

Intelligent Ship Arrangement (ISA) Passage Variable Lattice Network Studies and Results

Anthony S. Daniels¹, Farah Tahmasbi², and David J. Singer³

ABSTRACT

The Intelligent Ship Arrangements (ISA) system, under development at the University of Michigan, is a Leading Edge Architecture for Prototyping System (LEAPS) compatible software system that assists the designer in developing rationally-based arrangements that satisfy the design specific needs as well as general Navy requirements and standard practices to the maximum extent practicable. This software system is intended to be used following or as a latter part of Advanced Ship and Submarine Evaluation Tool (ASSET) synthesis. Recent improvements to the current ISA application are discussed. The issues covered are: modifications to the objective function and results from a newly developed ISA demonstration code.

KEY WORDS

Agents; genetic algorithms; hybrid; optimization; combinatorial optimization; general arrangements; facilities planning

INTRODUCTION

In Parsons et al. (2008), the ISA platform was introduced as a software package for the quantification and optimization of general arrangements in surface ships. It is a native C++ / Microsoft Foundation Class (MFC) application, which is intended to be used as a post processing step to, or as a latter phase, of the U.S. Navy's ASSET synthesis process (ASSET 2007). This allows the users to gain insight into the general arrangements of the vessel at a much earlier stage in the design process during the Analysis of Alternatives level of design. Generating arrangements earlier in the ship design process opens up the opportunity to, in turn, perform more detailed analyses (such as survivability) earlier. ISA provides four new design enabling capabilities to the naval architecture community as well as an overall paradigm shift in general arrangements theory and practice. It provides:

- The ability to capture U.S. Navy design rules, regulations, best practices, and intent in a quantifiable and consistent manner using the ship specific template databases (e.g. NAVSEA 1992 requirements). This provides an important knowledge capture capability to the naval architecture community.
- The ability to quantify and compare general arrangements of vessels in a rational process.
- The ability to apply that rational process to the improvement and optimization of the general arrangements for a ship design.
- The ability to directly integrate general arrangements trades studies into the Analysis of Alternatives level of design.

ISA gets its inputs from an ASSET generated LEAPS database (LEAPS 2006), and a ship specific template library (Figure 1). The ASSET / LEAPS database provides the ship geometry as well as manning, major components, etc. The ship specific template library database, or ISA Library, provides a set of template spaces and accompanying constraints for that specific ship type to be used in the population of the model. From these two sources of information the ship model is created and the optimization on the general arrangements for the vessel can be performed. Resulting arrangements can then be exported back into the LEAPS database as new Concept objects.

¹ Research Engineer, Department of Naval Architecture and Marine Engineering, University of Michigan

² Graduate Students, Computer Science, University of Michigan

³ Assistant Research Scientist, Department of Naval Architecture and Marine Engineering, University of Michigan

Report Documentation Page				Form Approved OMB No. 0704-0188	
Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.					
1. REPORT DATE 2010		2. REPORT TYPE		3. DATES COVERED 00-00-2010 to 00-00-2010	
4. TITLE AND SUBTITLE Intelligent Ship Arrangement (ISA) Passage Variable Lattice Network Studies and Results				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S)				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Research Engineer, Department of Naval Architecture and Marine Engineering, University of Michigan, Ann Arbor, MI				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited					
13. SUPPLEMENTARY NOTES ASNE Day - Engineering the Affordable Global Navy Through Innovation Location Arlington, VA held on April 8-9, 2010					
14. ABSTRACT The Intelligent Ship Arrangements (ISA) system, under development at the University of Michigan, is a Leading Edge Architecture for Prototyping System (LEAPS) compatible software system that assists the designer in developing rationallybased arrangements that satisfy the design specific needs as well as general Navy requirements and standard practices to the maximum extent practicable. This software system is intended to be used following or as a latter part of Advanced Ship and Submarine Evaluation Tool (ASSET) synthesis. Recent improvements to the current ISA application are discussed. The issues covered are: modifications to the objective function and results from a newly developed ISA demonstration code.					
15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT Same as Report (SAR)	18. NUMBER OF PAGES 17	19a. NAME OF RESPONSIBLE PERSON
a. REPORT unclassified	b. ABSTRACT unclassified	c. THIS PAGE unclassified			

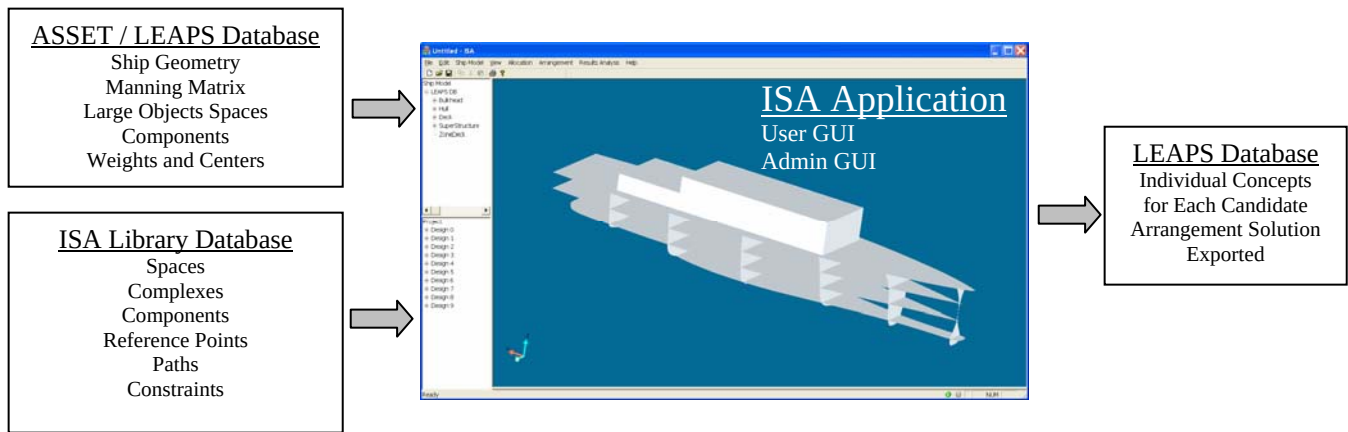


Figure 1: ISA Overall Schematic

As mentioned, ISA takes inputs from two separate databases and generates arrangements for the designer. The process in which this happens is a three step process. The first step is the allocation of the spaces in the ship template to the various zone decks of the ship. Recall that a zone deck is one major structural region of the ship, such as a region surrounded by one deck and one subdivision. The allocation of the spaces is handled by a Hybrid Genetic Algorithm – Multi Agent System (HGA-MAS). Once an optimal allocation is achieved the algorithm then enters a two step arrangement phase where the spaces within the individual zone decks are topologically and then geometrically arranged. This arrangement portion of the process is handled by a genetic algorithm and stochastic growth algorithms. For detailed information on ISAs design and its algorithms refer to Parsons (2008).

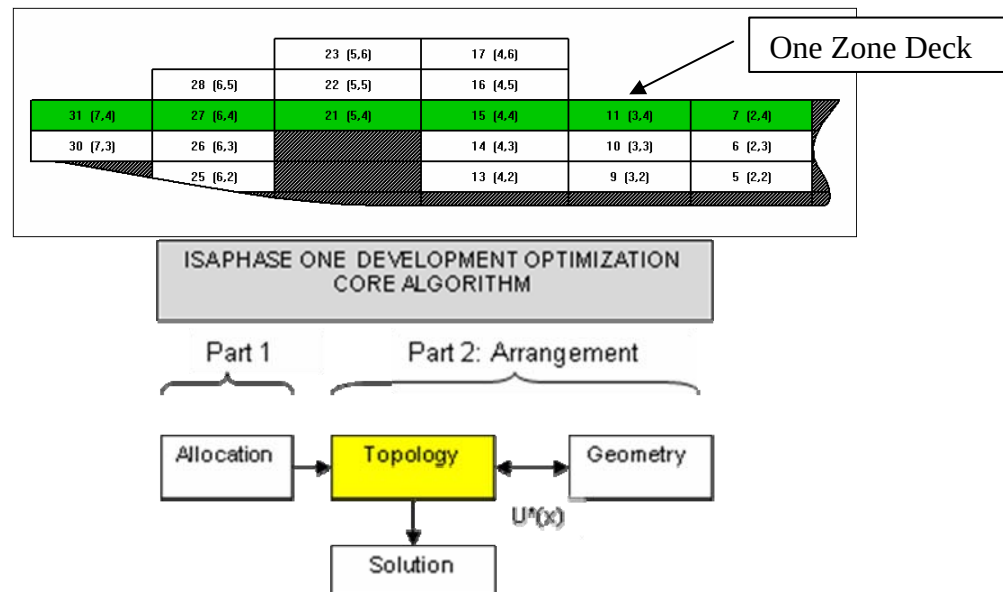


Figure 2: ISA Algorithm Review and Zone Deck Definition

In Daniels et. al. (IMDC 2009), the ISA platform was given modifications and upgrades to the manner in which the compartment and access networks were handled in the optimization algorithms. The methodology that was introduced was termed a Passage Variable Lattice Network (PVLN) and allowed the application to represent more complicated passage configurations above and beyond the standard H and parallel passage, shown in Figure 4, configurations available in phase one of development. The PVLN creates a much more adaptive and robust compartment and access model (Figure 2). This paper will discuss further modifications to the governing objective functions. Furthermore, this paper will also discuss some of the studies and results that have come out of the experimentation with the demonstration code outlined in the IMDC 2009 paper.

