Methods for Analyzing Early Stage Naval Distributed Systems Designs, Employing Simplex, Multislice, and Multiplex Networks

by

Douglas Tait Rigterink

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Naval Architecture and Marine Engineering)
in The University of Michigan
2014

Doctoral Committee:

- Assistant Professor David J. Singer, Chair
- Assistant Professor Matthew Collette
- Professor Scott Page
- Professor Nickolas Vlahopoulos

We sweat like boat builders and marvel at our work

Smiling at our imperfections

-“Boat Builders,” A Wilhelm Scream

For K.D.V. and R.A.R.

Here's to life.

ACKNOWLEDGEMENTS

I have received support from innumerable people during the course of my Ph.D., and I would like to specifically recognize:

My advisor, David J. Singer, for both supporting and criticizing my work depending on what was necessary, and making sure I always kept the big picture in mind.

My committee members for lending their time and expertise and making the dissertation possible.

My officemates and colleagues in the NAME department for making my time at the University of Michigan both academically and socially stimulating.

The players I coached on Romulus High School Boys' Lacrosse Team for giving me a distraction from research.

My wife, Fang Dong, for her love and support, which never fail to encourage me in this and all of life's endeavors.

And my mother and father for making me the person I am today.

TABLE OF CONTENTS

CHAPTER

LIST OF FIGURES

Figure

LIST OF TABLES

Table

ABSTRACT

Methods for Analyzing Early Stage Naval Distributed Systems Designs,
Employing Simplex, Multislice, and Multiplex Networks

by

Douglas Rigterink

Chair: David J. Singer

Naval ships are some of the most complex systems ever engineered. The process by which they are designed is similarly complex. The complexity and disjointedness of this process leads to the creation of disparate and incomplete ship design information created by different systems of analysis, completed by different design groups, using different tools, at different levels of fidelity. Distributed system design decisions based off this disparate and incomplete information lead to unnecessary complexity when the design is transitioned from the early design stage to the detailed design stage.

This dissertation presents novel network theory-based methods for better understanding and analyzing the implications of early stage distributed system design decisions. This new method introduces network theory concepts and measures such as degree distribution, system interdependence, and community to the field of distributed systems design as metrics for determining system robustness, as well as develops new techniques for representing physical systems as networks. Additionally, a personnel movement modeling and analysis method, derived from the network concept of betweenness centrality, is developed.

This dissertation documents the first use of multislice and multiplex structures in the analysis of physical systems. System design evolutions are analyzed using multislice network structures and the interactions between systems are investigated using multiplex network structures. These two structures are combined into a novel time-dependent multiplex network structure that is developed in this work. This new structure is used to track the evolution of systems interactions.

A new network complexity metric based on the concepts of planarity and network communities is created for this research in a response to lack of methods for studying the planar and near planar networks that often arise in the study of real systems.

The methods presented in this dissertation do not require complex 3D CAD models or simulations. Therefore, they can be used by a single naval architect to gain insight into the implications of design decisions in the early design stages. This will result in improved naval distributed systems designs that are easier to design, maintain, and upgrade.

CHAPTER I

Introduction

Ships, especially naval ships, are some of the most complex systems ever engineered. A single ship design must be able to accomplish a plethora of missions, sustain a crew that can number in the thousands, accept upgrades and refurbishments, and operate for decades. The design process takes several years. During that period every decision, no matter how minute, will affect the final size, performance, and cost of the ship. Nearly half the cost of a ship is spent on the systems installed on-board (*Miroyannis*, 2006), and the designing of these systems is a complex task. Combining them into a functional design poses an even greater challenge.

Navy designs, unlike commercial designs, are created in an effort to understand the effects and impacts of mission requirements on the design (*Hope*, 1981). These exploratory designs are used for making acquisition decisions in the absence of detailed knowledge about the design space (*Government Accounting Office*, 2002) and allow designers to concurrently manage technological, design, and manufacturing risk. Even though the concept of an exploratory design is almost solely used in naval ship design, it is done using methodologies borrowed from the commercial design process.

The problem with following the commercial design paradigm is twofold: Navy ships are substantially more complex than commercial ships (*Keane Jr.*, 2011) and different stages of the naval ship design are conducted by different entities, as can be seen in Figure 1.1. The Navy oversees the early stage, concept designs while the builders have responsibility for the detailed, production designs. The Navy attempts to forecast which technologies and systems will be available and then design them into the ship without the benefit of a physical model of either the ship or the system. Often, the forecasts or models are incorrect, which leads to systems requiring more of a particular resource (physical space, electrical power, cooling air, etc.) than expected.

This leaves the builders to improvise a solution or, more likely, to request a costly redesign. These redesigns frequently consume what little design margin is originally allocated and lead to the extreme growth in budget that plagues naval shipbuilding (*Government Accounting Office*, 2007).

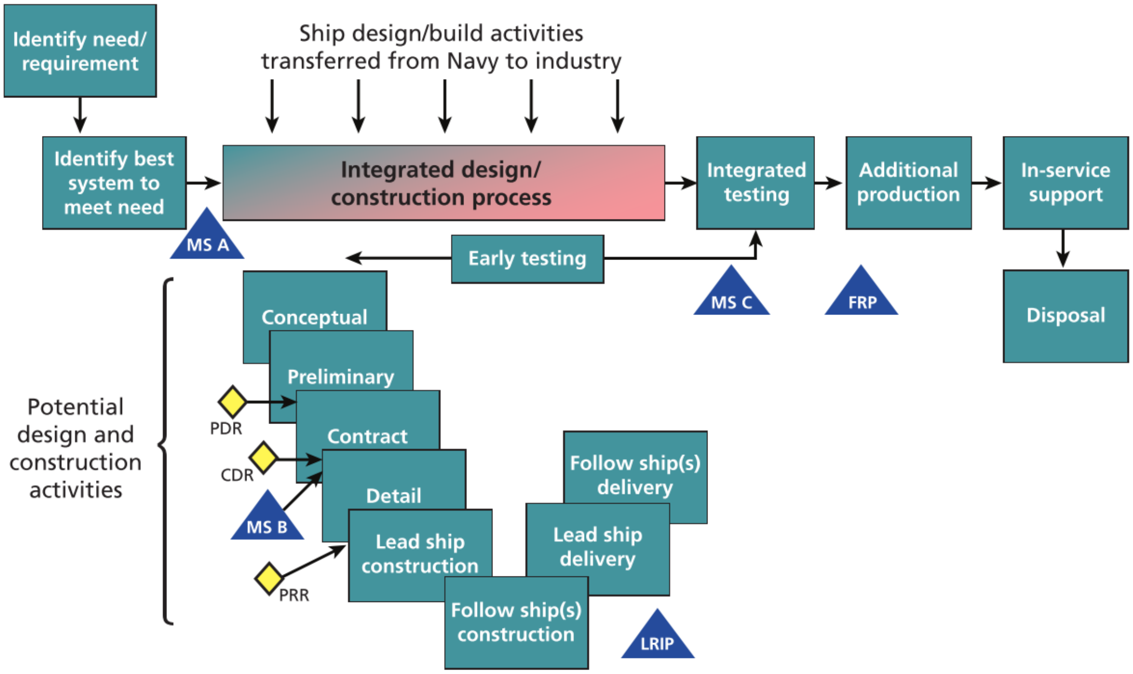


Figure 1.1: The ship design/build process (*Drezner et al.*, 2011).

