed to operate in multi threat environments. These ships have the ability to strike and defend any kind of environment such as air, surface and subsurface. DDG51 class vessels are equipped with the integrated weapon system called AEGIS combat system (ACS). The cooperative engagement capability installed on DDG51 class vessels gives the advantage of network centric warfare capability.

The AEGIS combat system (ACS) [15] provides several capabilities such as Anti-Air Warfare (AAW, 4), Anti-Surface ship Warfare (ASUW, 2), Anti-Submarine Warfare (ASW, 5), Command Control Communications Connectivity and Intelligence (C4I, 2), Mine Counter Measures (MCM, 3), Naval Surface Fire Support (NSFS, 3), Sensor and Electronic Warfare (SEW, 3), Strike Warfare (STK, 3), Vertical Launch System (VLS, 4). The numbers in the parenthesis represent the number of alternatives available for each of these warfare areas. Tables A2 to A10 show the components of all the alternatives of these war areas.

These ACS capabilities, each with a discrete set of alternatives, form a set of discrete design variables. There is another set of design variables which are used as input values for the hull geometry module in ASSET. They are length, beam, depth, breadth to draft ratio, prismatic coefficient and maximum section coefficient. These are continuous variables and the suggested range of each of these variables is shown in Table 8.

Table 8: Continuous Design variables and their ranges

Parameter	Lower bound	Upper bound
Length	450 ft	750 ft
Beam	30 ft	120 ft
Depth amidships	36 ft	50 ft
Breadth to draft ratio	2.8	3.8
Prismatic coefficient	0.57	0.63
Maximum section coefficient	0.76	0.84

OBJECTIVE FUNCTIONS

A DDG51 class vessel performs different types of missions such as Surface Action mission (SAG), Mine Counter Measures mission (MCM), Marine

Amphibian (MARG), etc. Figure 11 shows the hierarchical arrangement of the MOPs of DDG51 [16].

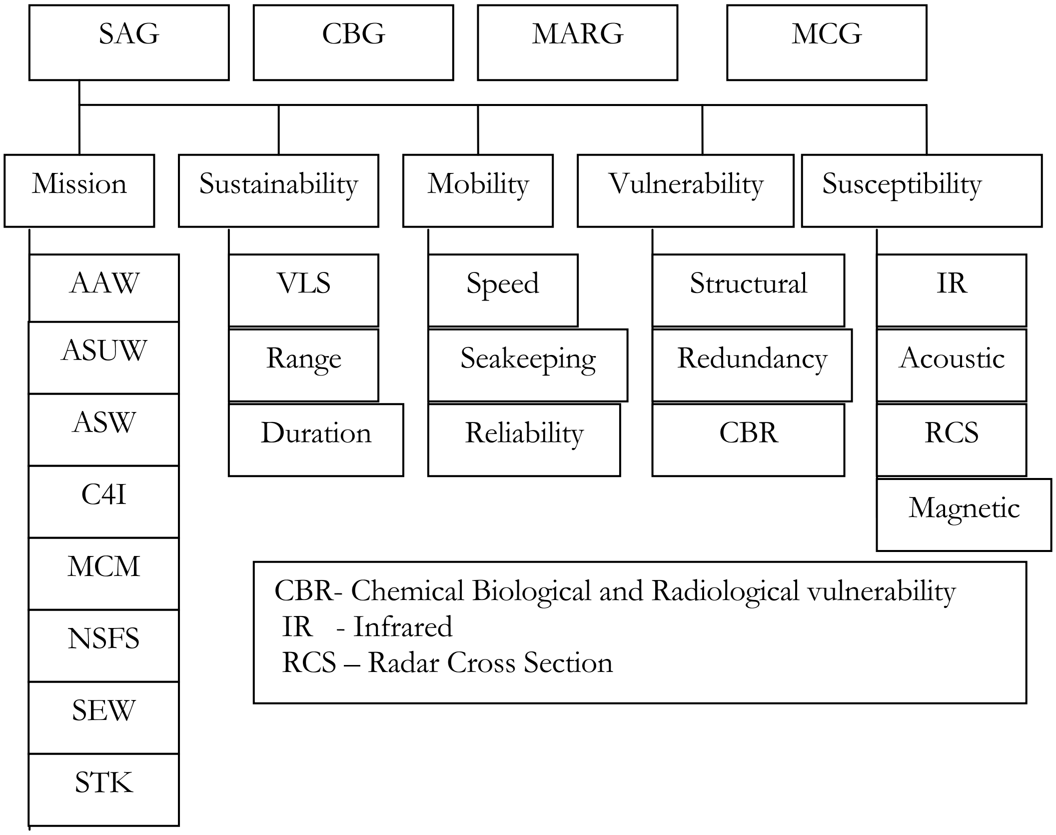


Figure 11. Hierarchical order of MOPs of DDG51

The Ship design expert at Virginia Tech, Dr.Brown, did the pair-wise comparison of the elements of the hierarchy. The WMOPs and VOPs calculated based on the results of pair-wise comparison are shown in Table 9.

Table 9. MOPs of DDG51 and their WMOP and VOP values

ACS capability	Option 1 VOP	Option 2 VOP	Option 3 VOP	Option 4 VOP	Option 5 VOP	WMOP
AAW	1.0	0.9	0.5	0.0	--	0.09
ASUW	1.0	0.5	--	--	--	0.088
ASW	1.0	0.9	0.5	0.3	0.0	0.065
C4I	1.0	0.0	--	--	--	0.084
MCM	1.0	0.9	0.0	--	--	0.048
NSFS	1.0	0.9	0.0	--	--	0.097
SEW	1.0	0.8	0.0	--	--	0.055
STK	1.0	0.8	0.0	--	--	0.034
VLS	1.0	0.822	0.5	0.1	--	0.082
Range	1.0	0.667	0.187	0.071	--	0.052
Stores Duration	1.0	0.2	--	--	--	0.032
Speed	Calculated by interpolation: Shown below.					0.019
Seakeeping	Calculated by interpolation: Shown below.					0.049
Reliability	0.333	1.0	--	--	--	0.032
Structural Vulnerability	1.0	0.333	--	--	--	0.043
Redundancy	0.333	1.0	--	--	--	0.032
CBR	0.2	0.6	1.0	--	--	0.014
IR	1.0	0.2	--	--	--	0.012
Acoustic	1.0	0.333	--	--	--	0.014
RCS	Calculated by interpolation: Shown below.					0.018
Magnetic	1.0	0.0	--	--	--	0.037

$\sum$WMOP = 0.997 ≈ 1.0

Interpolation Function:

The interpolation function, interpolate (a, b, x, y, F), gives the interpolated value of a function at the point F which is between x and y. The value of the function at x and y are given by a, b respectively.

MOP = interpolate (a, b, x, y, F) = [(b – a)*(F-x)]/(y – x) + a

VOPs of Speed:

Speed is a continuous variable. The value of the VOP depends on the range of Speed.

If Speed <27 then VOP = 0
If 27<Speed<28 then VOP = interpolate (0.000, 0.082, 27, 28, Speed)
If 28<Speed<29 then VOP = interpolate (0.082, 0.177, 28, 29, Speed)
If 29<Speed<30 then VOP = interpolate (0.177, 0.491, 29, 30, Speed)
If 30<Speed<31 then VOP = interpolate (0.491, 0.818, 30, 31, Speed)
If 31<Speed<32 then VOP = interpolate (0.818, 1.000, 31, 32, Speed)
Else VOP = 1.0

VOPs of Seakeeping:

McC is McCreight Index., its value is obtained from ASSET.

If McC <6.0 then VOP = 0
If 6<McC<8 then VOP = interpolate (0.000, 0.076, 6, 8, McC)
If 8<McC<10 then VOP = interpolate (0.076, 0.097, 28, 10, McC)
If 10<McC<12 then VOP = interpolate (0.097, 0.197, 10, 12, McC)
If 12<McC<14 then VOP = interpolate (0.197, 0.452, 12, 14, McC)
If 14<McC<16 then VOP = interpolate (0.452, 0.717, 14, 16, McC)

If 16<McC<18 then VOP = interpolate (0.717, 0.883, 16, 18, McC)
If 18<McC<20 then VOP = interpolate (0.883, 1.000, 18, 20, McC)
Else VOP = 1.0

VOPs of Radar Cross Section (RCS)

VD is volume of Deck house.

If VD <44000 then VOP = 1.0
If 44000<VD<100000 then VOP = interpolate (1, 0.405, 44000, 100000, VD)
If 100000<VD<220000 then VOP=interpolate (0.405, 0.0, 100000, 220000, VD)
Else VOP = 0.0

Now that the WMOPs and VOPs of all MOPs are known the OMOE is calculated by the sum of the products of WMOP and VOP of all the MOPs of DDG51. LCA is determined as described in the chapter objective functions.

COMBAT SYSTEMS:

When the optimizer selects the option for each war area, such as AAW or ASW, the combat system components corresponding to that option of the war area are applied to the base line ship. This process is explained in APPENDIX under the section COMBAT SYSTEMS AND THEIR IMPLEMENTATION IN ASSET [16].

PROPULSION MACHINERY:

There are three different types of transmission systems available. They are mechanical, Electrical and Integrated power systems.

In this project we used only mechanical transmission type for all the designs. Lack of information about IPS systems did not permit us to implement them. For the mechanical transmission type we are having six different options to select from. The characteristics of all the six different options was provided in Table 10 [17].When implementing them in ASSET all we need to do is to select the engine type and all the other characteristics will be taken from within ASSET.

Table 10. Propulsion machinery options and their characteristics

Machinery Option	**No. of prop shafts**	**Total BHP (hp)**	**SFC at endurance speed (kg/kW-hr)**	**Machinery box minimum height (m)**	**Machinery weight (MT)**
2 – LM2500	1	52500	0.264	6.14	655.2
2 ICR (Wesths WR21 29)	1	58100	0.199	6.22	745.8
1 ICR, 1 LM2500	1	55300	0.202	6.22	767.2
4 PC2.5 V16	2	41600	0.21	6.67	902.9
2 LM2500 2 PC2.5 V16	2	77300	0.21	6.67	1187.6
4 LM2500	2	105000	0.261	6.62	423

PROBLEM DEFINITION

Now that we know the objective functions and design variables we can formally define the DDG51 design optimization problem. The constraint involved in this problem says that the Length/Depth ratio should be less than 15. So that problem can be defined as

Maximize OMOE and Minimize LCA subject to the condition that the Length/Depth ratio should be less than 15.