ISA PASSAGE VARIABLE LATTICE NETWORK REVIEW

The following section is an excerpt from Daniels 2009, whose purpose is to provide continuity and review. A new passage formulation was introduced for the development of more realistic compartment and access networks in a ship. This was needed because most ships often have passage configurations that do not follow the H and II configurations that were used in the previous round of ISA development. Therefore a more generic method of passage network generation had to be devised. After studying General Arrangements drawings from sample ships, it was determined that a Passage Variable Lattice Network (PVLN) was a good candidate for representing most of the various types of passage networks that are seen on ships. Most passage networks on a ship follow a city grid style lattice of passage of intersecting longitudinal and transverse (athwartships) passages as seen in Figure 3. The city grid street pattern from civil engineering was the inspiration behind this methodology. It should be mentioned that these interlacing passages are not required to be orthogonal in layout. In addition, each zone-deck's lattice passage members can be optionally fixed in their geometry and also linked to neighboring zone-deck (ZD) lattices.

The result is a collective passage lattice structure that can span every level of the ship and represent a large number of possible passage configurations (Figure 4). It should be noted that these lattice networks do not have to be symmetric, nor do their passage segment members have to span an entire zone-deck (Figure 5). In addition, passage segments in the lattice may have multiple waypoints to represent more complex inter-node geometries. The individual passages will also have geometry configuration controls including:

- Minimum and maximum segment transition angles
- Minimum and maximum segment lengths
- Orthogonality restrictions
- Limit controls on number of waypoints allowed

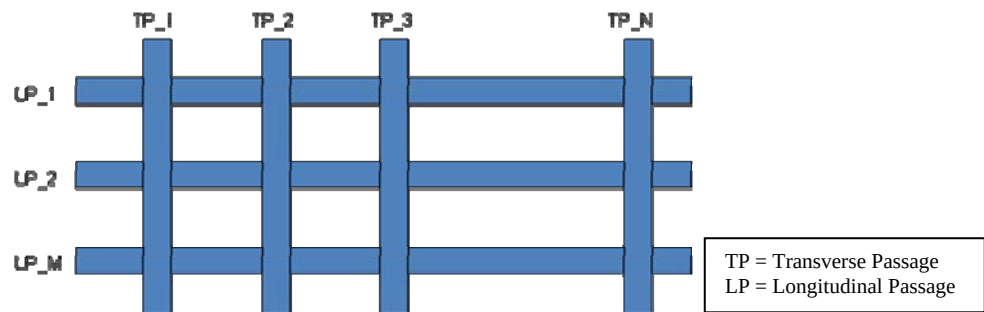


Figure 3: Passage Variable Lattice Network (PVLN)

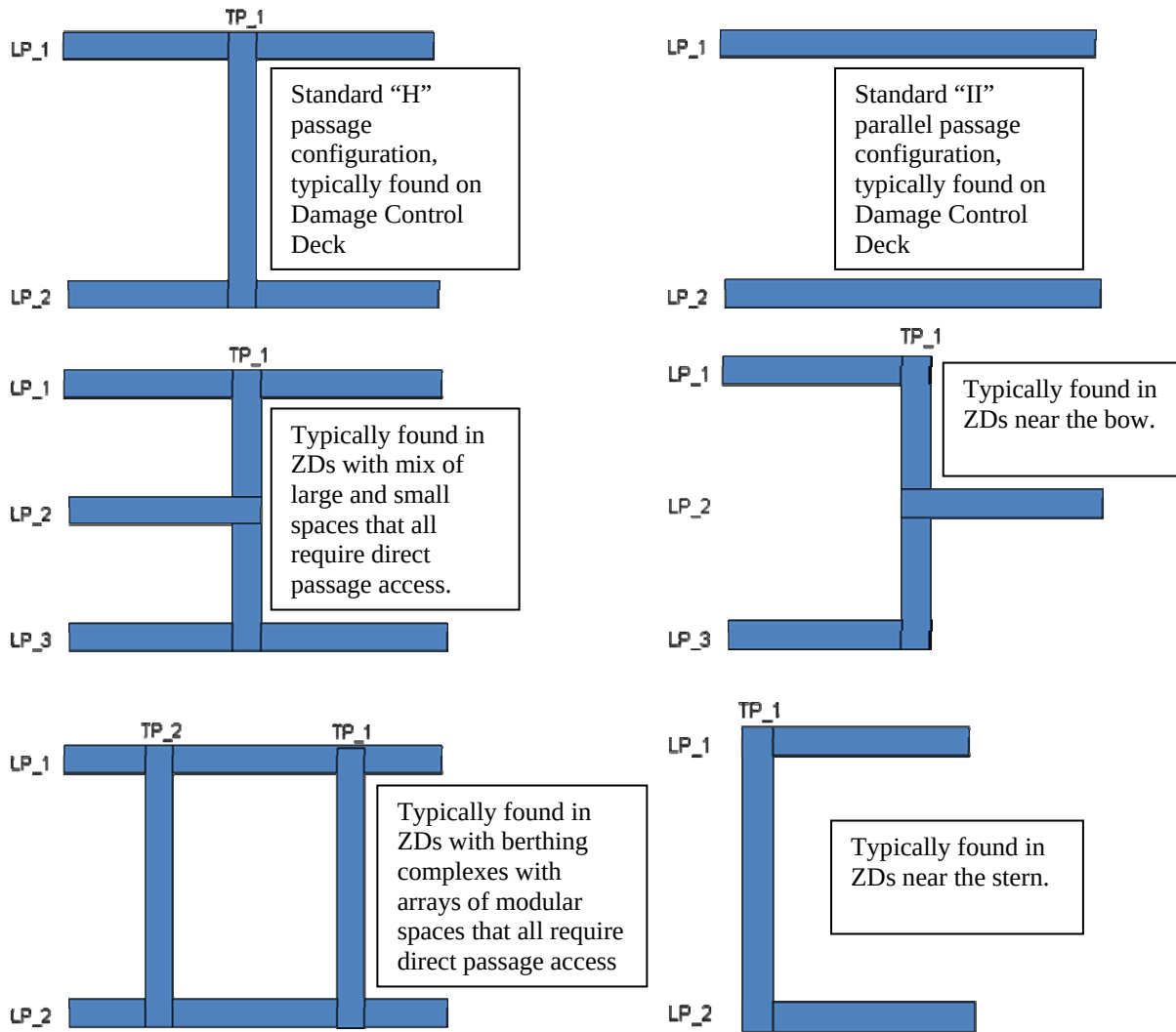


Figure 4: Some Candidate Passage Configurations Illustrating the Flexibility of the PVLN method