The issue is not only that new systems can have requirements in excess of predicted levels, but also that meeting these requirements effects the rest of ship and ship systems design, including potential interactions between systems. Currently, these interactions are found via visual inspection of 3D computer-aided design models, but often times visual models are not available in the early design stage. To effectively evaluate conceptual designs during the design space exploration phase, naval ship designers need a new perspective that does not require on creating physical or visual models, but rather one that focuses on capturing and evaluating system structures, interconnections, and interdependencies while quickly and easily evolving with the changing design.

In this dissertation, network theory structures and concepts will be used to represent, analyze, and evaluate ship distributed systems. The goal of this dissertation is to create a new, novel set of concepts, measures, and methods that will allow a single naval architect, or small team of naval architects, to understand the impact of distributed system design decisions, in the early stage, on the final form of the ship without the need for 3D system models.

1.1 Background and motivation

This section aims to further elaborate on the difficulty of the ship design problem, especially in the early stage, and motivate the further discussion of the ship distributed systems design problem.

Design is essentially analyzing tradeoffs between alternatives and choosing the best option given the knowledge at hand. These decisions are necessary to limit the design space to a manageable size. In early stage design, the design space is at its maximum size while the amount of knowledge the designer has about the design space is at its minimum. Yet, as much as 70% of the life-cycle costs of a design are locked in at the completion of this stage (*Calkins et al.*, 2001). Through design space exploration

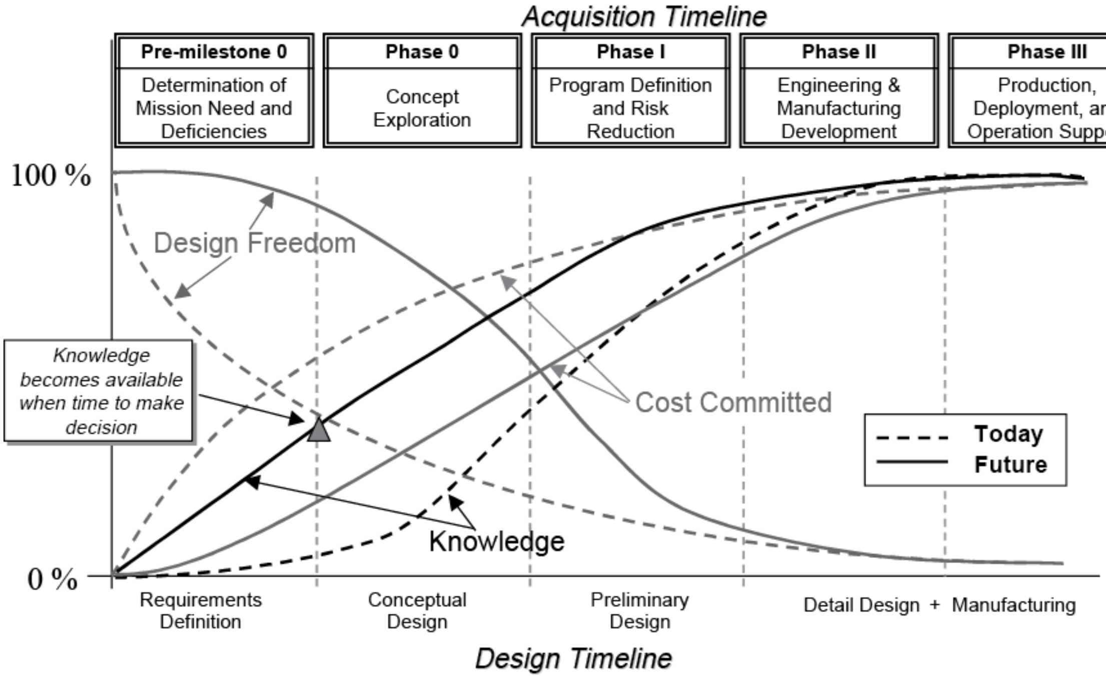


Figure 1.2: A depiction of the early stage design environment (*Marvis and DeLaurentis*, 2000).

and design decisions, the amount of knowledge is increased while shrinking the space. This process is repeated iteratively until one either runs out of time or money and the final design is produced, as can be seen in Figure 1.2. A thorough discussion of design processes and philosophies can be found in (*McKenney*, 2013).

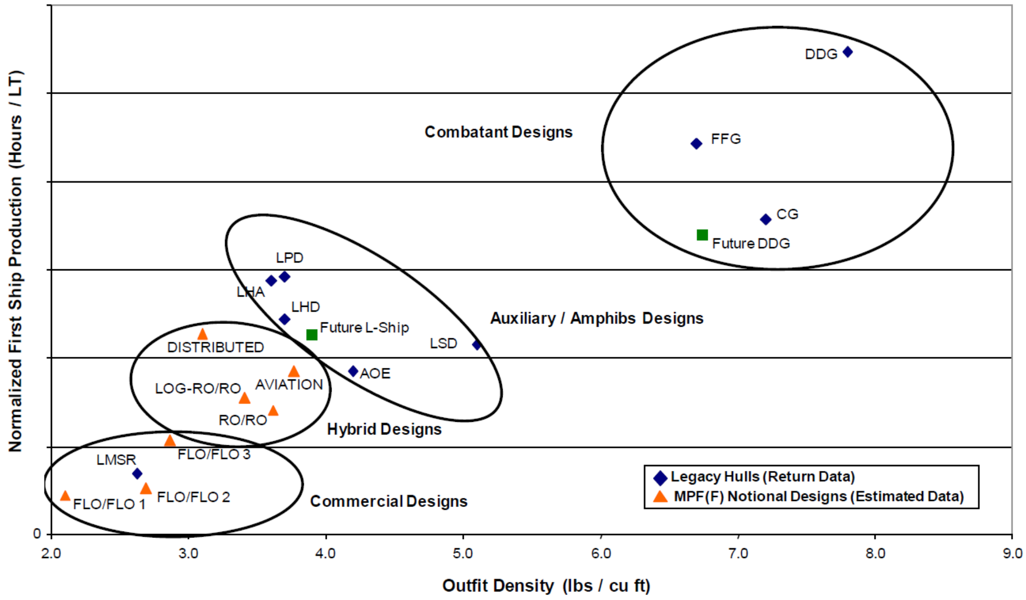
Ships, and other physically large and complex system such as offshore oil and gas rigs, large scale public transportation projects, or chemical processing facilities, are unique in the design world because they are typically extremely expensive, one-off entities that must be designed from the ground up without the use of prototypes (*Andrews*, 2011). This exacerbates the lack of knowledge of the design space; a problem which will only get worse as the designers attempt to the push the envelope of naval design and create revolutionary designs based on new paradigms, like the US Navy's next generation guided missile destroyers (the Zumwalt Class) and Littoral Combat Ship. Without the help of a validation phase for the design process, the disconnect between the requirements creation and realization will continue to grow.

Naval ships, and combatants in particular, are unlike any other type of ship when measured according to their complexity. Figures 1.3 and 1.4 show two commonly accepted analogies for ship complexity: outfit density and compensated gross tonnage. It can be seen that surface combatants have between two and four times the outfit density and compensated gross tonnage of commercial ships, while submarines are an order of magnitude more complex. This added complexity makes the early stage design decisions even more crucial because the density of outfit and difficulty of installation increase the probability of unforeseen and sub-optimal interactions between systems. Also, higher complexity goes hand-in-hand with higher costs, longer production times, and increased risk of cost and schedule overruns (*First Marine International*, 2005). The high outfit density and system complexity make the removal or replacement of equipment extremely labor intensive and expensive (*Schank et al.*, 2009), so it is imperative that system(s) design(s) be correct.