MODEL DESCRIPTION

Figure 12 shows the Model Center version of the DDG51 design optimization model. Figure 13 represents the same model exposing the contents of the assembly components of the model.

In Figure 12 the "Optimizer" represents the GA based optimizer Darwin. We specify our design variables and objective functions using the GUI of "Optimizer". The "Optimizer" generates a new design and sends the values to the component "Setup". "Setup" gets access to ASSET using the COM operability and loads the baseline ship. Then "Setup" checks whether the generated values are within their range and if they are, it will send the design values to the corresponding combat system option components. Each combat system option component is represented by a separate assembly component. When this combat system component receives the design option value, the corresponding combat system study option component will run. It will get access to ASSET using the COM operability and apply the corresponding changes to the baseline ship. Then the component "OMOE PL Completed" will wait to make sure all the combat system options are applied. When all the payload adjustments corresponding to combat systems are applied to the baseline ship, the synthesis modules corresponding to the MONOSC ship type in ASSET are run. Every time we complete executing all the synthesis modules in ASSET, the converger checks for the convergence of the four parameters - Endurance, Full load wt, Sustained speed and GMt from ASSET. When all these parameters are converged we run the seakeeping analysis module in ASSET. The output from this module gives the value of McCreight Index which is used to evaluate the seakeeping effectiveness of the design. As we discussed earlier the output from analysis modules is for display purposes only. Hence to access this value, we developed script components to do this.

- Copy the string containing the value of the McCreight Index from the printed reports in ASSET and save it in a notepad file in the "analyses" directory of the Analysis server.
- The file wrapper utility of Analysis Server is used to parse the notepad file we generated and obtain the value of the McCreight Index.
- Send the value of McCreight index to Model Center.

We then proceed to the evaluation of objective functions. The "Test" component gets the ASSET parameters required in calculating the cost and sends them to "Cost" component. "Cost" component determines the value of the objective function LCA and activates the "OMOE" component. The "OMOE" component determines the value of the second objective function "OMOE". The results from both Cost and OMOE components were input to the "Optimizer". This process is repeated until the maximum number of generations is reached. As we discussed in Chapter 6, ASSET cannot remain open for more than 218 iterations. Hence ASSET was restarted, each time, when 50 iterations through the ASSET modules are completed.

The optimization process was repeated with different population sizes and different number of generations. The results generated are discussed in the next section.

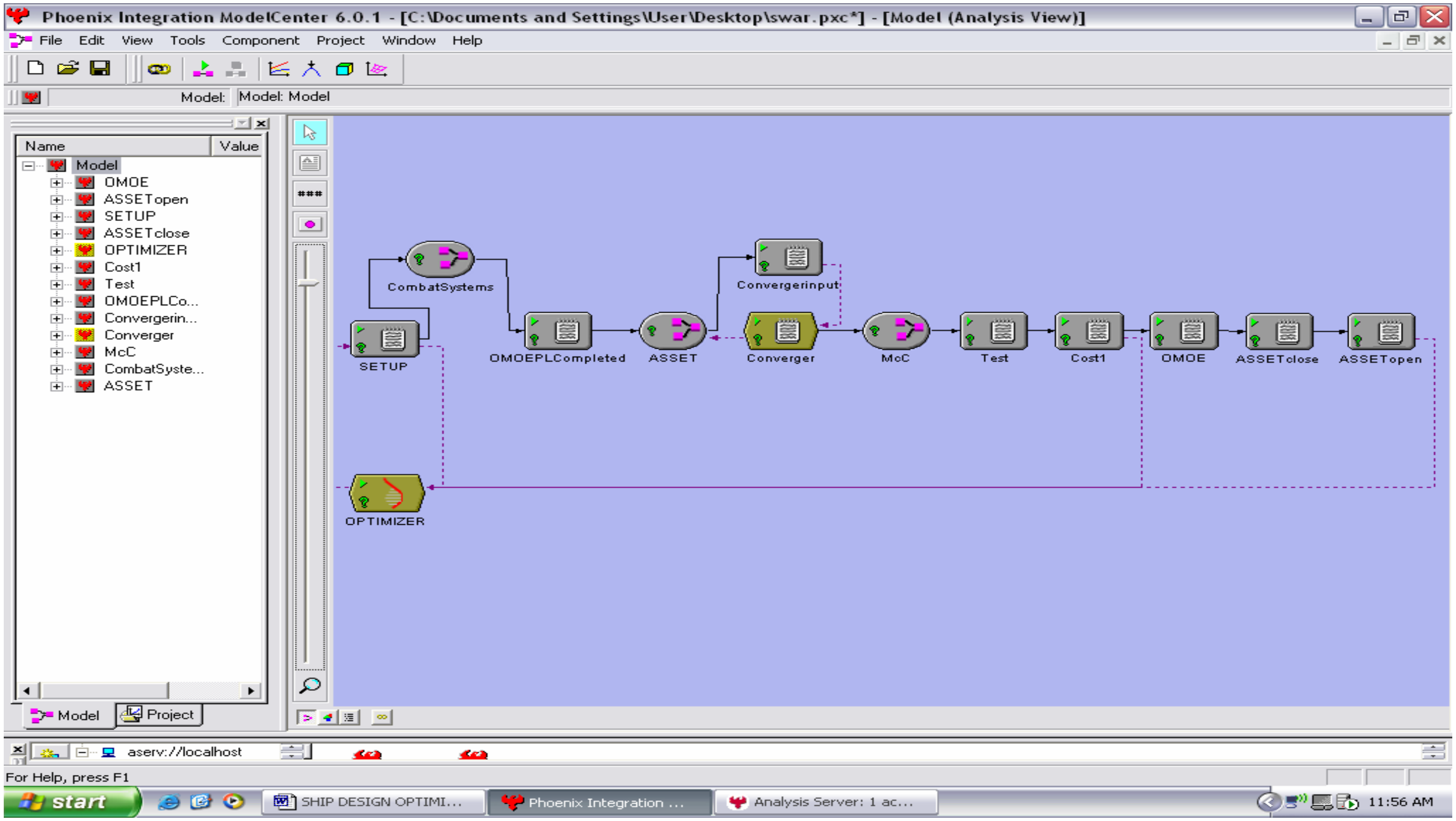


Figure 12. Screen shot showing the DDG51 model in Model Center.

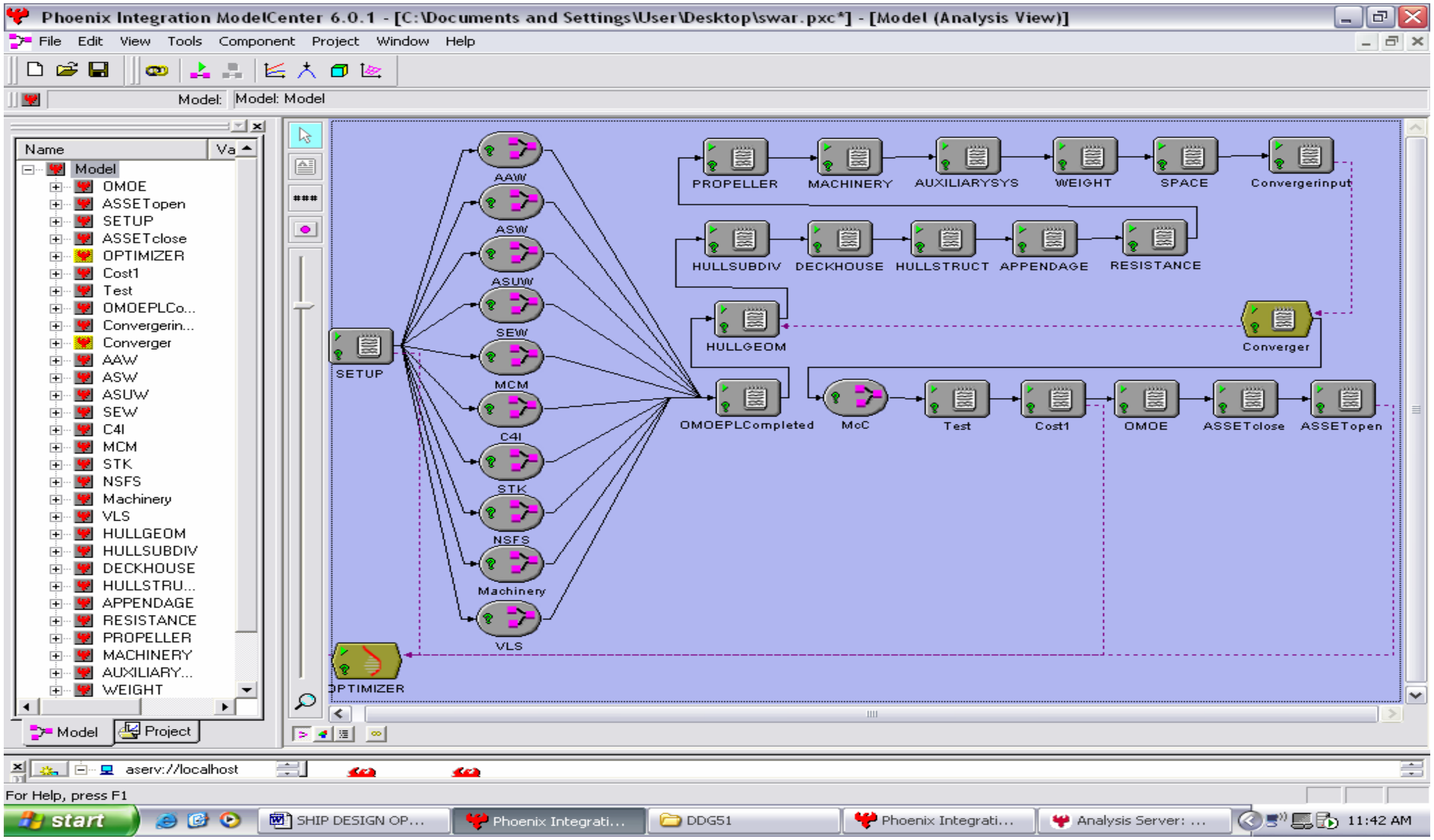


Figure 13. Screen shot showing all the combat system options and Asset modules of DDG51 model in MC.