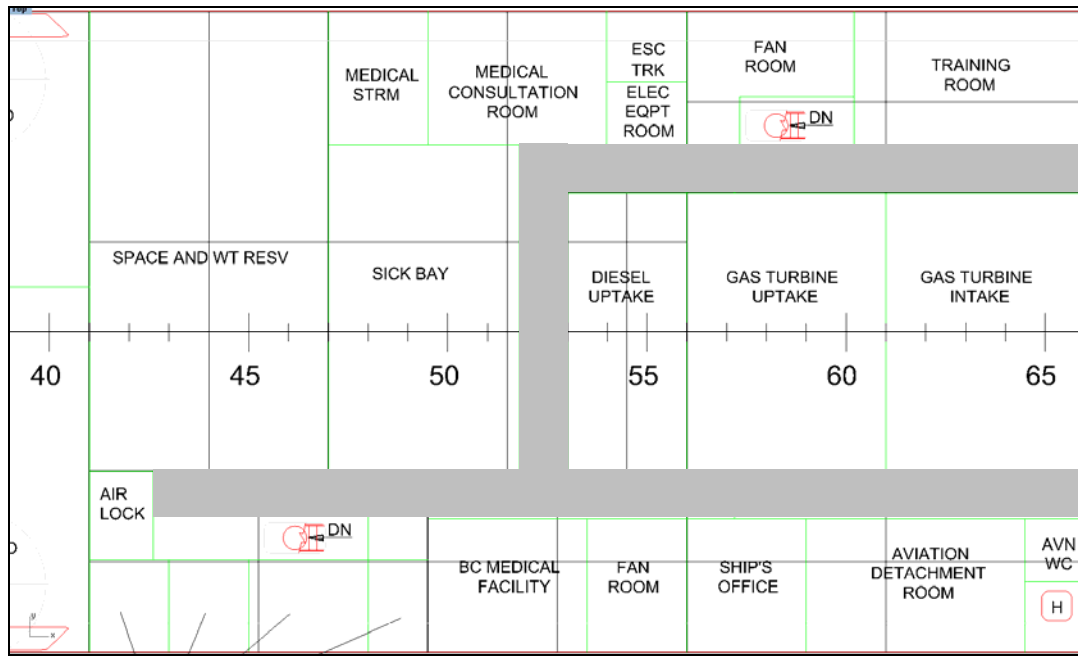


Figure 5: Example of Non-symmetric Lattice from Notional Corvette

The variable aspect of the PVLN refers to the passage's ability to vary in both geometry as well as existence. Each zone-deck of the ship has a passage lattice of M longitudinal passages by N transverse passages as an upper limit on the lattice size. These parameters are settable, however practical use limitations suggest that grid sizes will most likely be less than ten by ten. Whether or not a passage is used in the arrangement is a variable Boolean flag that is manipulated via the optimization algorithm. In a given round of arrangement generation, the passage and access network are generated based on which passages are active at that moment. From this starting point, a stochastic growth algorithm is applied to each space until a stable arrangement has been achieved, as seen in Figure 6. This process is repeated generating multiple geometries per allocation before cycling back to the allocation level of detail for the next generation.

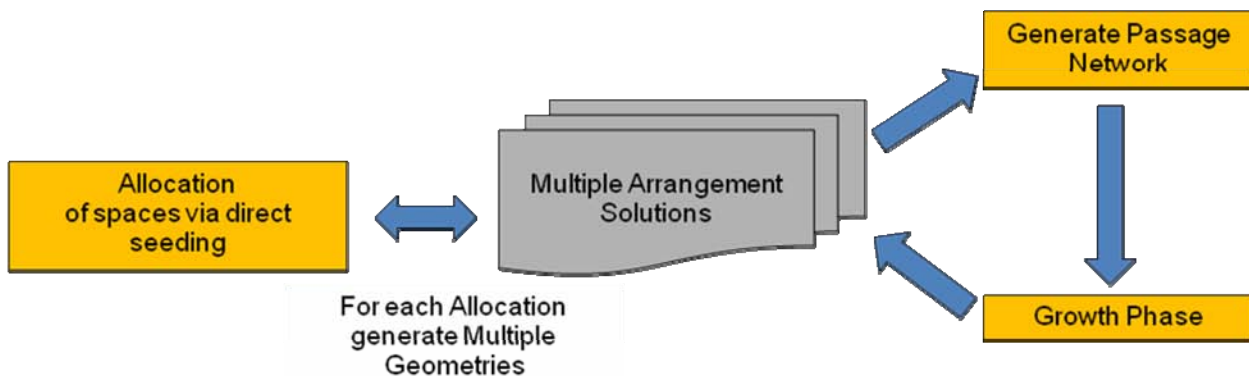


Figure 6: Passage Network Generation and Stochastic Growth Phase Schematic

The PVLN was chosen over heuristic and rule based methods for passage generation because it would require substantially fewer lines of code to implement. Programming all of the rules necessary for good passage generation would be prohibitive and complicated from a code maintenance standpoint. In addition, the PVLN method allows for the objective function to determine the “goodness” of a particular passage network configuration. The ability to change and update the underlying constraints in the system makes the PVLN expandable and updateable over time.

ISA DEMONSTRATION CODE REVIEW

Following the end of Phase One development, the ISA development team conducted an analysis in which the team took a critical look at ISA, including where the platform currently stood, its strengths and weaknesses, areas to improve, and which future development paths the team wanted to pursue. The review covered the following major areas:

- General Programming Practices
- GUI Design and Development
- 3D Modeling and Geometry
- Application Architecture
- Algorithm Development and Design
- Allocation Reformulation
- Arrangement Reformulation
- Multi-Agent Systems Intelligence
- Performance and Potential for High Performance Computing
- Database Design
- LEAPS Integration

In order to provide a proof of concept for the PVLN, as well as work on improvements to the overall ISA application, it was decided that a demonstration add-on code was to be made and integrated with the existing ISA platform. The application was meant to address the following major improvements areas with brief summaries:

- Algorithm Development and Design
 - *Shift from three step to two step optimization algorithm:* It was determined that if the problem was reformulated, the algorithm could be reduced to an integrated allocation and arrangement cycle. The topological optimization portion of the arrangement step would be unnecessary. There are numerous advantages to this formulation change outlined in Daniels (2009).
 - *HGA-MAS core based solver system:* It was decided that the success of the HGA-MAS in the phase one development of ISA was to be improved and expanded on in the phase two development. Thus, the responsibility of both allocation and arrangement was assigned to the agent based system.
- Allocation Reformulation
 - *Direct seeding via X,Y,Z coordinates instead of zone-deck allocation:* As mentioned in the previous bullet the shift to a two step optimization where allocation and arrangement would happen simultaneously, necessitated a change in the allocation strategy. In phase one the allocation was achieved using the index numbers of the zone decks. Because arrangement is an immediate concern, it was decided to use the direct seeding x,y,z coordinates to designate the allocation of a space and which zone deck it resides in.
 - *Objective function reformulation:* Because of the changes to the allocation step of the algorithm, the objective function had to be modified. It was also determined that the influence of the zone deck utilities was too great on the algorithm versus the space topological constraints (see Daniels 2009).
- Arrangement Reformulation
 - *Objective function reformulation:* Similarly to the allocation objective function, the inclusion of the arrangement development step earlier in the algorithm coupled with the addition of a new compartment and access network discussed in the previous section, the arrangement objective function also needed modification (see Daniels 2009).
 - *Passage Variable Lattice Network:* See previous section for full explanation.
 - *Space – Space direct geometry manipulation:* In phase one development of ISA the spaces generated their geometries on a matrix, or discretized grid. When a space moved into a region it occupied the cells of that grid. The problem encountered with this methodology was that, if there were errors in the control logic, spaces could move out of a region and not free up the cells they previously occupied. This led to erroneous geometry and incorrect growth moves by spaces trying to occupy cells. With phase two development, it was decided to have the spaces directly query one another for geometry information when making a move. It is a more computationally expensive, but more robust methodology.
 - *Real clipped space areas (with respects to hull form):* In phase one development of ISA, the arrangement step did not take into account the real ship geometry when growing spaces. The zone decks were approximated as Axis Aligned Bounding Boxes. For phase two development, the real ship geometries for deck plans will be used to determine actual space geometries and areas.

In order to save on programming infrastructure, the demonstration add-on code was built on top of the ISA main application. An additional menu was created to control the new optimization manager included within the demonstration code. A screenshot of the application with the demo code GUIs active is shown in Figure 7. The GUI is a tabbed interface with a tab for each of the major model components (zone-decks, spaces, passages, and stair towers), as well as tabs for the optimization control, results viewing, and debugging window. The ISA demonstration add-on code required approximately 60,000+ lines of code in native C++ with Microsoft Foundation Class GUIs.