Along with, or because of, the higher complexity of naval designs, the cost of naval ships has risen between 7% and 11% over the past sixty years (Figure 1.5. One quarter of this increase is due to increasing complexity of which the ship's mission, arrangements, and systems are major drivers (*Arena et al.*, 2006).

From the naval architect's perspective, mission requirements may not be negotiable, though *Andrews* (2003) argues that this is not proper because ship design is a "wicked problem." In such a problem, the definition and solution are so intertwined that they cannot be decoupled. There is no stopping criteria, but there is only one opportunity to create the right answer. Beyond this, there is no optimal answer, just solutions that are good and bad (*Rittel and Webber*, 1973).

A ship design's arrangements are absolutely under control of the naval architect and



Ship Production hours increase with density and fall into predictable groupings.

Figure 1.3: Relationship between outfit density and ship production hours (*Keane Jr.*, 2011).

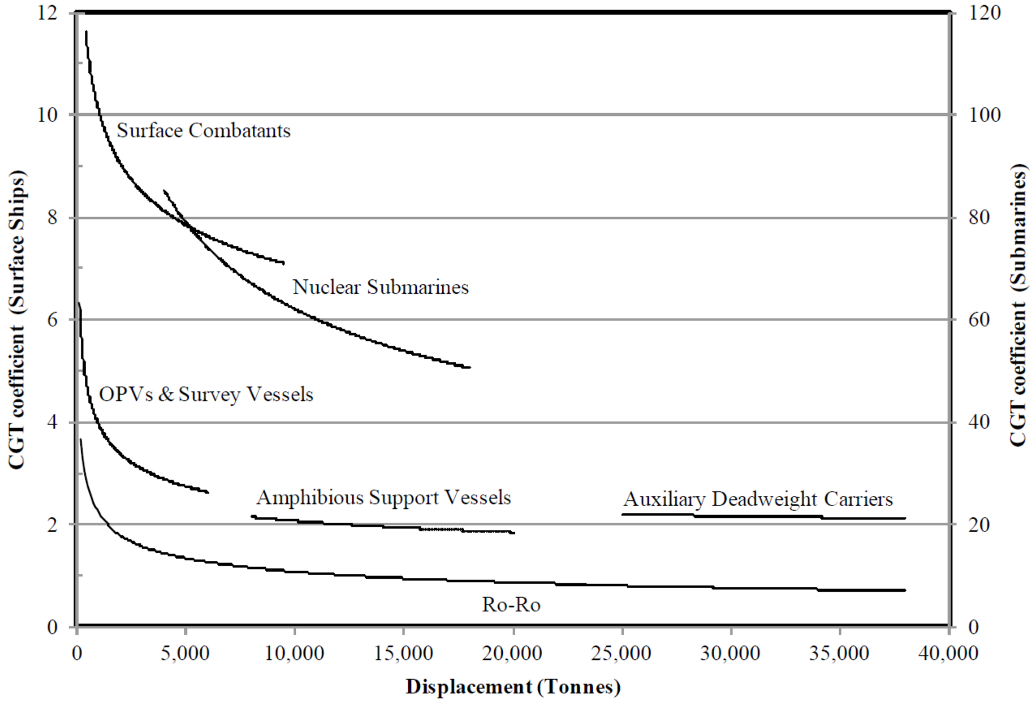


Figure 1.4: Compensated gross tonnage coefficient by ship type (*Craggs et al.*, 2004).

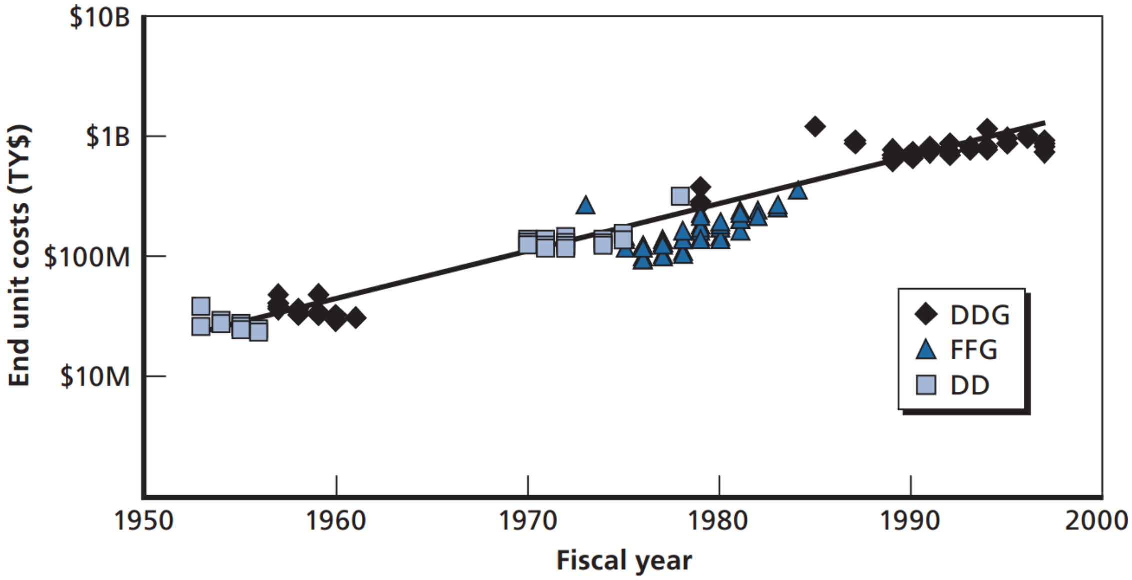


Figure 1.5: Cost escalation for selected surface combatants (*Arena et al.*, 2006).

much work has been done in an effort to improve the creation of general arrangements. The three primary tools for early stage arrangements creation are the Design Building Block approach from the University College of London (*Andrews and Pawling*, 2003; *Pawling*, 2007), the packing approach of the Delft Technical University (*van Oers*, 2011; *van Oers and Hopman*, 2012), and the Intelligent Ship Arrangements program of the University of Michigan (*Parsons et al.*, 2008; *Gillespie*, 2012). An extended discussion of the preliminary ship design process, mainly in the context of ship arrangements, can be found in *Andrews et al.* (2012). While this dissertation draws upon the methods introduced for understanding the general arrangements design, it is not focused on the general arrangement problem.

Gillespie (2012) states that system selection may not be negotiable from the naval architect's perspective, and while this may be true, the naval architect still has authority over where and how the systems are implemented within the ship. Unfortunately, in the early design stage, there is a lack of tools to properly and confidently make these decisions. The goal of this dissertation is to provide a new prospective on the ship systems design process and give a naval architect the tools he or she needs to make informed decisions about systems design and layout.

1.2 Current research

The main challenge of this research was to create a new way to analyze individual distributed systems, the interactions between systems, and the evolution of the systems and their behaviors in both the design and operational environment, **without** the use of sophisticated 3D computer aided design (CAD) systems. 3D CAD models simply take to long to create and analyze for them to be useful in the early, conceptual design stage. This research attempts to create a method similar to the concept of design sketching (*Pawling and Andrews*, 2011), but for distributed systems instead of entire ships. The goal is to create a way to analyze distributed system design concepts at the earliest of design stages to give the naval architect as much information, as early as possible, and understand the effects of distributed system design decisions and changes on the system itself and all related systems. Such a method would reduce the chance of design failing due to a lack of information about system complexity and systems interactions in the early design stage.