RESULTS – DDG51 DESIGN OPTIMIZATION

We began the optimization process with a population size of 20 and the GA parameters – crossover and mutation probabilities are chosen to be 1 and 0.05 respectively. Figure 14 shows the Pareto front obtained in 20 generations and Figure 15 shows the progress of the Pareto front for the same problem.

We obtained two more sets of results using population size 30 for 28 generations and using population size 50 for 14 generations. The Pareto designs obtained are shown in figures 16, 17, 18 and 19.

All three sets of results were combined to get a global set of optimum designs. Figure 20 represents this combined set of Pareto designs. In this global set of designs we tried to identify the points that form steps in the Pareto front. As discussed in LHA(R) design optimization, these steps in the Pareto front indicate regions where we can get designs with good improvement in OMOE while there is no much increase in cost.

From the results most of the Pareto designs were observed to have the following combat system characteristics.

AAW option	2
ASUW option	1
ASW option	2
C4I option	1
MCM option	2
NSFS option	1
SEW option	1
VLS option	1
MC option	4

The principal dimensions of most of the Pareto designs were observed to be in the following way.

Prismatic coefficient is around	0.62
Length is around	470 ft.
Breadth/ Draft is around	3
Breadth is around	58 ft.
Depth amidships is around	48 ft.
Mid ship section coefficient is around	0.78

Figure 21 highlights the points that form the steps in the Pareto front. Table 11 shows the design variable values of the designs forming the steps in the Pareto front.

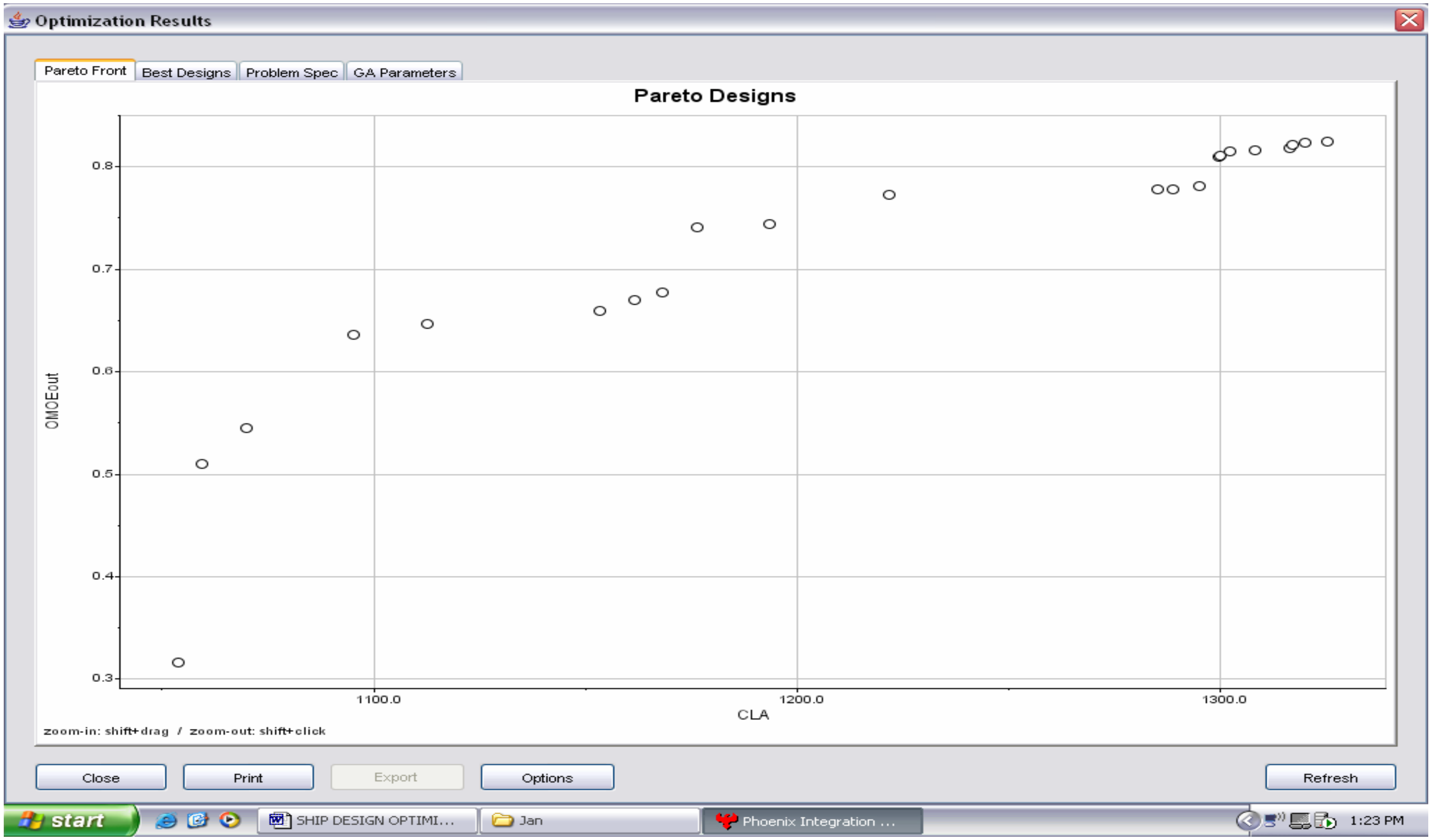


Figure 14. Pareto front obtained using population size 20 in 20 generations.

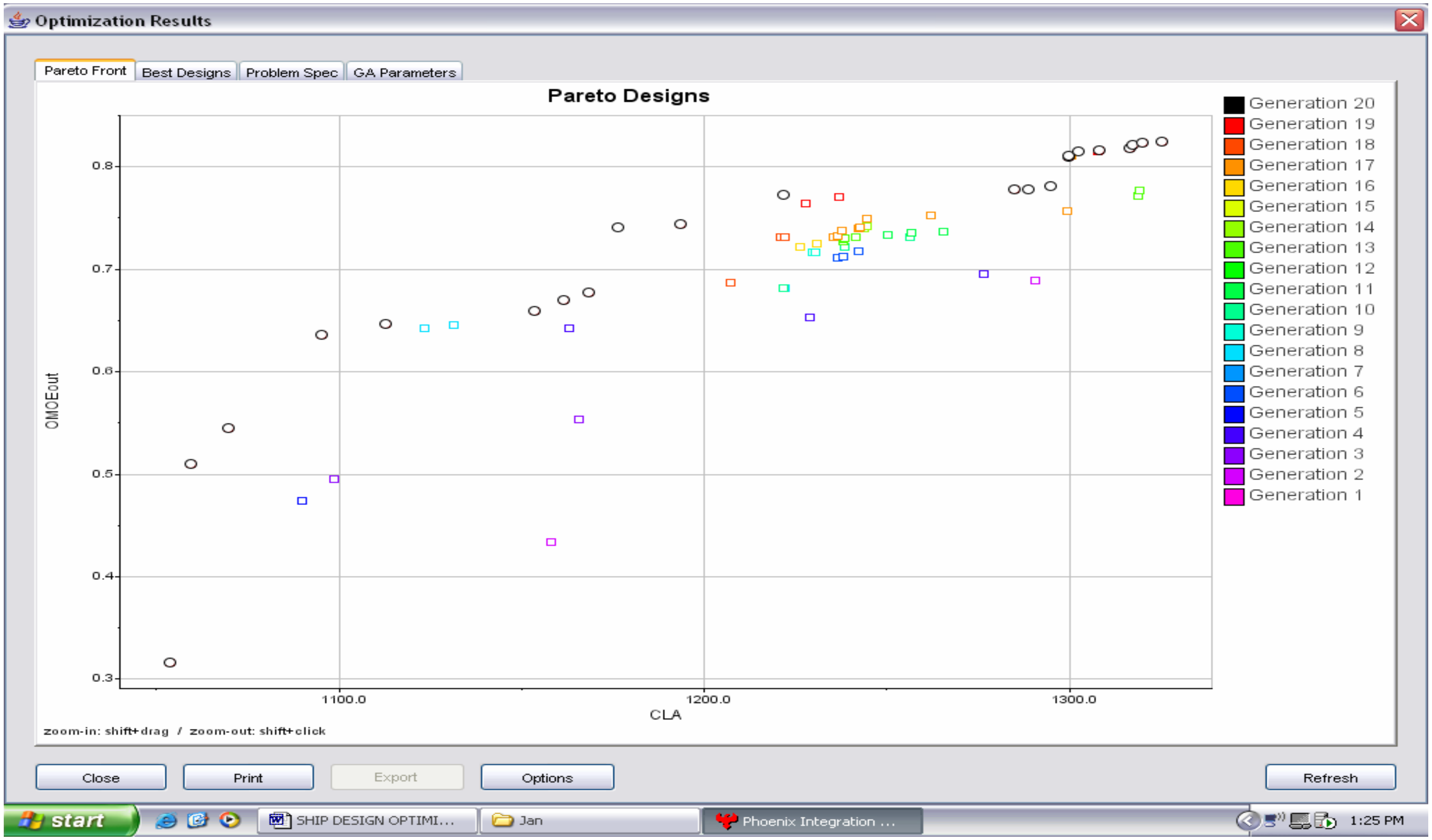


Figure 15. Pareto front development over 20 generations using population size 20.