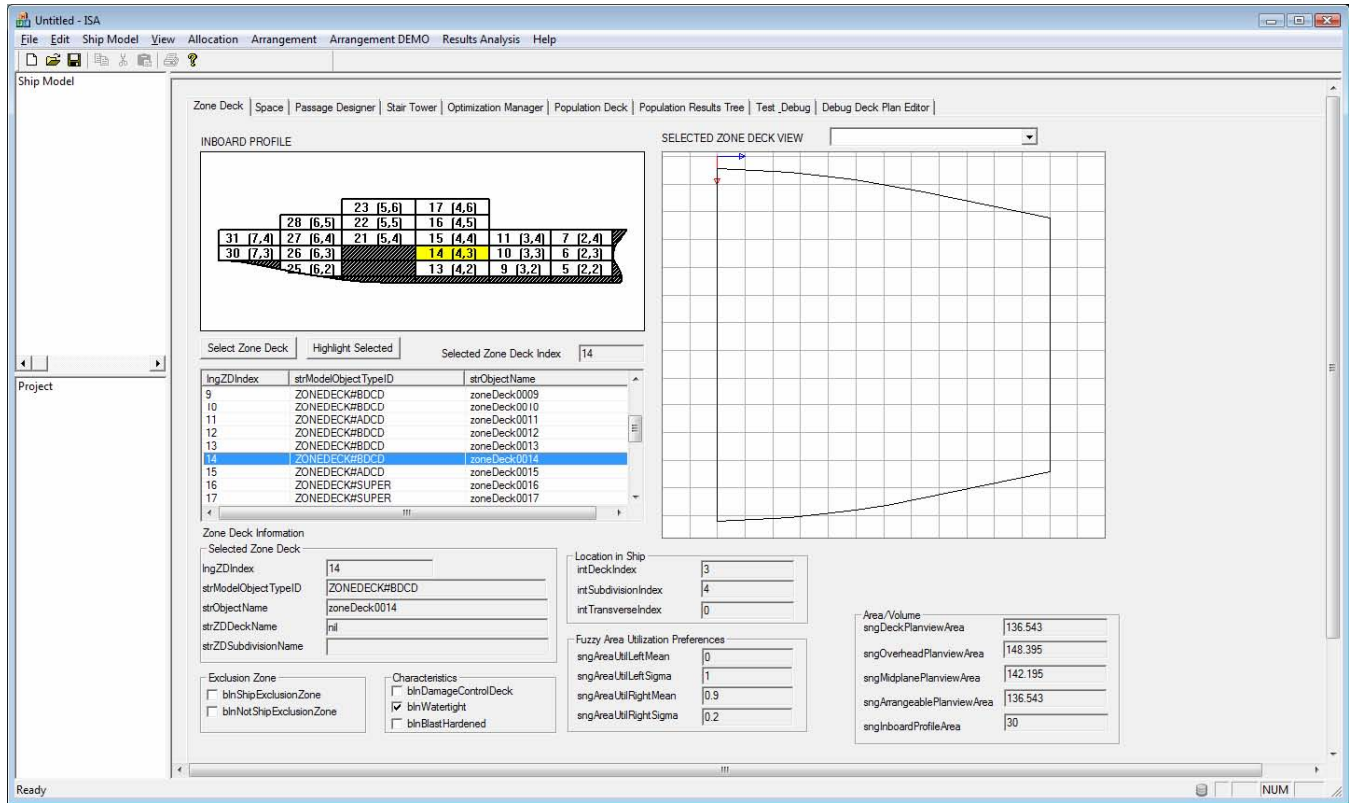


Figure 7: ISA Add-On Demonstration Code Graphical User Interface

ISA PHASE TWO OBJECTIVE FUNCTION MODIFICATION

As mentioned above, one of the primary areas in the reformulation of the optimization approach was the addition of the PVLN to model more complex passage networks than were previously capable of being handled by ISA. In order to accommodate the design considerations of the new passage network formulation, two additional terms were added to the objective function for the arrangement formulation. These additions can be seen in Equation 1, below. The first term is an Nth Lowest Percentile set of the minimum passage network utilities. The passage network utilities are aggregated by zone-deck. For an individual zone-deck the minimum passage constraint utility is the minimum utility of the various passages in the zone-deck. The fuzzy utility constraints concentrate on feasibility of the passage network and applicable regulations. An example of the type of constraint a passage would have is the maximum dead end hallway constraint, where dead end passages are not allowed to be longer than X feet / meters without having a second means of egress. The second term is an Nth Lowest Percentile set of the minimum Compartment and Access Network constraints. These are a set of the constraints that determine compartment and access feasibility of the arrangement. For example, if the space is currently allocated to a zone-deck below the damage control deck, it will require two means of egress for satisfactory access. The compartment and access feasibility is being handled by a new Compartment Network Agent, a service agent providing information to the other domain and design agents in the system. It does not actively make design change requests to the domain agent. The Compartment Network Agent provides information on compartment adjacencies, feasible egress routes, distance traveled between compartments using actual passage network paths, etc.

$$\begin{aligned}\max(U(x)) &= \max(U_{NthLowPassage} \times U_{NthLowCompAccess} \times U_{WeightedAvgSpaceMin}) \leq 1.0 \\ U_{NthLowPassage} &= Nth_{Low}(U_{P1}, U_{P2}, \dots, U_{PK}) \leq 1.0 \\ U_{NthLowCompAccess} &= Nth(U_{S_ACC1}, U_{S_ACC2}, \dots, U_{S_ACCN}) \leq 1.0 \\ U_{WeightedAvgSpaceMin} &= \sum_{i=1}^I \frac{w_i \times \min((U_{RAS}, U_{AR}, U_{MOD}, U_{MSD}, U_{PER}, U_{ACC}), (U_{i1}, U_{i2}, \dots, U_{iJ}))}{I} \leq 1.0\end{aligned}$$

where:

RAS – Rational Area Satisfaction (Current Area/ Required Area)
AR – Aspect Ratio
MOD – Minimum Overall Dimension
MSD – Minimum Segment Dimension
PER – Perimeter
ACC – Access Requirements

also where:

P – Passage network on Zone-deck k
K – The number of Zone-decks
N – The number of Spaces
I – Total Fuzzy Importance of the Spaces
J – The number of constraints for an individual object
w_i – The relative importance weighting of Space i

Equation 1: Phase Two Arrangement Objective Function presented in IMDC (2009)

Early on in the development of the demonstration code the decision was made to make two slight modifications to the objective function. The passage utility term was split out into two sets; the minimum passage set and minimum stair tower set. This was primarily done because it was easier to code the utility gathering for them in discrete chunks instead of pooling the terms into one set. The second modification was that it was deemed that spaces required area utilities were important enough to be promoted to the main optimization objective function, instead of just being one of the spaces constraints. This was done to more directly provide a replacement for the zone-deck area utilization sets from Phase One development. Equation 2 shows the modified objective function incorporating these changes.

$$\max(U(x)) = \max(U_{MinPassage} \times U_{MinStairTower} \times U_{NthLowCompAccess} \times U_{MinSpaceRAS} \times U_{WeightedAvgSpaceMin}) \leq 1.0$$

Equation 2: Modified Objective Function for Arrangement

ISA DEMONSTRATION CODE RESULTS

The zone-deck chosen for running preliminary tests for the ISA demonstration code was zone-deck 14, located below the damage control deck (see Figure 8). As can be seen by the deck plan of the zone-deck, it has moderate curvature which will task the real area calculations of the spaces. This was an improvement to the system over the previous version, which used axis aligned box calculations in space area calculations. The actual areas involving trimming the boundaries by the hull form were not previously used. The spaces that are assigned to this zone-deck from previous results are two Petty Officer Berths, the Ship Department Supply Room, and three Specialist / Enlisted Berths.