1.2.1 Scope

The goal of this dissertation is to introduce a new perspective for analyzing ship distributed system designs and the interactions between designs that can be used at all stages of the ship design in lieu of visual methods. This dissertation does not attempt to solve the entire ship distributed systems problem. Rather, a perspective that facilitates the jump between design rules of thumb and low fidelity models to high fidelity 3D product models is explored. There is no exploration into the creation of a system for the automated generation of distributed systems, though that is an extremely interesting topic for future exploration. Also, this research focuses on distributed system robustness and interactions analysis with no work on system component sizing. Additionally, this work seeks not only to expand the application of network theory to ship design, but also to provide insights and additions to the field of network theory.

1.2.2 Contributions

The main contribution of this thesis is a new method for bringing design information that is typically only available in the later stages of a ship design to the earliest stages of ship design. This method does not remove the complexity of a ship design, but makes its inherent complexity more legible.

The need for such a method came from the recognition for the need to shift from the 3D CAD paradigm of creating and analyzing distributed systems design in the later design stages and focusing on methods that can be used in the early design stage to bridge the gap between concept studies and detailed geometric models. Additionally, the was a recognition of the inherit differences between distributed systems and other ship systems, which requires a new medium for storing, displaying, and analyzing distributed systems information.

In support of the main contribution, several supporting contributions are presented in this dissertation. These contributions are summarized in the remainder of this section and split between contributions for the analysis of solitary systems and contributions for the analysis of multiple systems.

1.2.2.1 Contributions to single system analysis

- Creating a new network complexity measure based on the network concepts of planarity and communities.
- Identifying and applying network metrics for determining potential choke-points within a ship's passageway system.
- Creating a new betweenness measure "goal betweenness" which added the concept of a goal node to previously developed betweenness measures.
- Identifying and applying network methods for analyzing system robustness.
- Abstracting additional system complexity information into the edge weights of distributed system networks.
- Grouping physical system communities into communities and using the community structures as a predictors of system interactions.

1.2.2.2 Contributions to multiple system analysis

- Demonstrating that multiplex and multislice network structures are applicable to more than just social networks.
- Applying multiple resolution, multislice analysis to analyze change propagation in a design process.
- Using network science to demonstrate the advantages of zonal power distribution systems over radial power distribution systems in the early stage design evaluations.

- Tracking distributed systems design evolution using a multi-slice network structure.
- Analyzing distributed systems interactions using a multiplex network structure.
- Creating the time-dependent multiplex network structure and using it to evaluate the evolution of distributed system interactions over sequential iterations.

1.3 Overview of dissertation

This dissertation is divided into 7 chapters. This first chapter served to introduce the complexity of naval design as well as preview the contributions of this dissertation.

Chapter 2 presents an overview of the ship distributed systems design process as how it is done today as well as other research conducted to improve the process. This chapter will identify the shortcomings of the current process, the biggest of which is the disjointed nature of the design process. This disjointed process creates a huge amount of disparate information which causes a complicated problem to become truly complex. This complexity is what causes designs to fail. The chapter ends with a discussion of necessary characteristics that a new distributed systems design method needs to possess.

Chapter 3 introduces complex systems theory as a new paradigm for distribute systems designs, and presents network theory as a structure with which to model, analyze, and evaluate naval distributed systems designs. Network theory structures and metrics are introduced and connected to relevant naval architecture concepts and the gaps between the two subjects are described.

Chapter 4 discusses a novel planarity-complexity metric, developed for this dissertation, that was created as a new way to quantify the complexity of planar or near-planar networks. These kind of networks are of interest because the are often found in real networks such as those used to represent ship distributed systems.

Chapter 5 demonstrates the validity of network theory as an approach for analyzing distributed system designs. This chapter both provides a new naval context for traditional network metrics as well as introduces the use of multiplex and multislice networks for the modeling of real systems for the first time. Additionally, a new time-dependent multiplex network structure, developed for this dissertation is introduced.

Chapter 6 expands on the concepts presented in Chapter 5 and shows how network theory can be used as the keystone of a new method for analyzing the implications of distributed systems design decisions during the course of a ship design.

Chapter 7 concludes the dissertation with a recapitulation of the novel contributions of this dissertation followed by a discussion of potential avenues for future study.

CHAPTER II

The Ship Systems Design Process

The purpose of this chapter is to capture the current state of the ship distributed systems design process and identify the issues therein with the goal of motivating the network theory-based solutions presented in the remainder of this dissertation. This chapter starts with an overview of the ship design process to give context to the distributed systems design process. Then, tools used in both the total ship design process and the distributed systems design process are cataloged. Lastly, the chapter concludes with a discussion of the key characteristics that must be included in a new ship distributed systems design method.

This chapter will show that the ship design process is a disjointed event that relies on disparate and incomplete information to make decisions about complicated tradeoffs and interactions. These frequently ill-informed decisions cause the design to gain unintended complexity when they are transitioned from the early design stage to the detail design stage. This results in suboptimal system designs and layouts, an example of which is pictured in Figures 2.1 and 2.2.

Figure 2.1: Example 1 of a sub-optimal distributed system layout. A jet fuel pipe, in purple, routed through the enlisted mess of the USS John C. Stennis (CVN-74) (*Blumenfeld*, 2012).

Figure 2.2: Example 2 of a sub-optimal distributed system layout. Again, the jet fuel pipe, in purple, can be seen in the foreground. In the background, two large HVAC ducts, circled, are shown running from the deck and through the mess space, causing an obstruction to personnel movement.

These issues arise because a ship has a finite space. Room for extra piping runs or HVAC ducts cannot be made simply by “bumping out” a wall or window. Unlike architecture, form does not always follow function in ship design. A ship is a packing problem and as the density of systems on board a ship increases the problem becomes more difficult.

2.1 The ship design process

The process of designing a ship requires the coordination of vast amounts of information. This information comes in various forms (drawings, results, variables, etc.), developed by various people, produced in various software environments containing different levels of fidelity and uncertainty, and created by different analysis tools that require unique inputs. For this reason, ship design is usually defined in stages: **concept**, **preliminary**, **contract**, and **detailed** design (*Gale*, 2003). Concept design defines the basic ship requirements, preliminary design analyzes potential alternative designs, contract design finalizes the overall design of the ship including major systems and dimensions, and detailed design produces the developed design necessary for construction.

Concept design is the process by which a ship’s mission and required performance goals are defined. This is typically done through the use of parametric models that allow designers to quickly create many different design alternatives. These alternatives are often created by a single designer or a small team of designers working closely with the ship owner. These parametric models are based on past designs and can be inadequate if new or unconventional designs are attempted. At the completion of this stage the ship’s concept of operations is known as well as rough dimensions and hull type.

Once a design concept is decided upon, the design process progresses to the preliminary design stage. At this stage, there is a significant increase in both the size of the design team and the cost of the design work being performed. Preliminary design is characterized by a series of trade-off studies used to refine the design and select major ship systems. This stage of design can be visualized using the design spiral (Figure 2.3). Often, at the completion of the design spiral, the design is not viable due to any number of issues and the process is begun again. The design will go through many iterations of this process until a satisfactory design is produced. Research is ongoing within the topic of set-based design with the goal of eliminating the need for iteration at this stage (*Gray*, 2011; *McKenney et al.*, 2011; *Singer*, 2003; *Ward et al.*, 1995), but set-based design has only been used for one U.S. Navy design (*Buckley and Singer*, 2013).