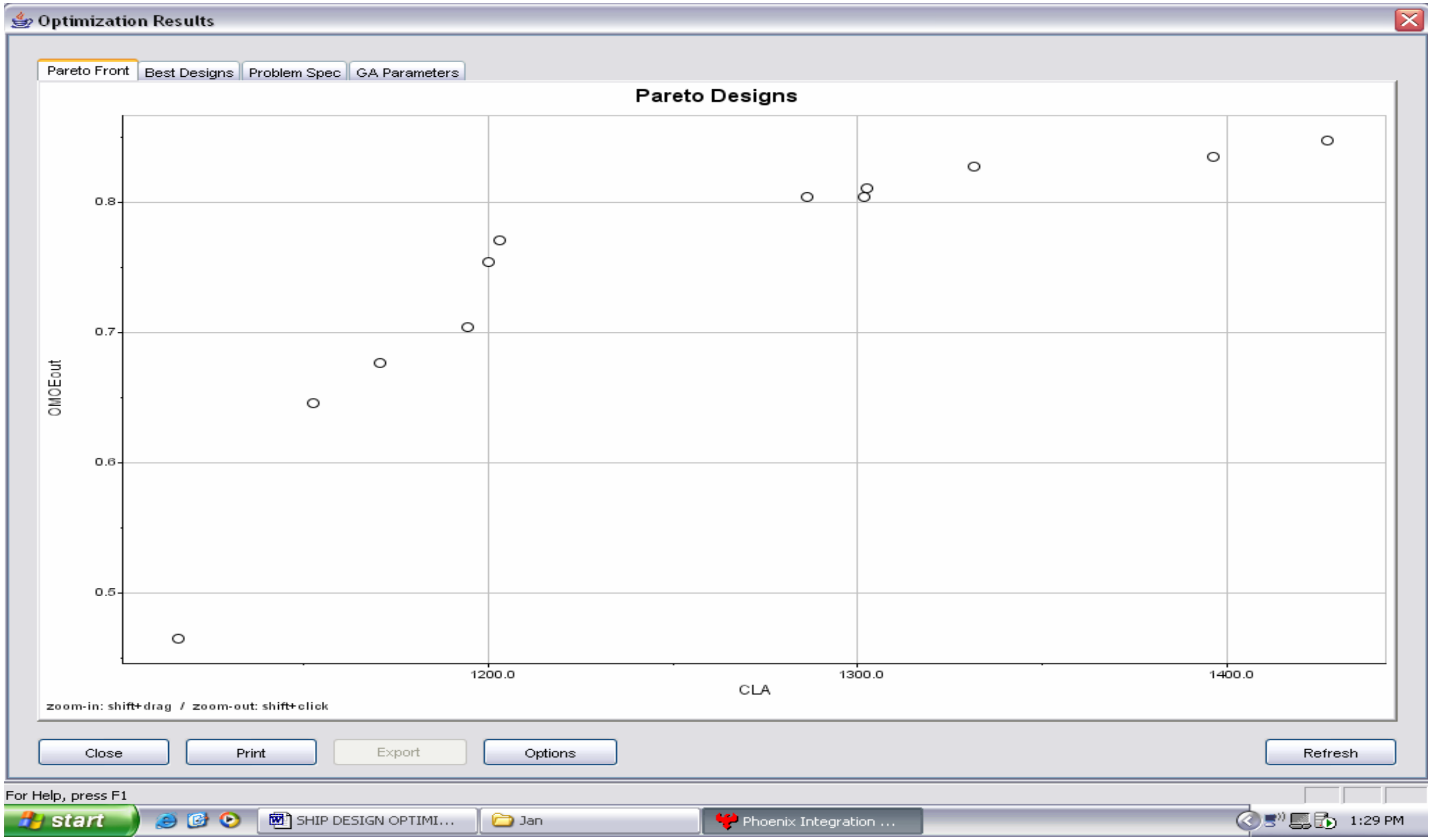


Figure 16. Pareto front obtained using population size 30 in 28 generations.

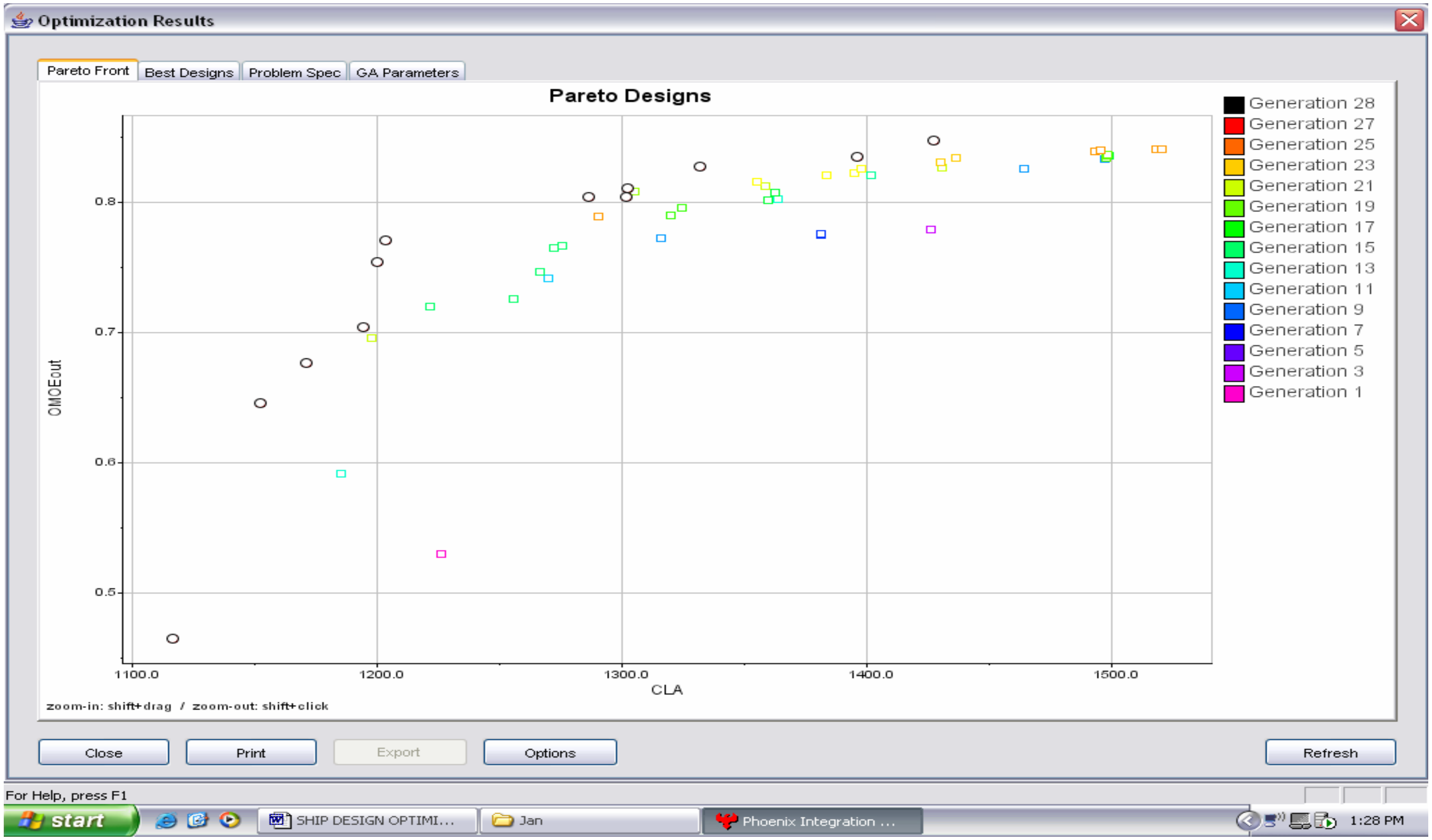


Figure 17. Pareto front development over 28 generations using population size 30.

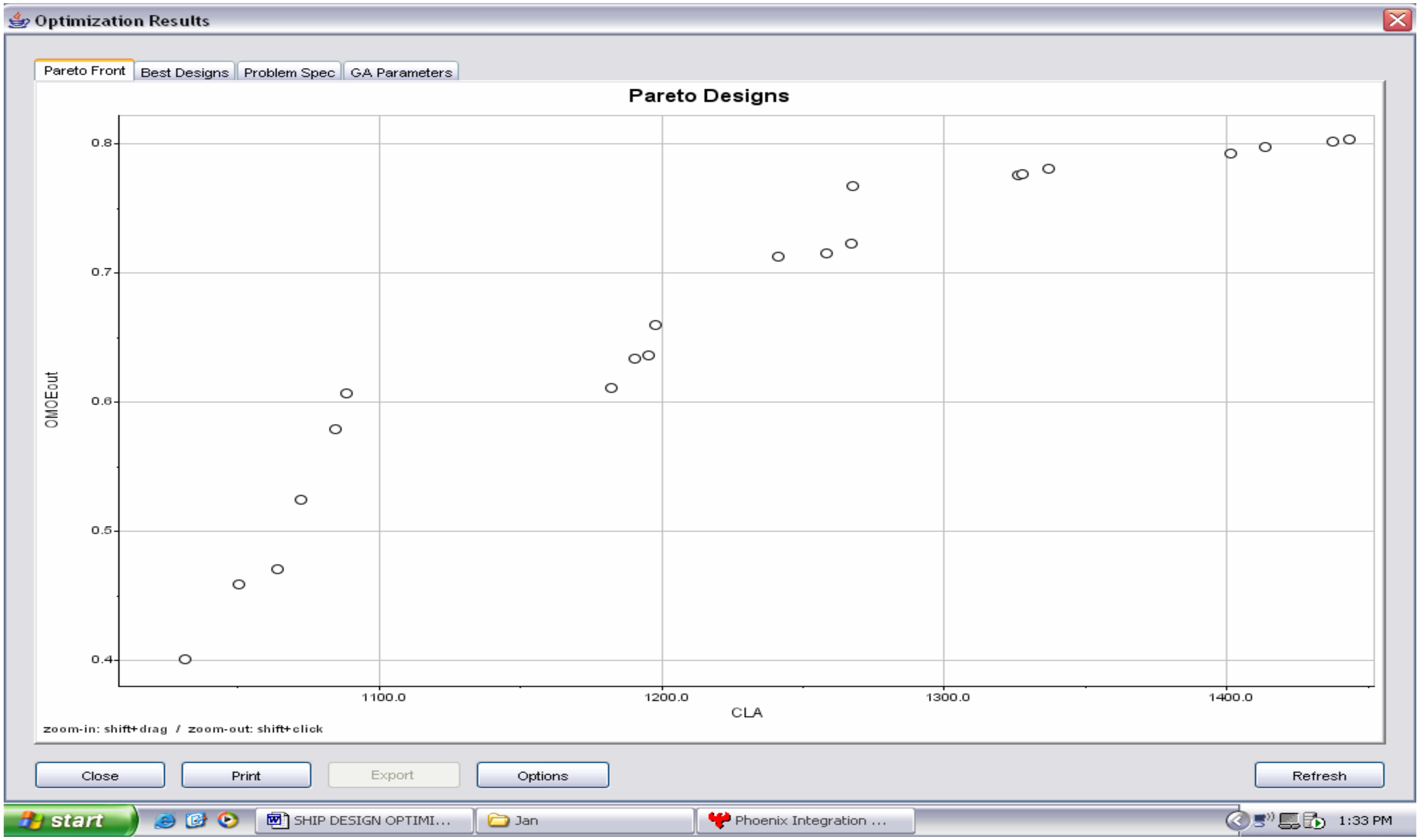


Figure 18. Pareto front obtained using population size 50 in 14 generations.