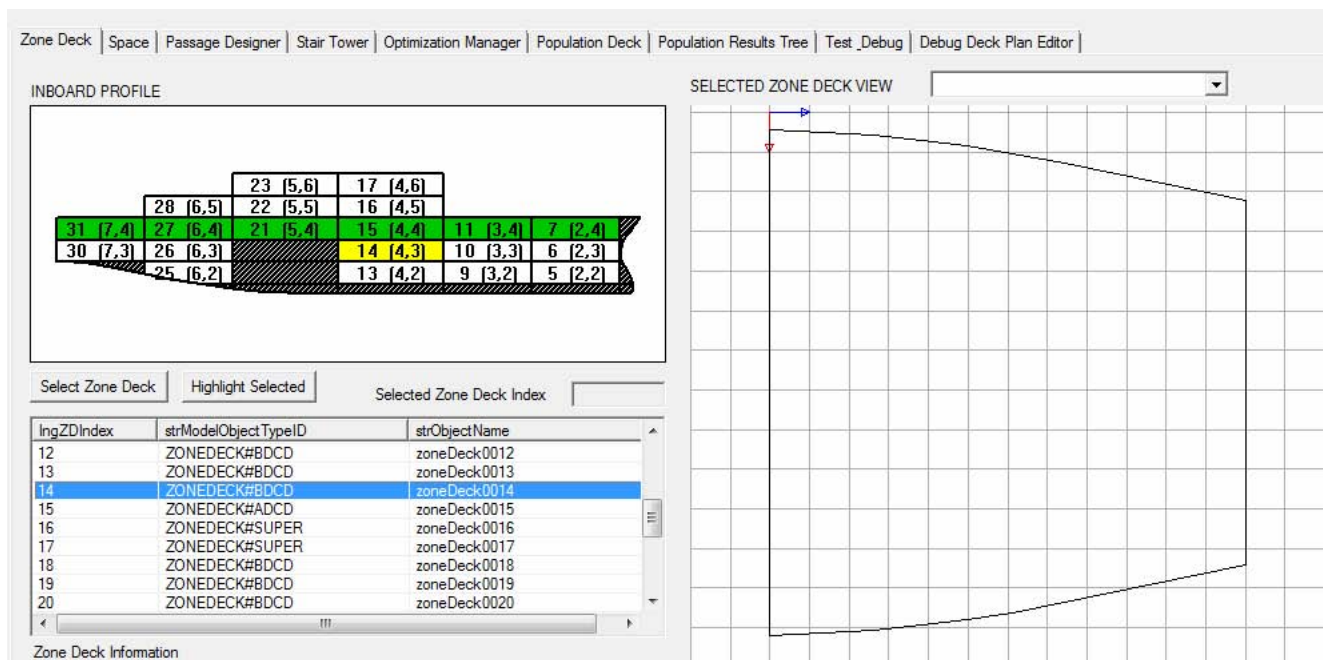


Figure 8: Zone-deck 14 Inboard Profile and Deck Planview

One of the new additions to the demonstration code is a powerful debugging and editing GUI (Figure 9). This tab on the main interface allows the user to query the active population, as well as the top N candidate solution history. The user can also select the individual compartments within those solutions and retrieve information about their current state. In addition, the user can interrogate all of the individual constraints of the compartments. One key feature is the ability for the user to view and edit the compartment geometries individually to either make corrections or to “draw” in a solution. This is useful in conducting “what if” scenario testing and benchmarking.

The optimization kernel was run at both 1 x 1 (meaning one longitudinal passage and one transverse passage) and 2 x 2 PVLNs. The size of the experimental grid will be noted as a part of discussion. A few of the runs with their output arrangements will be discussed. A manually entered arrangement will also be addressed as well to illustrate the “what if scenario” use case.

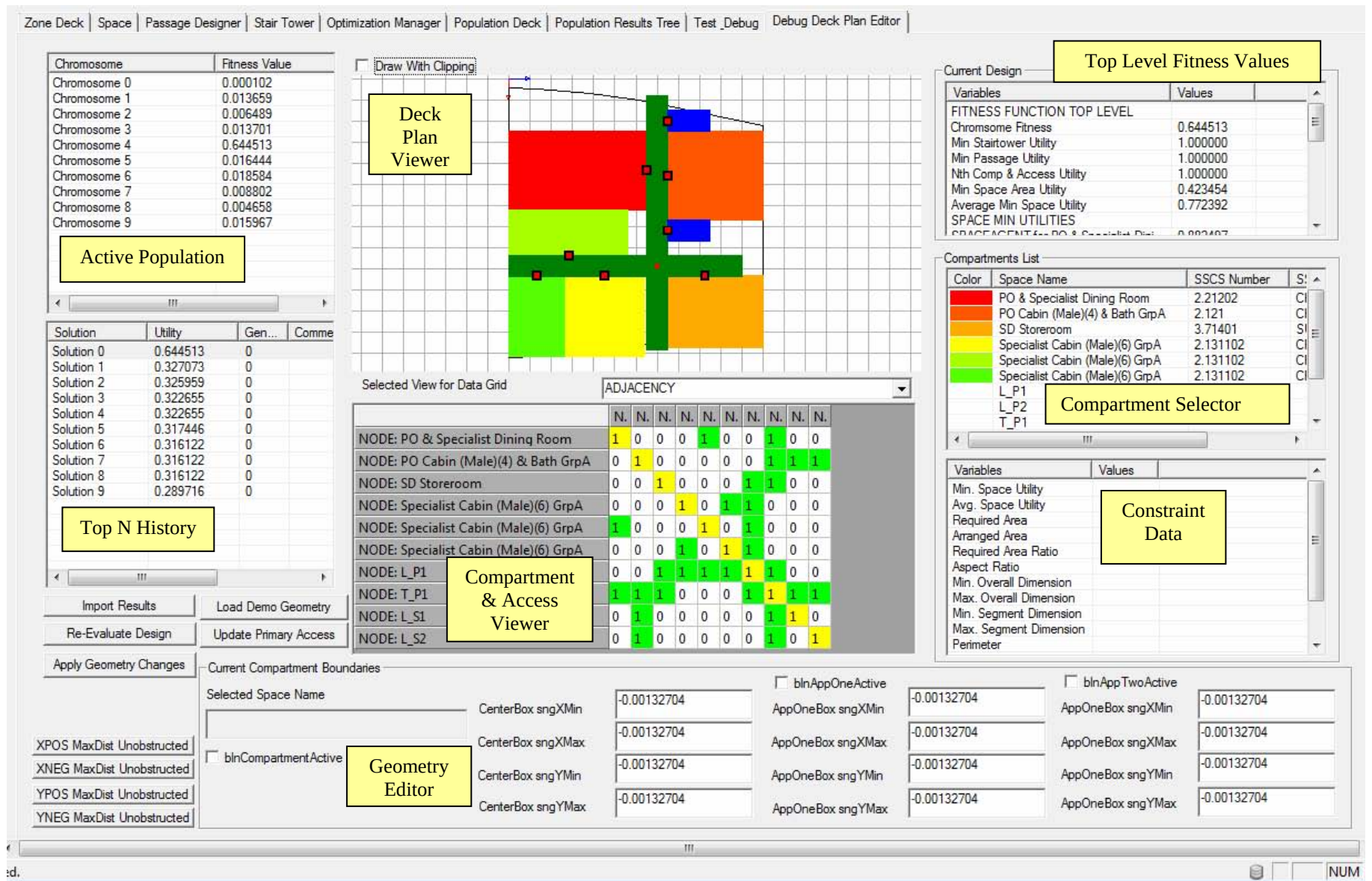


Figure 9: ISA Demo Results Viewer / Editor

Results One

The first run shown had a 2 x 2 PVLN and yielded a best arrangement of 0.798 at 24 generations out of 100 generations run (Figure 10). The configuration shown was an “L” shaped passage configuration. As can be seen in Figure 11, the spaces have good aspect ratios and the required area satisfaction is reasonably satisfied at a 0.794, however it is still the driving factor in the objective function roll-up. The two Petty Officer berths have clustered together at the end of a passage to make an “officer’s country” and the Specialist Cabins have clustered together as well. This is a good illustration of the relative adjacency constraints in a problem. The stair towers, while having adequate access to passages, are too close to one another. This is a limitation of the demonstration code as a relative separation constraint for the stair towers has not yet been implemented due to scope limitations of the project. They are intended to be added at a later date.

Figure 11 has both the deck plan of the best candidate solution, the compartment and access network adjacency matrix, and the Utility rollups for the fitness function shown. For the compartment and access network, the red dots outlined in black are either space / passage or stair tower / passage accesses. Red dots with no black outline are passage / passage connection points. It can be seen in the debugging adjacency matrix that symmetry of node connections is preserved in the model.

Also, from a use case standpoint, the user can make modifications to the existing candidate solution and re-evaluate the solution manually to see what effects the modifications had. For example, in this case the user can extend the forward most bulkhead of the SD Storeroom to take up the void space in front of it. User editing and intervention is part of the ISA use case and operational philosophy.