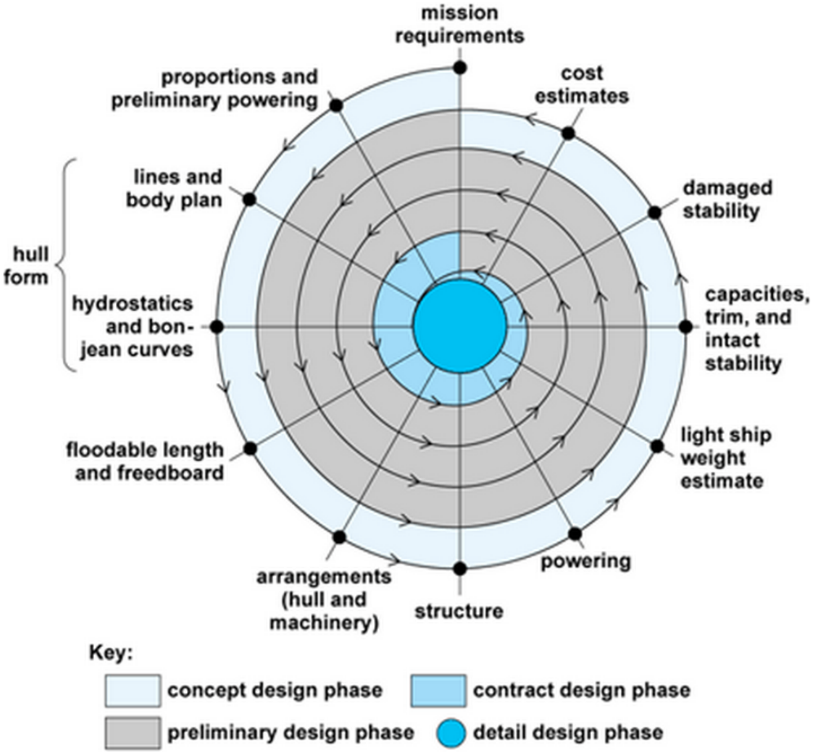


Figure 2.3: The design spiral (*Larsson and Eliasson*, 2007).

This process is aided by the use of low fidelity design tools that include simple CAD drawing systems such as AutoCAD, naval architecture specific programs used for hullform generation and hydrostatic analysis such as Maxsurf, and general spreadsheet programs for tracking weights and centers of gravity. All-in-one design synthesis tools also have been created in an attempt to streamline this process (*Code 2230 NSWC-Carderock*, n.d.), but these synthesis tools also suffer from a low level of fidelity. This low fidelity is necessary due to the iterative nature of the preliminary design process. Designers simply do not have the time to generate complex and expensive design models when many prospective designs are being created. These tools will be further discussed in the following section.

When the main design characteristics are confirmed in preliminary design, the design moves to the contract design phase. This progression once again causes an increase in the size of the design team and the engineering effort. The completed contract design will be used to solicit bids for construction from shipyards. Therefore, a more in-depth analysis of the individual ship systems is required. This requires higher fidelity, specialized tools which necessitate the growth of the design team. Unfortunately, this increase in size and specialization often reduces the communication between different parts of the design team in a phenomenon known as "over the wall engineering" (*Collette*, 2011). Concurrent engineering methods have been proposed to facilitate information sharing(*Bennett and Lamb*, 1996; *Keane and Tibbitts*, 1996), but there is still substantial segregation of subject matter experts during the ship design process (*Code 223 NSWC-Carderock*, 2012). The contract design phase is the last stage where major design changes can be done at a moderate cost.

The final design stage, functional design, is when the design is transitioned into detailed drawings and bills of materials required for construction. The minute details of the individual systems are finalized, including scantling sizes for the hull structure, pipe diameters for plumbing systems, and routing for electrical systems. This stage requires the highest fidelity models, including full FEA structural models and potentially life-sized system mock ups (those can be done in early stages if there is sufficient design risk). This is the stage where interactions and conflicts between different systems become clear; unfortunately, this is also the stage where redesign is the most costly. Due to the high cost of redesign, even if extra space or some other opportunity for design improvement is found, it is typically not exploited, though lessons learned are typically incorporated into future designs.

Sometimes this redesign is authorized, or if the conflict is not caught, it is left for a shipyard worker to make adjustments on the spot. This can lead to systems that are built very differently from how they are designed. The ability to identify such interactions, or the potential for such interactions, early in the design process would be useful for reducing such rework and therefore reducing the time and cost of ship design while potentially increasing its quality (*Government Accounting Office*, 2002). A framework for identifying such interactions is the goal of this research.

2.2 Ship design complexity

The research presented in this dissertation is not the first attempt at quantifying ship design complexity, but it is the first work that does not assume complexity is a purely physical or geometrical concept. Previous works consider complexity through the lens of producibility, where this dissertation considers complexity from the point of view of interactions.

Caprace and Rigo (2010a,b) consider a ship's complexity to be a function of the complexity of producing the shape of the hull, the complexity of the assembly, and the complexity of working with the structural materials. These three facets are combined into a global complexity factor for the entire ship design. Further research has been conducted on ways to quantify the complexity of producing the hull shape (*Parsons et al.*, 1998) and the complexity of the assembly (*Rigterink and Singer*, 2009; *Rigterink et al.*, 2012, 2013a). The differences in complexity of working with different grades of steel and aluminum, as well as higher strength metals like titanium, is well known (*Van Dokkum*, 2007). A further overview of methods for handling complexity and information growth in ship design is detailed in *Gaspar et al.* (2012).

These attempts to measure complexity have all relied on physical attributes of the ship design to model complexity, for example looking at spacings in grillage structures or the amount of curvature of a hull plate. Distributed systems to not lend themselves to such physical models for two reasons. First, the distributed system design from ship type to ship type is vastly different that it cannot be captured using a small set parameters describing the spacings. Second, the complexity of distributed systems lies not only in their individual structures, but also into the interactions of different systems. To capture this interaction information using classical modeling methods requires multiple 3D models. Making these models is not tenable to early stage design because they simply take to long to make. That is why this dissertation presents a new way to model and analyze distributed systems designs.

2.3 Ship distributed systems design

Figure 2.4 shows another representation of the preliminary ship design process with the major design areas highlighted. It can be argued that distributed systems design is part of each design area, but in this representation, distributed systems design is only part of the propulsion/power/machinery design area, shown in Figure 2.5. It should be noted that each activity group interacts with every other activity group and the there is both an external design spiral across the entirety of the preliminary design stage as well as internal design spirals within each activity group (Figure 2.6).

It can be seen that the distributed design process is an integral part of the overall ship design process, but there is a growing disconnect between distributed system designers and naval architects. This separation is evident in the splitting of the ship design discipline into naval architecture and marine engineering. Even in the US Navy, ship design and integration research is conducted at a different facility than machinery research. This division makes sense at one level because the two are very different disciplines, with the naval architect primarily focused on the ship's hull design, hydrodynamics, hydrostatics, and arrangements and the marine engineers focused on the various systems that go into the design. The design of the individual systems requires specialization, as shown in Figures 2.7 and 2.8, and attempting to design all the systems as well as the ship itself would quickly overwhelm a single naval architect. The job of the naval architect is to be the systems engineer and properly integrate all the systems into the design.

The relationship between arrangements and systems design is tightly coupled, so it is necessary for the naval architect and the systems designer to work closely together to ensure an integrated, converged design. This process is aided by a host of computer tools that will be detailed in the next section. While these tools do help with the overall design, they tend to require high levels of fidelity to be useful, so a new method that requires lower fidelity models and is easily understood by the both the naval architect and the systems designer would be preferred. Such a method would not only assist in the ultimate step of integrating the final design, but also give the naval architect and the system designer a common language in which to pass ideas and intermediate designs back and forth.