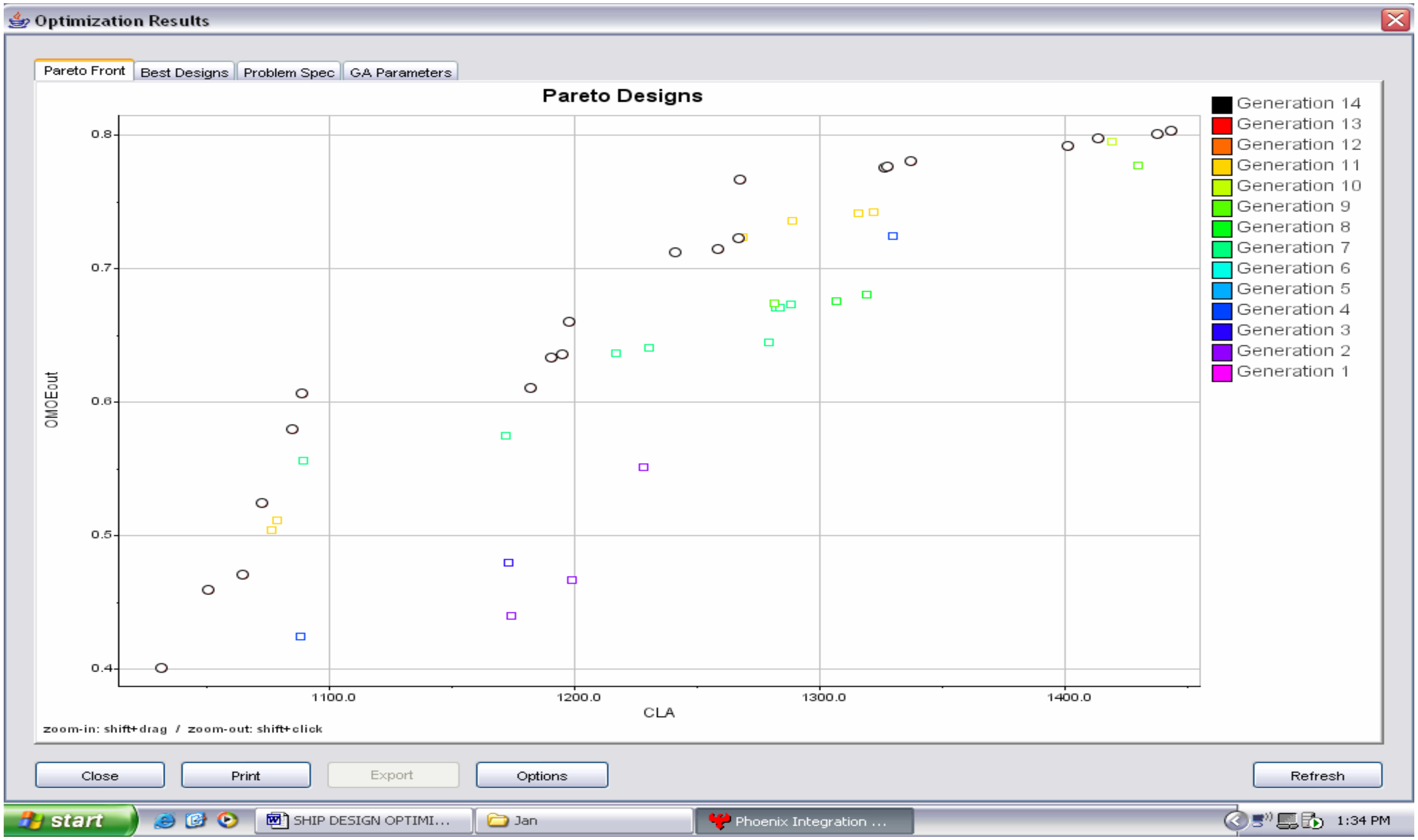


Figure 19. Pareto front development over 14 generations using population size 50.

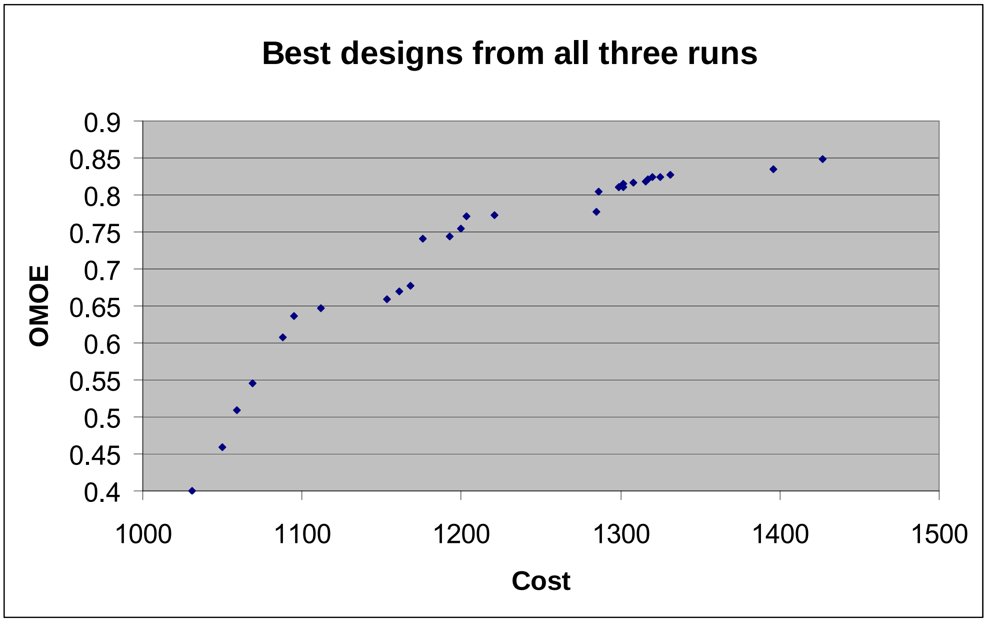


Figure 20. Pareto designs from all three runs

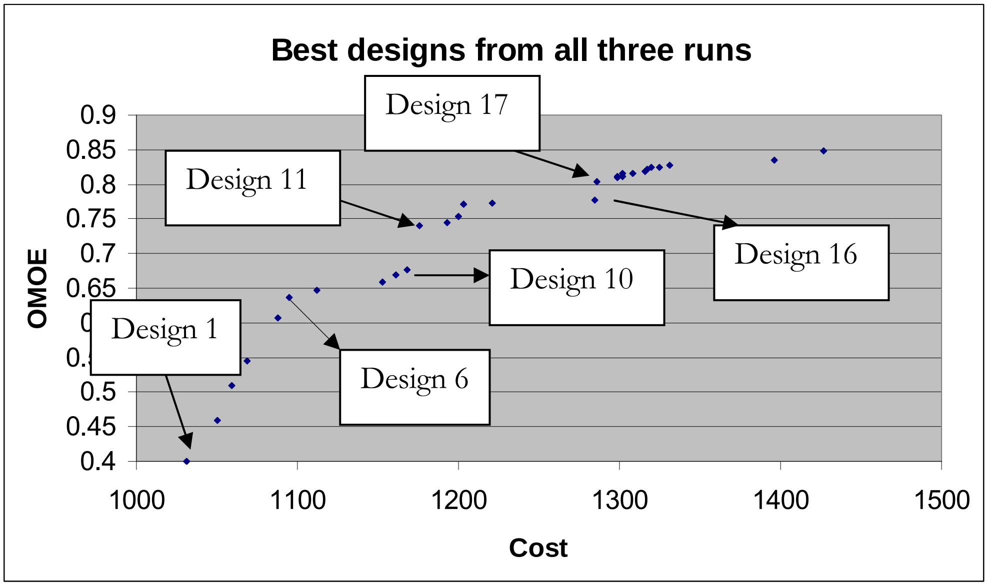


Figure 21. Designs at lower and upper parts of steps in the Pareto front.

Table 11. Design variables and designs forming steps in the Pareto front.

Design Variable	Design 1	Design 6	Design 10	Design 11	Design 16	Design 17
Cp	0.5941	0.614	0.6252	0.6232	0.6268	0.6171
L	558	472	473.3	467.2	457	474.6
B	43.55	42.51	60.77	59.48	59.5	56.15
B/T	2.94	3.02	3	2.99	3.02	3.05
D	49.64	44.73	49..51	48.2	49.12	44.44
Cx	0.816	0.793	0.782	0.783	0.7926	0.8072
AAW	4	4	4	2	3	2
ASUW	2	1	1	1	1	1
ASW	5	1	4	2	2	2
C4I	1	1	1	1	1	1
MCM	2	2	2	2	2	1
NSFS	3	1	1	1	1	1
SEW	2	2	1	1	1	1
VLS	4	4	1	4	1	1
McOption	4	4	4	4	4	4
CLA	1031	1095	1168	1176	1285	1286
OMOE	0.4	0.6365	0.6769	0.7404	0.7776	0.8041

From Table 11 and the VOP values in Table 9 two design variables were found to be of major influence in forming the steps in the Pareto front. They are Anti Air Warfare and Anti Submarine Warfare combat systems. These two are having four and five options respectively. Option 2 of both these systems was found to be more effective.

CONCLUSION

The goals of this thesis work were to get the non-dominated designs for the two types of naval vessels – LHA(R) and DDG51. These designs demonstrate the capability of the ship design optimization process. They provide information that could help the decision maker in evaluating various design options.

The LHA(R) optimization system was developed by integrating DARWIN with the MONOCV version of ASSET. VBScript components, to run various ASSET modules, to apply the trade study option configurations and to calculate the objective functions were developed. Windows script components were developed to access the operating system and invoke ASSET.

The DDG51 optimization system was developed by integrating DARWIN with the MONOLA version of ASSET. VBScript components, to apply various combat system components and to calculate the objective functions were developed. Windows script components were developed to access the operating system and invoke ASSET. For the process to execute without any operating system related problems, the windows script component execution should not be interrupted. Hence the computer is disconnected from the internet and no other programs were executed on the system while running the optimization process. The parsing capability of analysis server was used to transfer the Mccreight index from ASSET to the OMOE component. Darwin was found to stop many times during the run without any error. This created problems in the optimization of DDG51 design but after several runs we were able to get the results. The problem with Darwin was reported to the Phoenix Integration.