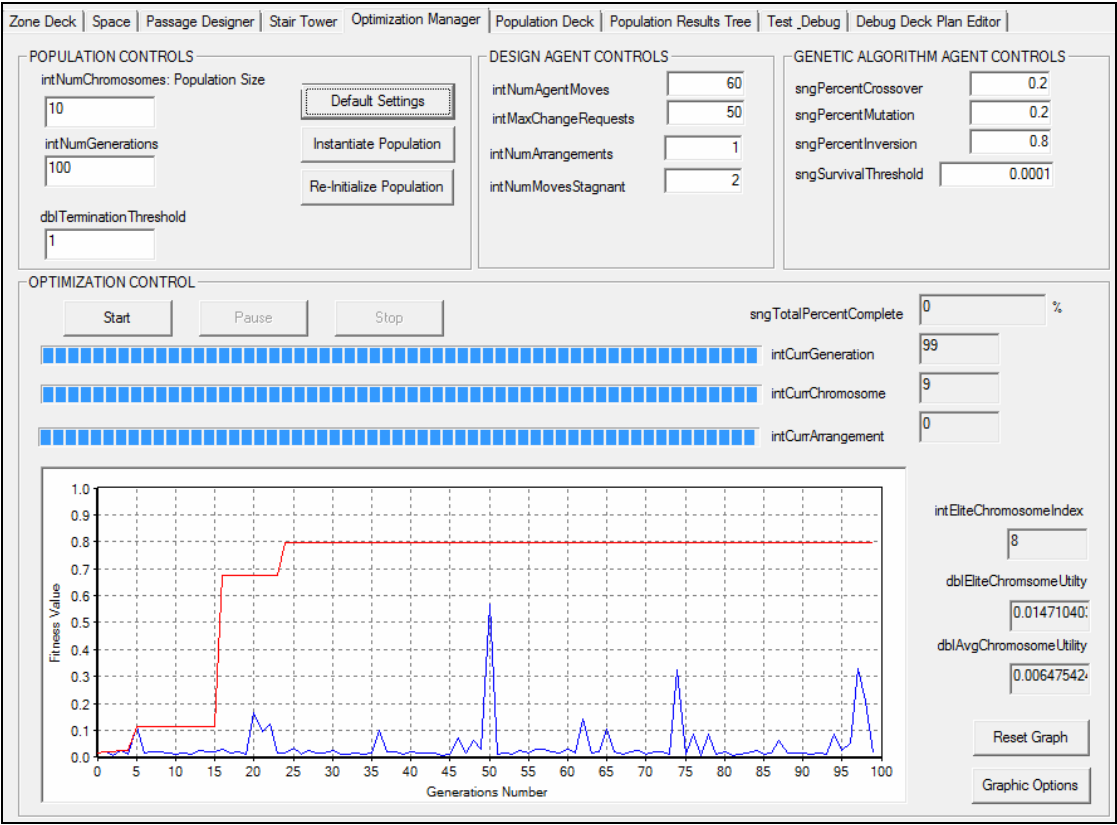


Figure 10: Results One Elite Time History

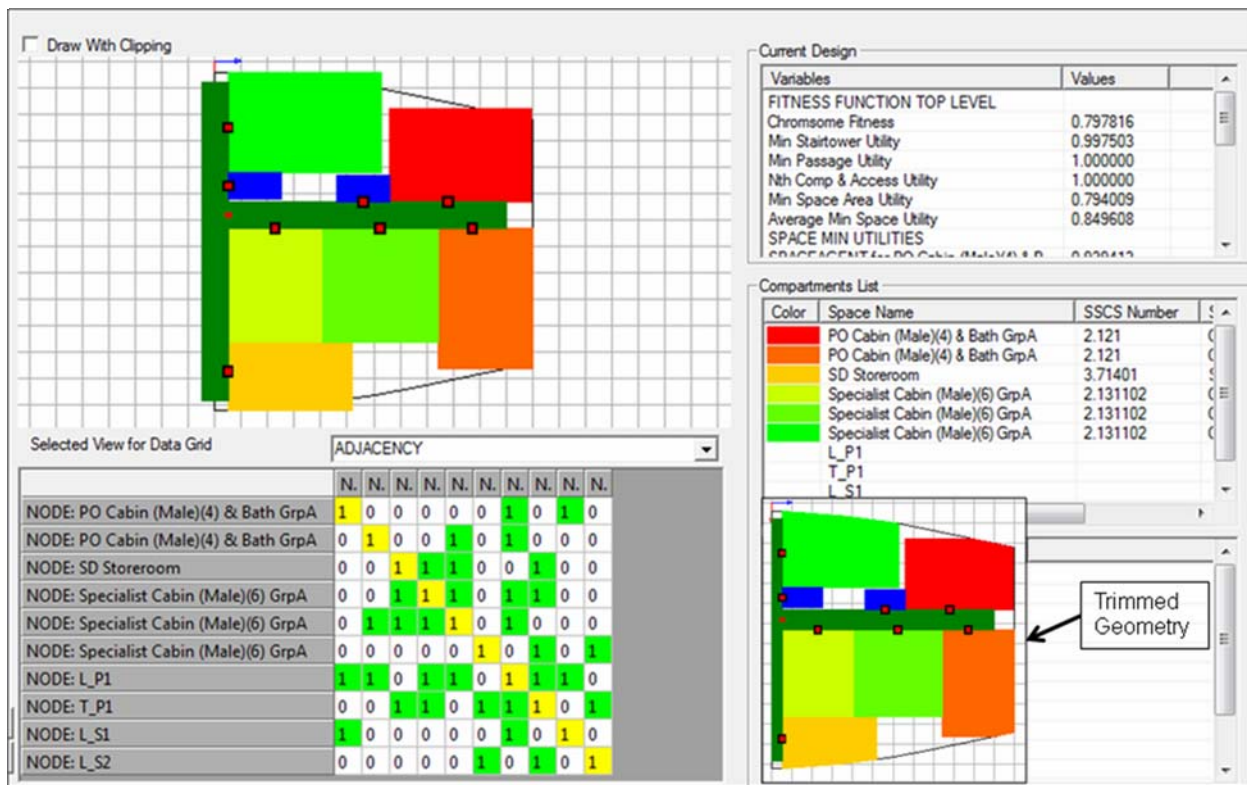


Figure 11: Results One Deck Plan and Utility Values

Results Two

The second run shown had a 1 x 1 PVLN and yielded a best arrangement of 0.787 at 34 generations out of 100 generations run (Figure 12). The configuration shown was an “I” shaped passage configuration with a single transverse passage (Figure 13). Note that the stair towers have good separation and are on opposite sides of the zone-deck. Because of a program limitation, the spaces on the aft side were unable to push the passage forward to improve their area utilities. Note that the organization of the spaces exhibits the same clustering of berthing spaces. However, due to a similar program limitation, the stair towers are not allowed to be pushed. This causes the area utility to suffer and result in a fitness value slightly lower than Result One.