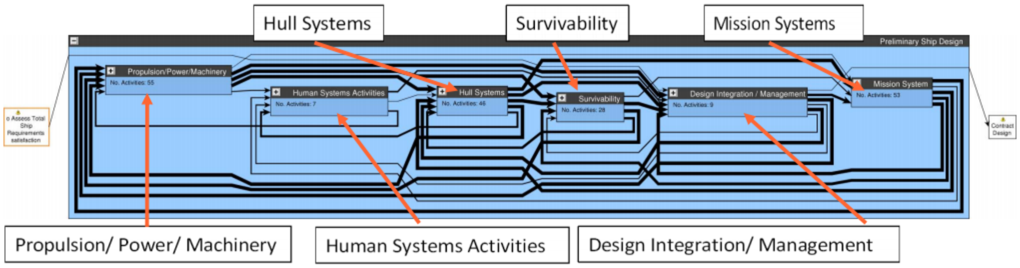


Figure 2.4: A breakdown of the preliminary ship design stage (*Code 223 NSWC-Carderock*, 2012).

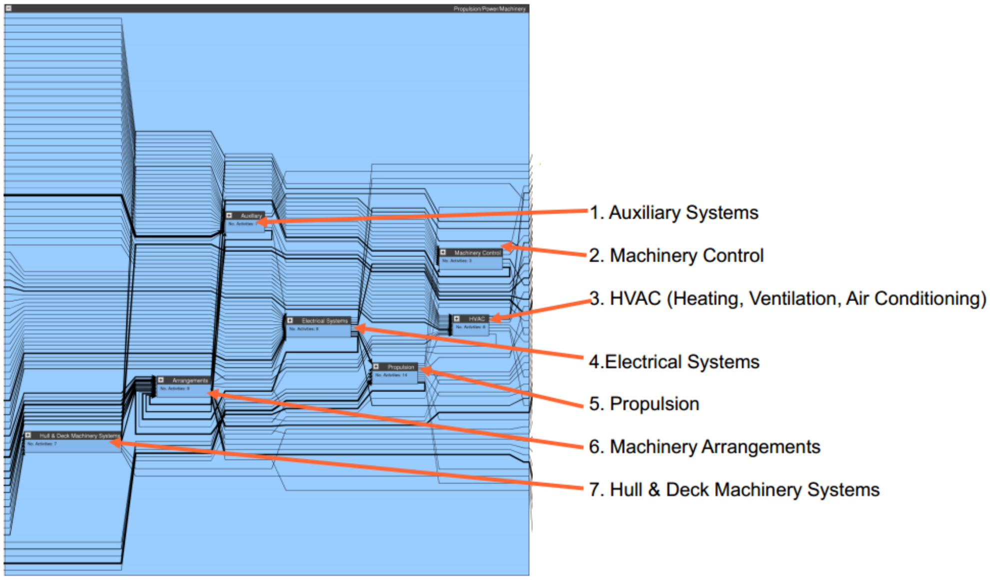


Figure 2.5: The propulsion/power/machinery preliminary design activity, highlighting the distributed systems design groups (*Code 223 NSWC-Carderock*, 2012).

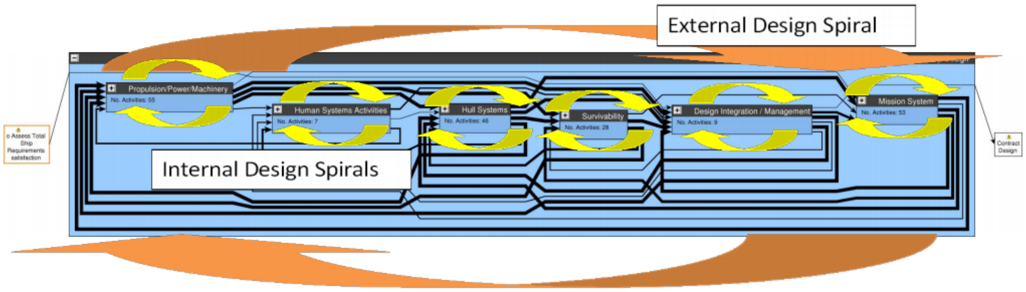


Figure 2.6: The external and internal design spirals of the preliminary design stage (*Code 223 NSWC-Carderock*, 2012).

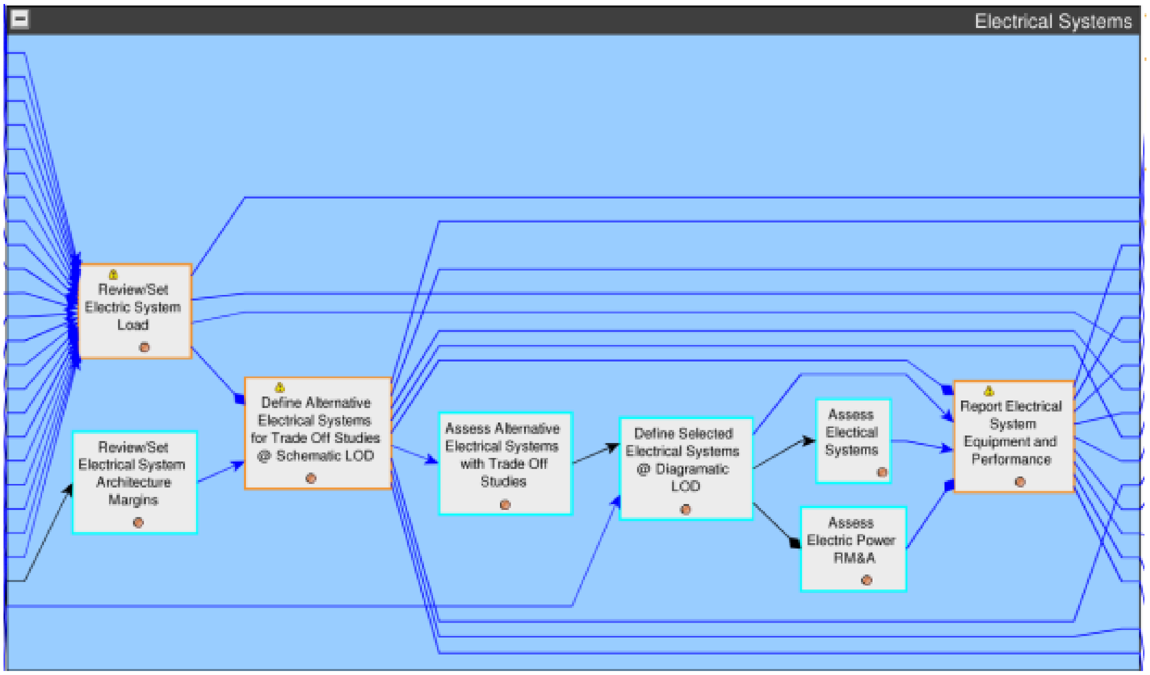


Figure 2.7: The electrical system design activity group (*Code 223 NSWC-Carderock*, 2011).