For the LHA(R) model, results were obtained by varying the population sizes and number of generations. These results seemed to fall on the same pareto front i.e. converging to sets of designs along the same front. It was observed that two designs having almost the same values of OMOE and LCA can be very similar or very different from each other in design characteristics. The design variables Aviation maintenance stowage and Damage tolerance plating are observed to be of major influence in forming the steps in the pareto front.

We were not able to run the DDG51 model for a large number of generations but the results from the small number of generations seemed to follow the observation made in the case of LHA(R) and fall on to the same non-dominated frontier. From the results the principal dimensions of most of the pareto designs were observed to be around the same set of values. They are Cp: 0.62, Length: 470 ft, B/T: 3, Breadth: 58ft, Depth: 48ft and Cx: 0.78. The Anti air warfare and Anti submarine warfare combat systems were observed to be of major influence in forming the steps in the pareto front.

FUTURE WORK

The information about IPS machinery options was not available and hence not implemented in this project. When the information is available this can be implemented.

Due to problems with DARWIN the DDG51 model was run only for 28 generations. When the problem with Darwin is resolved the DDG51 model can be run for a large number of generations to obtain better results.

In this project we ran the models by fixing the maximum number of generations. Better results may be obtained by running the models until there is no improvement in the pareto front over a number of generations.

APPENDIX

COMBAT SYSTEMS AND THEIR IMPLEMENTATION IN ASSET:

Table A1 describes the combat systems components and their payload characteristics. Tables A2 to A10 represent the option available in each combat system category and their contents. The "Row No." is used to form a connection between these tables. The "Row No." in Table A2 to Table A10 represents the row number, of the corresponding Payload and Adjustments array parameters, in Table A.

Let us say that the optimizer selected option 4 of AAW. From Table A2 we get the constituent components for AAW option 4. From Table A1 we get the Payload and Adjustments table parameters and send them to ASSET. But Table A provides information about only a part of the Payload and Adjustments table (P+A table). We should determine the values corresponding to all the other arrays of P+A table. Almost all the other arrays in P+A table can be provided with default values except the P+A area key table array. The values of this array are represented in ship space classification system array (SSCS array) present in the ASSET documentation. We determine the values of the combat system capabilities by accessing this array manually. Thus values of all array of P+A table are determined and sent to ASSET.

In model center a separate component corresponding to each option of every war area is created. These components contain the information about all the arrays of P+A table. Hence when the optimizer generates the option number all the array options of P+A table will be applied.

Table A1. P+A table array value for various combat system components.

Row No.	P+A NAME TBL	P+A SWBS KEY TBL	P+A WT ADD ARRAY	P+A VCG ADD ARRAY	P+A AREA ADD ARRAY		P+A KW ADD ARRAY	
1	UNIDENTIFIED BALLISTIC PLATING	W164	25.9	20.56	0	0	0	0
2	DDG51 CIC W/12x UYQ-44&2X LSD	W411	17.34	0.22	1989	0	74.5	74.5
3	DATA DIPLAY GROUP BASIC	W411	5.74	12.19	1086	0	45	45
4	INTERFACE EQUIPMENT BASIC	W413	0.3	5.72	50	0	5	5
5	DATA PROCESSING GROUP BASIC	W413	1.47	6.47	210	0	10	10
6	SPS-49 2-D AIR SEARCH RADAR	W452	6.91	17.19	52	0	79	79
7	MK XII AIIMS IFF	W455	2.3	29.2	0	0	3.2	4
8	SPY ID MFAR - SINGLE TRANSMITTER	W456	54.3	14.5	0	1594	269	474.3
9	X-BAND RADAR FOR HORIZON SEARCH	W456	4.11	59	0	0	220.16	220.16
10	S BAND RADAR	W456	138.3	25.5	0	0	6227	10927
11	X-BAND RADAR FOR HORZ AND ABOVE SCH, SD ILLUM	W456	27.2	59.5	0	0	382.7	382.7
12	MK 16 CIWS WEAPON CONTROL SYSTEM	W481	1	14.5	0	464	3.2	10.4
13	MK 92 MFCS - STIR/CORT	W482	4.96	22.35	0	122	50.3	85.8
14	MK 99 GMFCS W/3 SPG-62 ILLUM	W482	14.3	20.9	0	959	34.7	65.2
15	MK 99 GMFCS W/3 BAND ILLUM	W482	0.7	-6.4	0	9	34.7	65.2
16	WEAPON SYSTEM SWITCHBOARDS	W489	2.24	7.28	55	0	4	4
17	COMBAT DF	W495	8.26	21	0	448	15.47	19.34
18	COOLING EQUIPMENT FOR X-BAND RADAR	W532	4.43	-21.81	47.85	0	13.64	13.64
19	COOLING EQUIPMENT FOR S-BAND RADAR	W532	276	-11.81	1731	0	2992	3442
20	COOLING EQUIPMENT FOR LARGE X-BAND RADAR	W532	13.16	-21.81	112	0	32.24	32.24
21	COOLING EQUIPMENT FOR SPY-1D	W532	9	-34	0	960.8	0	0
22	2X MK 16 20MM CIWS [VULCAN-PHALANX]&WORKSHOP	W711	13.2	21	0	321	14	42
23	2X MK31 RAM PDMS	W720	8.2	14	0	536	10	32
24	MK16 20MM CIWS AMMO - 16000 RDS	WF21	8.3	20	0	257	0	0
25	MK36 DLS SRBOC CANNISTERS - 100 RDS	WF21	2.2	13.6	0	0	0	0
26	RIM-116 RAM - 42 RDS	WF22	3.86	14	0	0	0	0
27	SPS-67 SURFACE SEARCH RADAR	W451	1.7	29.6	0	70	8	8
28	SPS-56 SURFACE SEARCH RADAR	W451	0.76	30.11	0	92	7	7
29	AN/SWG-1 HARPOON LNCH CONTROL SYSTEM IN CIC	W482	1.1	14	0	100	0	15
30	2X HARPOON SSM QUAD HCLS [LVL II HARDENED]	W720	5.4	14	0	0	0	15
31	DDG51 SMALL ARMS AND PYRO STOWAGE	W760	5.8	-6.3	203	0	0	0
32	HARPOON MISSILES - 8RDS	WF21	15.8	0	0	0	0	0
33	DDG51 SMALL ARMS AMMO-7.62MM+ 50 CAL +PYRO	WF21	4.1	-6	0	0	0	0
34	SQS-53C 5M BOW SONAR DOME STRUCTURE	W165	85.7	-43.14	0	0	0	0

35	SQS-56 1.5M KEEL SONAR DOME STRUCTURE	W165	7.43	-30.2	0	0	0	0
36	SQQ-28 LAMPS MK III ELECTRONICS	W460	3.4	3	15	0	5.3	5.5
37	PASSIVE SQS-53C SONAR ELEX	W462	21.88	-28.03	650	0	5	5
38	SQR-19 TACTAS [ELEX IN HULL SONAR]	W462	23.3	-25.72	473	0	26.6	26.6
39	SQS-56 1.5M KEEL SONAR DOME ELEX W/SSTD	W463	5.88	-28.3	1340	0	19.7	19.7
40	SQS-53C 5M BOW SONAR DOME ELEX	W463	67.4	-28.3	2870	0	55	55
41	AN/SLQ-25 NIXIE	W473	3.6	-5.72	172	0	3	4.2
42	TORPEDO DECOYS	W473	4.52	-4.89	0	0	0	0
43	UNDERWATER FIRE CONSTROL SYSTEM -BASIC	W483	0.4	8.32	124	0	11.5	11.5
44	ASW CONTROL SYSTEM [ASWCS]	W483	4.8	-11	185	0	19.5	19.5
45	LAMPS MKIII: HELO IN-FLIGHT REFUEL SYS	W542	7.6	-7.35	44	0	1.3	1.3
46	LAMPS MKIII AVIATION FUEL SYS	W542	4.86	-11	30	0	2	2.9
47	LAMPS MKIII RAST/RAST CONTROL/ HELO CONTROL	W588	31.1	-1.6	219	33	4.4	4.4
48	LAMPS MKIII:HELO SECURING SYSTEM	W588	3.6	9.62	0	0	0	0
49	SQS-53C 5M BOW SONAR DOME HULL DAMPING	W636	20.1	-37.07	0	0	0	0
50	LAMPS MKIII AVIATION SHOP AND OFFICE	W665	1.04	-4.5	194	75	0	0
51	2X MK32 SVTT ON DECK	W750	2.7	1.14	0	0	0.6	1.1
52	LAMPS MKIII:HELICOPTER REARM +MAGAZINE	W780	2.7	4.64	212	0	0	4.4
53	LAMPS MKIII 18X MK46 TORP & SONOBUOYS & PYRO	WF22	9.87	4.8	0	588	0	0
54	LAMPS MKIII 2 X SH-60B HELOS AND HANGER (BASED)	WF23	12.73	4.5	0	3406	5.6	5.6
55	LAMPS MKIII AVIATION SUPPORT AND SPARES	WF26	9.42	5	357	0	0	0
56	BATHY THERMOGRAPH PROBES	WF29	0.2	-16.11	0	0	0	0
57	LAMPS MKIII: AVIATION FUEL [JP-5]	WF42	64.4	-28.81	0	0	0	0
58	DDG51 EXCOMM	W440	32.3	-7.9	1270	95	93.3	96.4
59	RADIO SYSTEMS	W441	9.46	17.5	608	100	73	73
60	VISUAL AND AUDIBLE SYSTEMS-BASIC	W443	0.38	25.79	103	0	30	30
61	TTY&FACSIMILE SYSTEMS - BASIC	W445	1.12	12.78	50	0	10	10
62	SECURITY EQUIPMENT SYSTEMS - BASIC	W446	2.66	14.18	65	0	20	20
63	MINE AVOIDANCE SONAR	W462	11.88	-18.03	350	0	5	5
64	MINE HUNTING SIDE SCAN SONAR	W473	1.6	-5.72	72	0	1.3	1.3
65	MINE HUNTING AUV/ REMOTE MINE HUNTING SYSTEM	W473	3.6	-5.72	172	0	3	4.2
66	VFAS HY-60 ARMOR LEVEL II	W164	3	-10.3	0	0	0	0
67	MK45 5IN/54 GUN HY-80 ARMOR	W164	20.2	-0.35	0	0	0	0
68	MK92- GFCS	W481	7.18	-5.6	0	168	6	15.4
69	VGAS GFCS	W481	3.32	-6	0	0	9.84	11.77
70	NATACMS WEAPON CONTROL SYSTEM	W481	1	14.5	0	464	3.2	10.4
71	1X MK 45 5IN/54 GUN [HAND SD]	W710	36.8	1.44	285	0	36.6	50.2