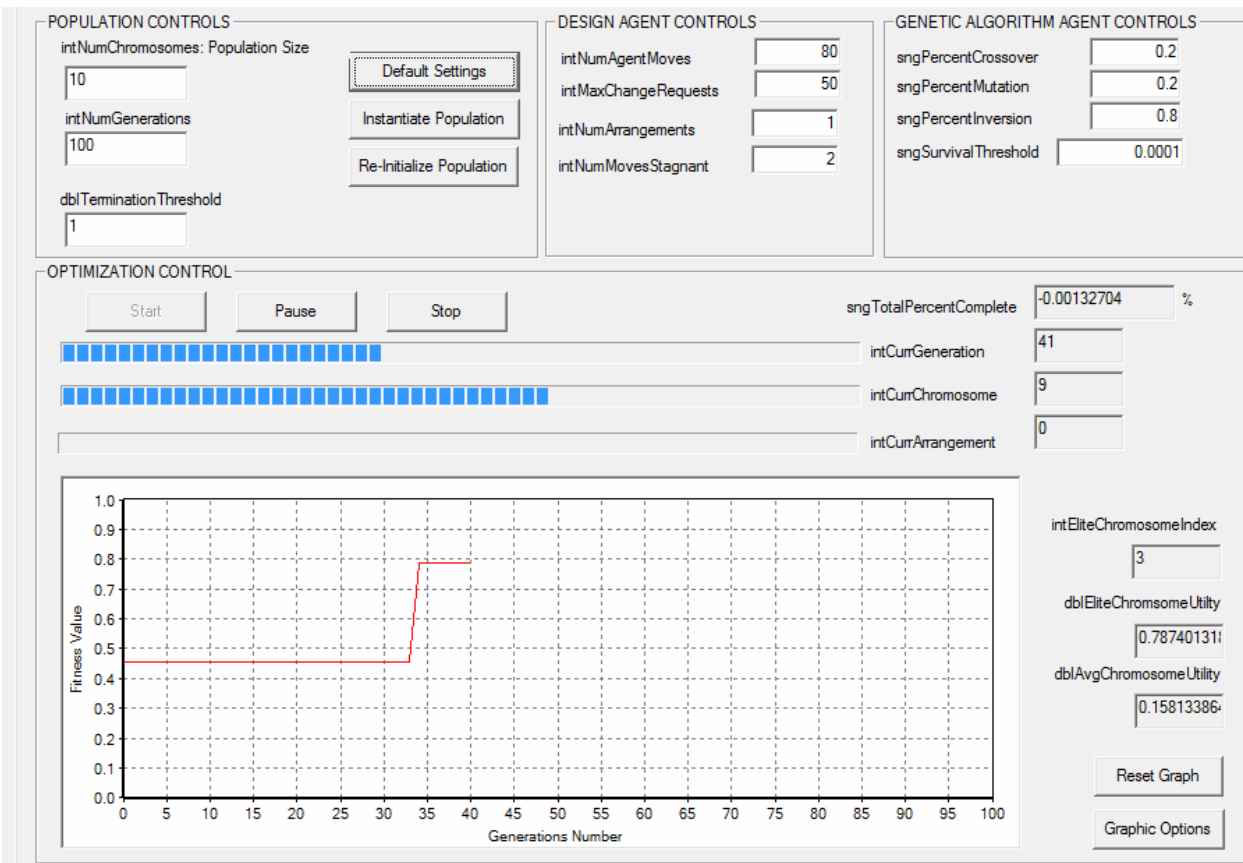


Figure 12: Results Two Elite Time History

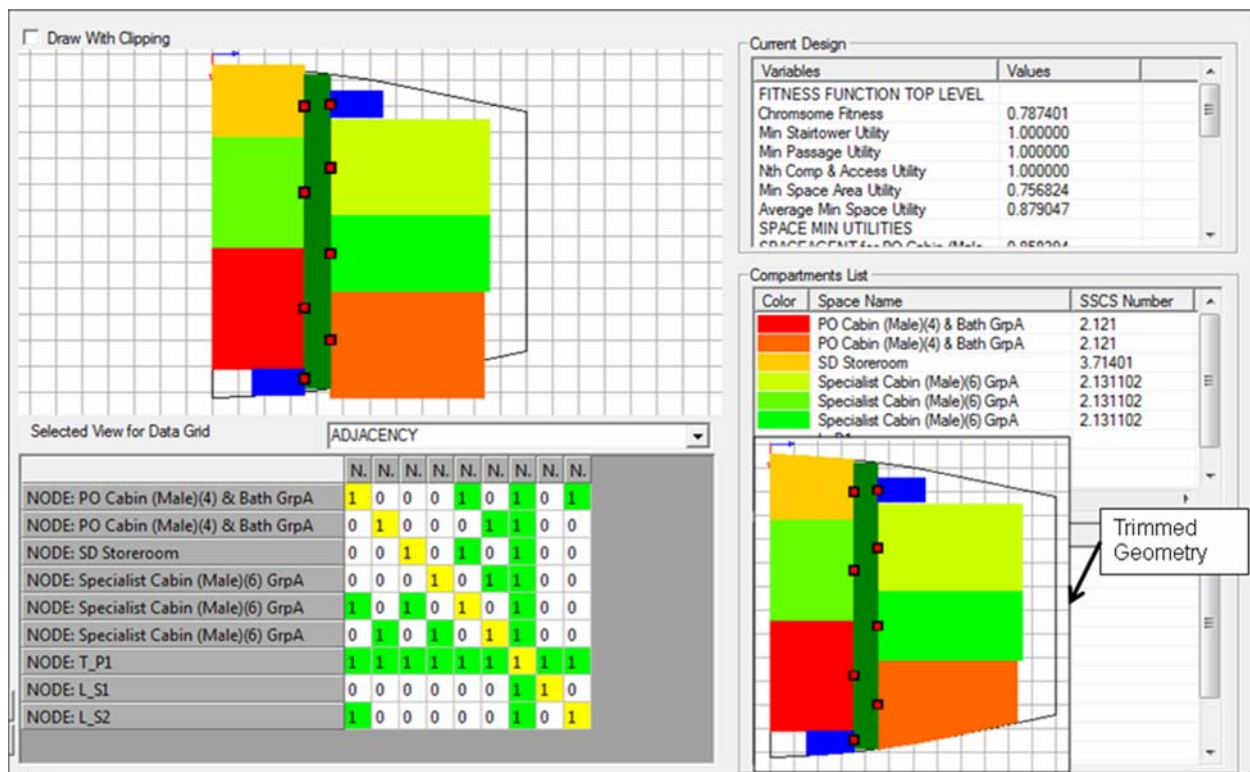


Figure 13: Results Two Deck Plan and Utility Values

Manually Entered Arrangement

Taking cues from preliminary results in the development of the software, the authors manually drew the arrangement shown in Figure 14 as an objective function validation step. This arrangement is similar to that shown in Result Two. It comes from a 1 x 1 PVLN and is a single athwartships passage that connects to the two stair towers symmetrically about the centerline. Again there is clustering in the officer and specialist cabins respectively. In addition it makes use of the spaces ability to grow appendages around the stair towers to take advantage of otherwise unused space. Note that the use of the appendages and the balancing of space fore and aft yielded a space minimum area utility of 0.993 satisfied. It should be noted, that for the optimization runs, the appendage growth was temporarily disabled to concentrate on the compartment and access algorithms. Currently the optimization code does not have the capability of appendage generation turned on. This illustrates the importance of the appendage growth feature. Appendage growth will be included in the full scale application development.

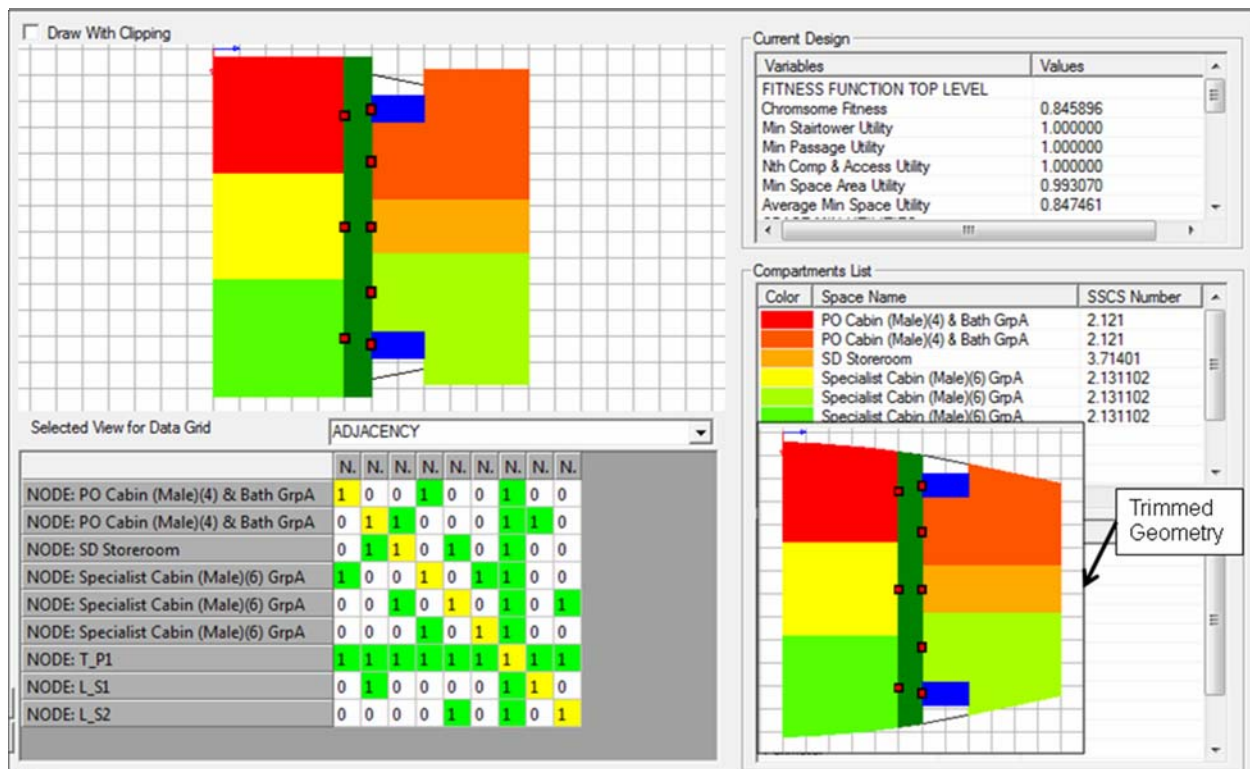


Figure 14: Manually Drawn in Arrangement Deck Plan and Utility Values

CONCLUSIONS AND FUTURE WORK

At this point in the research, enough features of the IMDC demonstration code are active for the software to produce realistic arrangements of spaces. Although it produced good results, there are some limitations of the current code. Space agent move logic needs improvements and the ability to push any type of compartment (space, passage, or stair tower) is necessary in future versions. Figure 15 illustrates the need to be able to push a passage and have it in turn push a stair tower, which is currently not allowed in the program. Two sample arrangement results were discussed and one manually entered arrangement was analyzed. The optimization core arrangements and the manually entered arrangement share a number of features in layout. Clustering of like berthing spaces exhibits that the relative adjacency constraints present are working properly.