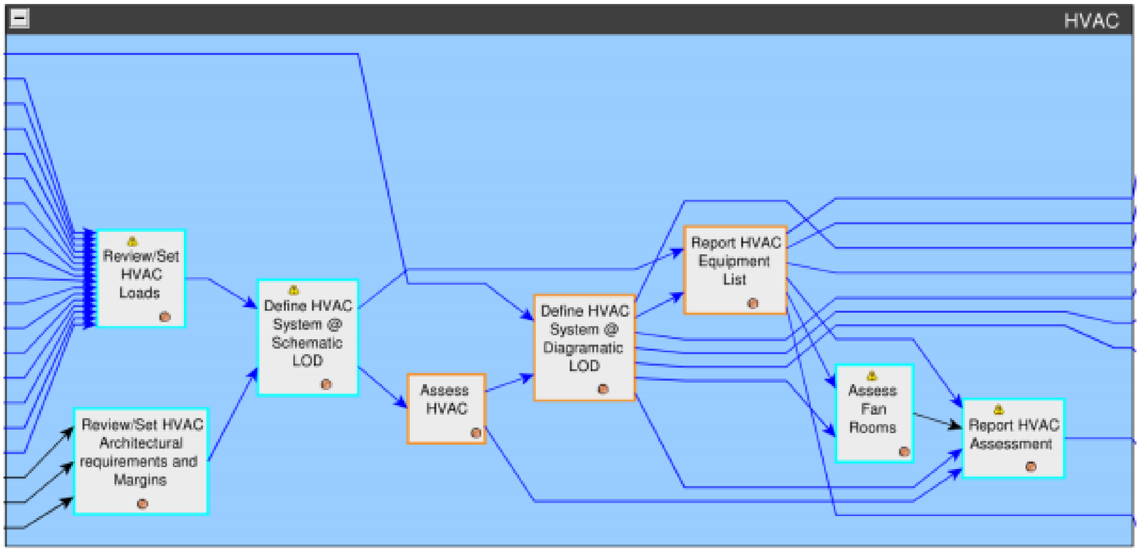


Figure 2.8: The heating, ventilation, and cooling activity group (*Code 223 NSWC-Carderock*, 2011).

2.4 Computer-aided design

In 1973, *Gallin* stated that "ship design without the computer [is] no longer imaginable" and over the intervening forty years, ship designers have shifted from using computers as one of many tools to using the computer almost exclusively (*Ross*, 2003). Some would argue that designers have become too reliant on the computer, and the level of man-machine interaction needs to be rebalanced. The human element of design needs to concentrate on problem formulation and results contemplation while using the computer analyses as assistive tools, not as a infallible solutions (*Nowacki*, 2009). This is part of the reason that the network-based methods introduced in this dissertation are designed for analysis rather than generation.

The most ubiquitous computer tools in ship design, and engineering design in general, are computer-aided design (CAD) packages. These programs allow designers to sketch up ideas, then add more and more detail to the model, as the design process progresses. Depending on the program, the designer is able to analyze certain properties of a ship design. For example, Rhinoceros and its maritime extension, Orca, will output a plethora of information useful for a naval architect, such as righting arm curves, hydrostatic coefficients, and speed and powering estimates (*DRS Technologies Inc*, 2013). ShipConstructor, based on AutoCAD, has similar capabilites and is designed to allow multiple stakeholders to simultaneously design different parts of the ship and ship systems within the same model (*ShipConstructor Software Inc.*, 2012). Maxsurf is another tool that aids in hull design, hydrostatics, and structural design (*Ross*, 2003).

The models created in CAD programs are desirable because they have the ability to grow with the design. What starts as a concept in AuoCAD can be built-up, then analyzed using any of numerous computer-aided engineering tools (discussed in the next section). Following this analysis, the model can be modified and the process repeated until the design is converged. This converged design can then be transferred to a computer-aided manufacturing program to develop data for welding, cutting, bending, and nesting of plates.

During this entire process, a product model program holds all the information about the design including geometries, equipment weights, and distributed systems locations. Products like CATIA can be used to completely model a ship design and analyze physical system interactions such as clearances and obstructions. While this is extremely helpful for designers, the fidelity of design models required to do such an analysis is not available in the early design stage, when there is the most flexibility to correct conflicts or other problems. Additionally, geometric models can only be used to detect interactions at the local, physical level. They cannot identify the similarities or differences between the structure of two systems that may lead to a coupling of failures between the two systems. Lastly, systems like CATIA are extremely expensive and therefore usually not considered within the realm of conceptual and early stage design.

In general, the use of CAD in ship design is not a bad thing. The issue with CAD stems from the size and expense of the models which makes them impractical for conceptual level designs. Essentially, once a 3D CAD model is created the design has moved on from the conceptual stage to the detail stage. If this is the only method for discovering complex system interactions, then a design is destined to fall in a costly spiral of CAD redesigns. To prevent the redesign later, it is imperative that a method for elucidating system interactions is available in the early design stage.

2.4.1 Design information sharing

As designs grow, so do the teams working on them, which means the efficient and effective sharing of information is paramount. The previously mentioned tools are useful for this because sharing things like electronic CAD files is generally fairly easy. The issue becomes tracking changes and the intent and effect of said changes.

One method of managing this information is called a "smart drawing." A smart drawing is a CAD drawing with external links to network-accessible relational databases and related documentation, such as memos, spreadsheets, and analytical results (*Dong*, 1997; *Dong and Agogino*, 1998). By selecting an element in a CAD file, a designer is automatically able to download the linked documents explaining the element. The attached documents, as well as the base CAD model, are controlled so as to prevent unauthorized edits while tracking the authorized changes.

The shortcoming of this technology is that it still relies on the visual medium of CAD, as well as the designers ability to effectively articulate the rationale behind their design decisions. Additionally, there is no automated analysis of interaction between systems, nor analysis of the effects of changes. The onus is on the individual designers to verify that their changes have no negative ramifications towards the overall design beyond the basic, local physics.

In addition to capturing what is occurring as a design progresses, there is a desire to understand why a decision was made. This desire to capture design rationale serves two purposes. One is to preserve best practices and prevent unnecessary rework in the future. Documenting best practices is especially critical as the naval architecture work force continues to age and retirements threaten to take many years of knowledge with them (*National Science Board*, 2012). Second, capturing design rationale allows for a critiquing of that rationale which could lead to improvements in the design process (*DeNucci*, 2012).

2.4.2 Naval architecture design tools

Computer-based naval architecture design tools can be split into two categories; **analytical** and **generative**. Analytical tools, also known as computer-aided engineering tools, require a designer to enter some type of model for the tool to then analyze. This model could be anything from a series of parameters in the case of a propeller optimization program, to a complex 3D model that is used for a finite element analysis. The analytical tools available to a naval architect are innumerable with some being specifically designed for the ship design realm, while many others are designed for general engineering design and have been adopted for use in naval architecture design (*Latorre and Vasconcellos*, 2002). Table 2.1 presents some computer aided engineering tools along with their capabilities.

Table 2.1: Examples of computer-aided engineering programs (*Ross*, 2003).

Program Name	Capability
NavCAD	Resistance and power prediction
GHS	Hydrostatics, stability, longitudinal strength
MAESTRO	Structural design and optimization
NASTRAN	Finite element analysis (FEA)
SafeHull	FEA for yielding, buckling, and fatigue strength of ship structures
ShipWeight	Weight and center of gravity estimation

Generative tools, or computer-aided synthesis tools, are used either to create a full ship model or a specific part of ship design such as a set of general arrangements. The US Navy's ASSET program is an example of the former. ASSET works in two steps. First, a feasible design is generated, then this design is analyzed (*Neti*, 2005). The process begins by inputing a parent hullform, a series of ship requirements, for instance speed, range, propulsion type, etc., and the desired crew size. The program then runs a series of modules and attempts to balance weight with displacement, confirm the ship is stable, and confirm that there is enough deck space for the required facilities. Once a feasible design is obtained, ASSET proceeds to analyze the design to estimate its top speed, range, sea keeping ability, etc.

The final output from the program is a rudimentary inboard profile showing the outline of the hull and deckhouse, watertight bulkhead locations, location and size of the propulsion system, major structural cross sections, ship work breakdown structure (SWBS) document, estimated electric loads, and estimated required areas for mission spaces.