72	VGAS GFCS	W711	30.2	-10.7	533	0	10	34
73	MK45 5IN AMMO- 600 RDS	WF21	33.1	-10.75	705	0	0	0
74	VGAS AMMUNITION	WF21	17.6	-11	0	0	0	0
75	MK45 5IN AMMO W/ERGM - 600RDS	WF21	36.1	-10.75	705	0	0	0
76	SLQ-32[V]2 PASSIVE ECM	W472	3	21.5	40	132	6.4	6.4
77	SLQ-32[V]3ECM	W472	11.61	20.6	40	300	6.4	87
78	MK36 DLS W/4 LAUNCHERS	W474	1	13.6	0	0	2.4	2.4
79	TOMAHAWK WEAPON CONTROL SYSTEM (IN CSER)	W482	5.6	-7.8	5	0	11.5	11.5
80	64 CELL VLS ARMOR - LEVEL III HY-80	W164	21.1	-6.17	0	0	0	0
81	32 CELL VLS ARMOR - LEVEL III HY-80	W164	14	-6.17	0	0	0	0
82	VLS WEAPON CONTROL SYSTEM (1 PER MODULE)	W482	0.7	-9.66	56	0	15	18
83	64 CELL MAGAZINE DEWATERING SYSTEM	W529	1.5	-6.97	0	0	0	0
84	32 CELL MAGAZINE DEWATERING SYSTEM	W529	3	-6.97	0	0	0	0
85	MK41 VLS 64CELL	W721	147.8	-13.66	2245	0	63.4	63.4
86	MK41 VLS 32CELL	W721	82.8	-7.97	1123	0	31.1	31.1
87	VLS WEAPONS HANDLING (USES 3 CELLS)	W722	1	-5.86	0	0	0	0
88	VLS MISSILES -61	WF21	93.8	-11.06	0	0	0	0
89	VLS MISSILES - 64	WF21	98.4	-11.06	0	0	0	0
90	VLS MISSILES -32	WF21	49.2	-5.37	0	0	0	0
91	ADVANCED GUN SYSTEM	W710	36.8	-2.44	285	0	36.6	50.2
92	2X MK16 20MM CIWS (PHALANX BLOCK 1B)&WORKSHOP	W711	13.2	21	0	321	14	42
93	SLAM ER MISSILES X 8	WF21	5	0	0	0	0	0
94	RIM-7M SEA SPARROW MISSILES X 8	WF21	1.79	0	0	0	0	0
95	SM-1 STANDARD MISSILE X 8	WF21	3.93	0	0	0	0	0
96	SM-2 STANDARD MISSILE (MEDIUM RANGE) X 8	WF21	4.93	0	0	0	0	0
97	SM-2 STANDARD MISSILE (EXTENDED RANGE) X 8	WF21	10.64	0	0	0	0	0
98	MK-50 ADCAP TORPEDOS X 8	WF21	2.68	0	0	0	0	0
99	VERTICAL LAUNCH ARSOC (VLA) MISSILES X 8	WF21	5.03	0	0	0	0	0
100	ADVANCED C4I SYSTEM	W440	32.3	-7.9	1270	95	93.3	96.4
101	AIEWS ADVANCED SEW SYSTEM	W472	3	21.5	40	132	6.4	6.4
102	DDG51 NAVIGATION SYSTEM	W420	7.5	16.1	0	50	16.4	20.5

Table A2. AAW options and their constituting components

Warfare Area option	Components of warfare area option	Row No.
AAW – option 1	DDG51 CIC W/12x UYQ-44&2X LSD	2
	MK XII AIIMS IFF	7
	S BAND RADAR	10
	X-BAND RADAR FOR HORZ AND ABOVE SCH, SD ILLUM	11
	MK 99 GMFCS W/3 BAND ILLUM	15
	COMBAT DF	17
	COOLING EQUIPMENT FOR S-BAND RADAR	19
	COOLING EQUIPMENT FOR LARGE X-BAND RADAR	20
	MK36 DLS SRBOC CANNISTERS - 100 RDS	25
	MK 16 CIWS WEAPON CONTROL SYSTEM	12
	2X MK 16 20MM CIWS [VULCAN-PHALANX]&WORKSHOP	22
	MK16 20MM CIWS AMMO - 16000 RDS	24
AAW – option 2	DDG51 CIC W/12x UYQ-44&2X LSD	2
	MK XII AIIMS IFF	7
	SPY ID MFAR - SINGLE TRANSMITTER	8
	X-BAND RADAR FOR HORZ AND ABOVE SCH, SD ILLUM	11
	MK 99 GMFCS W/3 SPG-62 ILLUM	14
	COMBAT DF	17
	COOLING EQUIPMENT FOR SPY-1D	21
	COOLING EQUIPMENT FOR LARGE X-BAND RADAR	20
	MK36 DLS SRBOC CANNISTERS - 100 RDS	25
	MK 16 CIWS WEAPON CONTROL SYSTEM	12
	2X MK 16 20MM CIWS [VULCAN-PHALANX]&WORKSHOP	22
	MK16 20MM CIWS AMMO - 16000 RDS	24
AAW – option 3	DDG51 CIC W/12x UYQ-44&2X LSD	2
	MK XII AIIMS IFF	7
	SPY ID MFAR - SINGLE TRANSMITTER	8
	X-BAND RADAR FOR HORIZON SEARCH	9
	MK 16 CIWS WEAPON CONTROL SYSTEM	12
	MK 99 GMFCS W/3 SPG-62 ILLUM	14
	COMBAT DF	17
	COOLING EQUIPMENT FOR X-BAND RADAR	18
	COOLING EQUIPMENT FOR SPY-1D	21
	2X MK 16 20MM CIWS [VULCAN-PHALANX]&WORKSHOP	22
	MK16 20MM CIWS AMMO - 16000 RDS	24
	MK36 DLS SRBOC CANNISTERS - 100 RDS	25

AAW – option 4	DATA DIPLAY GROUP BASIC	3
	INTERFACE EQUIPMENT BASIC	4
	DATA PROCESSING GROUP BASIC	5
	SPS-49 2-D AIR SEARCH RADAR	6
	MK XII AIIMS IFF	7
	MK 92 MFCS - STIR/CORT	13
	WEAPON SYSTEM SWITCHBOARDS	16
	MK36 DLS SRBOC CANNISTERS - 100 RDS	25
	X-BAND RADAR FOR HORIZON SEARCH	9
	COOLING EQUIPMENT FOR X-BAND RADAR	18
	MK 16 CIWS WEAPON CONTROL SYSTEM	12
	2X MK 16 20MM CIWS [VULCAN-PHALANX]&WORKSHOP	22
	MK16 20MM CIWS AMMO - 16000 RDS	24

Table A3. ASUW options and their constituting components

Warfare Area option	Components of warfare area option	Row No.
ASUW – option 1	SPS-67 SURFACE SEARCH RADAR	27
	DDG51 SMALL ARMS AND PYRO STOWAGE	31
	DDG51 SMALL ARMS AMMO- 7.62MM+ 50 CAL +PYRO	33
ASUW – option 2	SPS-56 SURFACE SEARCH RADAR	28
	DDG51 SMALL ARMS AND PYRO STOWAGE	31
	DDG51 SMALL ARMS AMMO- 7.62MM+ 50 CAL +PYRO	33

Table A4. ASW options and their constituting components

Warfare Area option	Components of warfare area option	Row No.
ASW – option 1	SQS-53C 5M BOW SONAR DOME STRUCTURE	34
	SQQ-28 LAMPS MK III ELECTRONICS	36
	PASSIVE SQS-53C SONAR ELEX	37

	SQS-53C 5M BOW SONAR DOME ELEX	40
	AN/SLQ-25 NIXIE	41
	TORPEDO DECOYS	42
	LAMPS MKIII: HELO IN-FLIGHT REFUEL SYS	45
	LAMPS MKIII AVIATION FUEL SYS	46
	LAMPS MKIII RAST/RAST CONTROL/ HELO CONTROL	47
	LAMPS MKIII:HELO SECURING SYSTEM	48
	SQS-53C 5M BOW SONAR DOME HULL DAMPING	49
	LAMPS MKIII AVIATION SHOP AND OFFICE	50
	2X MK32 SVTT ON DECK	51
	LAMPS MKIII 18X MK46 TORP & SONOBUOYS & PYRO	53
	LAMPS MKIII 2 X SH-60B HELOS AND HANGER (BASED)	54
	LAMPS MKIII AVIATION SUPPORT AND SPARES	55
	BATHY THERMOGRAPH PROBES	56
	LAMPS MKIII: AVIATION FUEL [JP-5]	57
ASW – option 2	SQS-53C 5M BOW SONAR DOME STRUCTURE	34
	SQQ-28 LAMPS MK III ELECTRONICS	36
	SQS-53C 5M BOW SONAR DOME ELEX	40
	AN/SLQ-25 NIXIE	41
	UNDERWATER FIRE CONSTROL SYSTEM -BASIC	43
	LAMPS MKIII: HELO IN-FLIGHT REFUEL SYS	45
	LAMPS MKIII AVIATION FUEL SYS	46
	LAMPS MKIII RAST/RAST CONTROL/ HELO CONTROL	47
	LAMPS MKIII:HELO SECURING SYSTEM	48
	SQS-53C 5M BOW SONAR DOME HULL DAMPING	49
	LAMPS MKIII AVIATION SHOP AND OFFICE	50
	2X MK32 SVTT ON DECK	51
	LAMPS MKIII 18X MK46 TORP & SONOBUOYS & PYRO	53
	LAMPS MKIII 2 X SH-60B HELOS AND HANGER (BASED)	54
	LAMPS MKIII AVIATION SUPPORT AND SPARES	55
	BATHY THERMOGRAPH PROBES	56
	LAMPS MKIII: AVIATION FUEL [JP-5]	57
ASW – option 3	SQS-53C 5M BOW SONAR DOME STRUCTURE	34
	SQR-19 TACTAS [ELEX IN HULL SONAR]	38
	SQS-53C 5M BOW SONAR DOME ELEX	40
	AN/SLQ-25 NIXIE	41
	ASW CONTROL SYSTEM [ASWCS]	44
	LAMPS MKIII AVIATION FUEL SYS	46
	LAMPS MKIII:HELO SECURING SYSTEM	48
	SQS-53C 5M BOW SONAR DOME HULL DAMPING	49