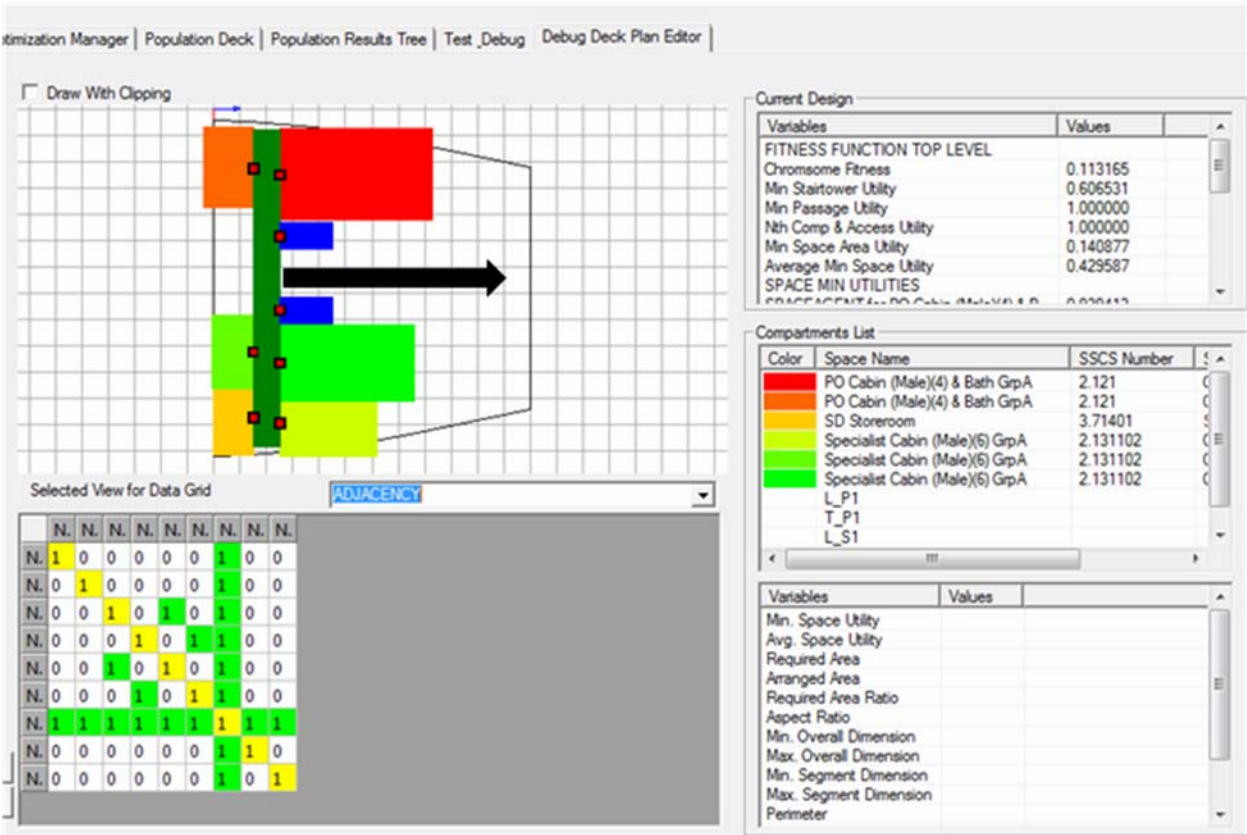


Figure 15: Illustration of need for pushing stair towers

Future research with the code will concentrate on three primary areas. First, the development team wants to directly support the comparison and what if scenarios that designers will most likely need to perform. Most of this development capability will be done via reprogramming better user interfaces with CAD-like functionality.

Second, speed needs to be improved dramatically. This can be done through algorithm improvements, parallelization, and optimization of the code. The focus on parallelism will investigate both standard parallel programming and hardware acceleration based programming. This is particularly important because the shift in logic that spaces interact directly with each other means the move subroutines carry an expensive processing time. $O(n^2)$ in the worst case, where n is the number of neighboring spaces in the zone deck of interest. Whenever spaces make a move they have to query the geometries of the neighboring spaces. Effort will be required to reduce the processing requirements of these subroutines.

Third, the constraint system in ISA needs an overhaul. The authors intend to implement a system that is better than the current version which relies on a centralized object query system. This system will be expandable and more flexible than the current framework. ISA continues its development of capabilities and usability in order to make the general arrangements optimization of a ship a more quantifiable and rational process.

ACKNOWLEDGEMENTS

The ISA development team would like to thank NSWC-Carderock and the Office of Naval Research for sponsoring this ongoing research and development effort. In particular the team would like to thank Bob Ames, Richard Van Eseltine, Keawe Van Eseltine and the rest of the LEAPS development group at NSWCCD. We would also like to thank Kelly Cooper of ONR for her ongoing support of the project. In addition the team would like to thank Mike Parsons, Eleanor Nick Kirtley, Su Liu, and Hyun Chung for their contributions on the wrapping up of Phase One development as well as their ongoing support contributions to this project. Their advice and consultation is much appreciated.

REFERENCES

- ASSET, "Advanced Surface & Submarine Evaluation Tool," Naval Surface Warfare Center, Carderock Division, Version 6.0, 2007.
- Daniels, A. and M. G. Parsons, "An Agent-Based Approach to Space Allocation in General Arrangements," *Proceedings of the 9th IMDC*, Ann Arbor, MI, May, 2006.
- Daniels, A. and M. G. Parsons, "Development of a Hybrid Agent-Genetic Algorithm Approach to General Arrangements," *Proceedings of COMPIT'2007*, Cortona, IT, April, 2007; to appear in *Ship Technology Research*, 2008.
- Daniels, A. and David Singer and Farah Tahmasbi, "Intelligent Ship Arrangement (ISA) Methodology Improvements and Capability Enhancements", *Proceedings of IMDC 2009*, Trondheim, Norway, May 2009.
- Goldberg, D. E., *Genetic Algorithms for Search, Optimization, and Machine Learning*, Addison-Wesley, Reading, MA, 1989.
- Kosko, B., *Neural Networks and Fuzzy Systems: A Dynamic Systems Approach to Machine Intelligence*, Prentice Hall, Englewood Cliffs, NJ, 1992.
- LEAPS, "Leading Edge Architecture for Prototyping Systems," Naval Surface Warfare Center, Carderock Division, Version 3.5, 2006.
- Medjdoub, B. and B. Yannou, "Separating Topology and Geometry in Space Planning," *Computer-Aided Design*, **32**, 1999.
- Michalek, J. J., R. Choudhary, and P. Papalambros, "Architectural Layout Design Optimization," *Engineering Optimization*, **34-5**, 2002.
- "NAVSEA Design Practices and Criteria Manual for General Arrangements Design Chapter 070," NAVSEA T9070-AB-PRO-010, Rev. A, 1992.
- Nick, E. and M. G. Parsons, "Fuzzy Optimal Arrangement of Spaces within a Zone-deck Region of a Ship," *Proceedings of PRADS*, Houston, TX, September, 2007.
- Nick, E., M. G. Parsons, and B. Nehrling, "Fuzzy Optimal Allocation of Spacers to Zone-decks in General Arrangements," *Proceedings of the 9th IMDC*, Ann Arbor, MI, May, 2006.
- Nick, E., "Fuzzy Optimal Allocation and Arrangement of Spaces in Naval Ship Design", Doctor of Philosophy Dissertation, University of Michigan, 2008.
- Parsons, M. et. Al., "*Intelligent Ship Arrangements (ISA): A New Approach to General Arrangement*", *Proceedings of ASNE 2008 Conference*, Alexandria, VA. June, 2008.
- Woodruff, D. and Zemel, E. "*Hashing Vectors for Tabu Search*", *Annals of Operations Research*, Volume 41, pp. 123-137. 1993.

Anthony Daniels earned his BSE, MEng, and Naval Architect degrees in NA&ME and MSE in Industrial and Operations Engineering from the University of Michigan. He currently is a Research Scientist with the department of Naval Architecture & Marine Engineering at the University of Michigan. His research areas include preliminary ship design, and large scale optimization

Dr. David J. Singer is an Assistant Research Scientist and Adjunct Assistant Professor in the Naval Architecture and Marine Engineering department at University of Michigan College of Engineering. He earned a BSE, M.Eng, and PhD degrees in NA&ME and a MSE in Industrial and Operations Engineering from the University of Michigan. He conducts research in the areas of design theory, design optimization, and ship production. Dr. Singer won an ONR YIP award in 2007 for his work in Set-Based Design implementation using fuzzy logic.

Farah Tahmasbi is a PhD student in Computer Science and Engineering and a NSF Fellowship Recipient at the University of Michigan. She earned her MSE degree from the University of Michigan, and her BSE degree from the University of California, Santa Barbara, both in Computer Science. Her research areas are databases, software engineering, and computer security.