	2X MK32 SVTT ON DECK	51
	LAMPS MKIII:HELICOPTER REARM +MAGAZINE	52
	BATHY THERMOGRAPH PROBES	56
	LAMPS MKIII: AVIATION FUEL [JP-5]	57
ASW – option 4	SQS-53C 5M BOW SONAR DOME STRUCTURE	34
	PASSIVE SQS-53C SONAR ELEX	37
	SQR-19 TACTAS [ELEX IN HULL SONAR]	38
	AN/SLQ-25 NIXIE	41
	ASW CONTROL SYSTEM [ASWCS]	44
	LAMPS MKIII AVIATION FUEL SYS	46
	LAMPS MKIII:HELO SECURING SYSTEM	48
	SQS-53C 5M BOW SONAR DOME HULL DAMPING	49
	2X MK32 SVTT ON DECK	51
	LAMPS MKIII:HELICOPTER REARM +MAGAZINE	52
	BATHY THERMOGRAPH PROBES	56
	LAMPS MKIII: AVIATION FUEL [JP-5]	57
ASW – option 5	SQS-56 1.5M KEEL SONAR DOME STRUCTURE	35
	SQS-56 1.5M KEEL SONAR DOME ELEX W/SSTD	39
	AN/SLQ-25 NIXIE	41
	ASW CONTROL SYSTEM [ASWCS]	44
	LAMPS MKIII AVIATION FUEL SYS	46
	LAMPS MKIII:HELO SECURING SYSTEM	48
	2X MK32 SVTT ON DECK	51
	LAMPS MKIII:HELICOPTER REARM +MAGAZINE	52
	BATHY THERMOGRAPH PROBES	56
	LAMPS MKIII: AVIATION FUEL [JP-5]	57

Table A5. C4I options and their constituting components

Warfare Area option	**Components of warfare area option**	**Row No.**
C4I – option 1	ADVANCED C4I SYSTEM	100
C4I – option 2	DDG51 EXCOMM	58

Table A6. MCM options and their constituting components

Warfare Area option	Components of warfare area option	Row No.
MCM – option 1	MINE AVOIDANCE SONAR	63
	MINE HUNTING SIDE SCAN SONAR	64
	MINE HUNTING AUV/ REMOTE MINE HUNTING SYSTEM	65
MCM – option 2	MINE AVOIDANCE SONAR	63
	MINE HUNTING SIDE SCAN SONAR	64
MCM – option 3	MINE AVOIDANCE SONAR	63

Table A7. NSFS options and their constituting components

Warfare Area option	Components of warfare area option	Row No.
NSFS – option 1	VFAS HY-60 ARMOR LEVEL II	66
	VGAS GFCS	69
	NATACMS WEAPON CONTROL SYSTEM	70
	VGAS GFCS	72
	VGAS AMMUNITION	74
NSFS – option 2	MK45 5IN/54 GUN HY-80 ARMOR	67
	MK92- GFCS	68
	NATACMS WEAPON CONTROL SYSTEM	70
	1X MK 45 5IN/54 GUN [HAND SD]	71
	MK45 5IN AMMO W/ERGM - 600RDS	75
NSFS – option 3	MK45 5IN/54 GUN HY-80 ARMOR	67
	MK92- GFCS	68
	1X MK 45 5IN/54 GUN [HAND SD]	71
	MK45 5IN AMMO W/ERGM - 600RDS	75

Table A8. SEW options and their constituting components

Warfare Area option	Components of warfare area option	Row No.
SEW – option 1	AIEWS ADVANCED SEW SYSTEM MK36 DLS W/4 LAUNCHERS	101 78
SEW – option 2	SLQ-32[V]3ECM MK36 DLS W/4 LAUNCHERS	77 78
SEW – option 3	SLQ-32[V]2 PASSIVE ECM MK36 DLS W/4 LAUNCHERS	76 78

Table A9. STK options and their constituting components

Warfare Area option	Components of warfare area option	Row No.
STK – option 1	TOMAHAWK WEAPON CONTROL SYSTEM (IN CSER)	79

Table A10. VLS options and their constituting components

Warfare Area option	Components of warfare area option	Row No.
VLS – option 1	64 CELL VLS ARMOR - LEVEL III HY-80 64 CELL VLS ARMOR - LEVEL III HY-80 VLS WEAPON CONTROL SYSTEM (1 PER MODULE) 32 CELL MAGAZINE DEWATERING SYSTEM 32 CELL MAGAZINE DEWATERING SYSTEM MK41 VLS 64CELL MK41 VLS 64CELL VLS WEAPONS HANDLING (USES 3 CELLS) VLS MISSILES -61 VLS MISSILES - 64	80 80 82 84 84 85 85 87 88 89

VLS – option 2	64 CELL VLS ARMOR - LEVEL III HY-80	80
	32 CELL VLS ARMOR - LEVEL III HY-80	81
	VLS WEAPON CONTROL SYSTEM (1 PER MODULE)	82
	64 CELL MAGAZINE DEWATERING SYSTEM	83
	32 CELL MAGAZINE DEWATERING SYSTEM	84
	MK41 VLS 64CELL	85
	MK41 VLS 32CELL	86
	VLS WEAPONS HANDLING (USES 3 CELLS)	87
	VLS MISSILES -61	88
	VLS MISSILES -32	90
VLS – option 3	64 CELL VLS ARMOR - LEVEL III HY-80	80
	VLS WEAPON CONTROL SYSTEM (1 PER MODULE)	82
	32 CELL MAGAZINE DEWATERING SYSTEM	84
	MK41 VLS 64CELL	85
	VLS MISSILES - 64	89
VLS – option 4	32 CELL VLS ARMOR - LEVEL III HY-80	81
	VLS WEAPON CONTROL SYSTEM (1 PER MODULE)	82
	64 CELL MAGAZINE DEWATERING SYSTEM	83
	MK41 VLS 32CELL	86
	VLS MISSILES -32	90

SCRIPTS ASSET

VBScript to get access to ASSET and to its Model parameter list:

```
Dim assetExecutive
Set assetExecutive = CreateObject("Asset.Executive")

Dim assetShipType
Set assetShipType = assetExecutive.GetShipType

Dim assetCommands
Set assetCommands = assetShipType.GetCommands

Dim assetMPL
Set assetMPL = assetShipType.getModelParameterList
```

Sending commands to ASSET:

```
assetCommands.SendCommand "USE, DDG51 BL"
```

Sending a value to ASSET by converting it to string

```
L = CStr(Length)           ‘// convert the value of length to string
L = "SET,LBP, " & L   ‘// Concatenate string to the command
assetCommands.SendCommand L  ‘//Send the new string as command.
```

To check the occurrence of any fatal errors in ASSET

If there is a fatal error in running any module ASSET returns 1 for iError otherwise ASSET returns 0 for iError.

```
iError = assetShipType.getError
if (iError =0) then
      HULLGEOMDONE = true
end if
```

To Exit ASSET

```
assetCommands.SendCommand "EXIT"
assetCommands.Wait
assetCommands.SendCommand "N"
assetCommands.Wait
assetCommands.SendCommand "Y"
```

To open ASSET:

Make a shortcut key to invoke ASSET. We used “CTRL+ALT+A” as the short cut. Go to Phoenix integration’s knowledge base website and install the PHXSleep.dll and then use the following windows script.

```
Dim shell
set sl = createobject("PHXSleep.Sleeper")
Set shell = CreateObject("WScript.Shell")

sl.phxSleep 2000           ‘// Wait for 2000 milli seconds or 2 seconds
shell.Sendkeys "^%a"       ‘//  Send keys “CTRL(^) + ALT(%) + A (a)”
sl.phxSleep 1000           ‘ // Wait for ASSET to open.
```

REFERENCES

[1]. Laverghetta T.., Brown A.J.,“ Dynamics of Naval Ship Design: A Systems Approach”, Naval Engineers Journal, Vol.111, No.2, pp. 307-324, May 1999.

[2]. Brown A.J., Thomas M, “Reengineering the Naval Ship Concept Design Process”, From Research to Reality in Ship Systems Engineering Symposium, ASNE, September 1998.

[3]. Brown A.J., Salcedo J., “Multiple Objective Genetic Optimization In Naval Ship Design”, Naval Engineers Journal, Vol.115, No.4, pp. 49-61, Fall 2003.

[4]. Garrett N. Vanderplaats, “Numerical optimization techniques for engineering design”, McGraw Hill, N.Y., 1984.

[5]. Zafer Gurdal, “Engineering Design optimization class notes”, Virginia Tech AOE Department, 2003.

[6]. Phoenix Integration, DARWIN “Darwin_Plug_ In_Help.html”, version 1.0.

[7]. NSWC, ASSET “ User Manual”, Version- 5.0, David Taylor Model Basin, Carderock Division, Naval Surface Warfare Center, May 1990.

[8]. Phoenix Integration, MODEL CENTER “Model Center help”, version 6.0.

[9]. Brown A.J., Wayne Neu, Swaroop N. Neti, Ben Young, Matthew Garner, “Optimization and overall measure of effectiveness (OMOE) for the preliminary design of LHA(R) – NSWC CISD Innovation cell project report”, 2004.

[10]. Brown A.J., “Ship Design Lecture Notes”, Virginia Tech AOE Department, 2004.

[11]. Brown A.J., “Mathcad version of LHA(R) cost Function”, Virginia Tech, AOE Department.

[12]. Brown A.J. “Generating Operational Measures of Effectiveness for the preliminary design of LHA(R)”, Virginia Tech, AOE Department.

[13]. Brown A.J., “Mathcad version of LHA(R) Effectiveness function”, Virginia Tech, AOE Department.

[14]. Phoenix Integration, “Knowledge base”,http://www.Phoenix-int.com.

[15]. http://peoships.crane.navy.mil.

[16]. Brown A.J., “Mathcad version of DDG51 Effectiveness function”, Virginia Tech, AOE Department.

[17]. Brown A.J., “Propulsion machinery options for DDG51 and their configuration”, Virginia Tech, AOE Department.