In addition to the basic ASSET program, there is the LEAPS toolkit (*Code 2230 NSWC-Carderock*, n.d.), which includes tools for hydrostatics, hydrodynamics, design space exploration (*Gray et al.*, 2013), manpower estimation (*Alion Science and Technology*, 2011; *Doerry*, 2006a), and general arrangement generation (*Parsons et al.*, 2008). Beyond this toolkit, there is ongoing research into optimized manning levels dependent on required ship systems (*Office of Naval Research*, 2011; *Scofield*, 2006; *Singer et al.*, 2012).

The automatic, or nearly automatic, generation of general arrangements is the next step for programs like ASSET. There are currently three well developed tools: the University College of London (UCL) Design Building Block approach (*Andrews and Pawling*, 2003, 2008), the Delft University of Technology (TUD) packing approach (*van Oers*, 2011; *van Oers and Hopman*, 2012), and the University of Michigan (UM) Intelligent Ship Arrangement approach (*Nick*, 2008; *Parsons et al.*, 2008). Table 2.2 summarizes these approaches.

These generative tools rely on large databases and rule sets to produce potential designs. Such databases and rule sets are labor intensive to create and can be difficult to understand. Outputs of these tools are displayed in visual representations of the

Table 2.2: Comparison of early-stage ship arrangement tools (*Gillespie*, 2012).

	UCL	TUD	UM
Fullship design	Yes	Yes	Deck plans only
Number of dimensions	3D	2.5D or 3D	3D
Driver	Volume	Volume	Area
Layout generation	Manual	Automated	Automated
Optimization scheme	Manual	Genetic Algorithm	HGA-MAS
Concepts generated	Few	Hundreds	Hundreds
Adaptable hull shape	Yes, manual	Yes, automated	No, fixed in ASSET

layouts and then it is the prerogative of the designer to determine if a layout is acceptable with suboptimal interactions that were not included in the rule sets. More in-depth analyses including personnel movement (*Casarosa*, 2011) or piping layouts (*Asmara and Nienhuis*, 2006) require additional software packages linked to these original programs. This increases both the time and expense of doing such analysis but also requires additional information that may not be available in the early design stage. Currently, there are attempts to combine all three tools in an effort to elucidate more detailed design drivers (*Pawling et al.*, 2013).

These existing general arrangement tools are capable of generating feasible layouts but lack the underlying knowledge regarding why layouts are configured as they are. These design drivers have been explored using visual analysis (*Pawling*, 2007), but this analysis still relies on spatial and numerical models that are exclusive to one software package.

Network-based methods, developed by *Gillespie* (2012), can be used to bridge the gap between designer intent and the design, without the need for spatial models, as they provide a unique perspective to the traditional view of ship arrangements by relying on the relationships between shipboard elements and components. A network analysis expands the scope of the general arrangement process and helps to understand complexity by revealing underlying character and structure, providing insight into the function of a complex system.

Gillespie's work analyzes a single type of interaction, namely the desire of spaces to be adjacent or separated from other spaces. A method designed for distributed systems analysis and evaluation would need to be able to handle individual systems, but also multiple systems and their interactions.

2.5 A new method for ship distributed systems design

It can be seen that the ship design process is a disjointed event which leads to the creation of “disparate” ship design information. An example of disparate information is a ship's required power. It is a fundamentally different kind of information than the electrical load that a radar system requires. It is based on a different system analysis, completed by a different set of designers, using different tools in different programs with different levels of fidelity. In a ship design, such disparate information in unavoidable, but the sources of such information can be managed, using network theory, so that the final amalgam of information forms a cohesive ship design.

In an attempt to bring disparate information together, complex computer-aided design (CAD) systems and large product life-cycle management (PLM) processes have been developed over the last thirty years. The issue with these product models is that they are most effective at the later stages of design which means analysis of the interaction between different systems can only be done once a large amount of time and money has been invested into a design. At this stage, it is often too late to redesign large components of the ship. Instead, small patches are engineered to mitigate undesirable interactions rather than remove them all together. Network theory allows designers to bring the information together at a much earlier stage, without the need for multi-million dollar CAD or PLM packages, to begin to analyze and understand system interactions at a stage where major changes are practical and possible.

As identified previously, the first source of disparate design information is the multitude of people working on a design. These designers all have their own specialties, speak in their own languages, and go about their task according to specific methodologies. Network theory gives these different groups a unified framework to put their thoughts into, and more importantly, a common language in which to communicate. Additionally, network theory could eliminate the need for these various groups, as even a very large network can typically be understood by a single person through the use of network tools. Lastly, network theory can potentially allow one to decouple the people dependencies associated with traditional methods.

Similar to how there are many different people working on a design, there are many different analysis tools being used in a design, each requiring different operator skills and different information. While detailed computational fluid dynamics analysis for hullform resistance and finite element analysis for hull strength will still be required as a design goes forward, they are not required at the earliest design stages. In early design, it is more important to have one set of tools that can be widely applied. These broad tools come from network theory. Network theory and its methods are agnostic to what a network represents; just that it is represented in a network structure.

Different kinds of analysis also require different amounts of detail. For example, a ship egress analysis requires a full general arrangement and crew roster to run a detailed discrete event simulation while creating the general arrangement could just require a simple sketch on a piece of paper or an extensive database of required areas and volumes. Network theory gives a baseline of information as different levels of fidelity within a network are not possible.

The overarching reason these different sources of disparate information exist is the ship systems themselves. Every system is different and requires a different type of analysis, while the entire group of systems need to be analyzed together. Individual and group analyses need to be done throughout the ship design process, as opposed to designing the systems individually early then combining them towards the end of the process, which is the current paradigm. Network theory, and multiplex networks in particular, allow for analysis of both the individual and aggregate system. Each system can be represented as its own level in the overall multiplex structure which allows for analyzing the electrical system or the HVAC system using different metrics. Then, since all the systems are combined in the same larger structure, they can be easily analyzed as a whole. If the systems were placed in a large, simplex network, the overall structure and interactions would be possible, but quickly assessing the individual systems would be difficult if not impossible.

CHAPTER III

Network Theory and Distributed Systems

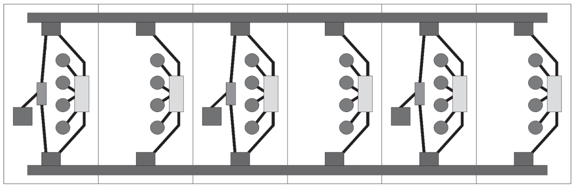


Figure 3.1: A zonal shipboard electrical-distribution system (*Doerry*, 2006a).

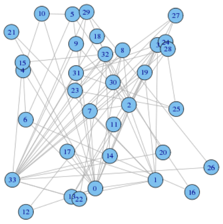


Figure 3.2: The Zachary Karate Club network.

Figures 3.1 and 3.2 show a hypothetical zonal electrical distribution system and a classic relationship network used by network scientists. At first glance, the similarities between the structures in the two figures quickly become evident. In fact, the similarities are more than skin deep and the same methods that were used to create the latter figure can be used to create a network representation of the former figure, and then network theory tools can be used evaluate this representation, and therefore the original electrical system. This chapter will introduce and motivate the use of complex systems theory and network theory to reconceptualize, represent, and evaluate the ship distributed systems problem.

3.1 Complex systems theory

Complex systems theory breaks systems into four broad categories: simple, complicated, complex, and random or chaotic (*Rickles et al.*, 2007). The four groups are defined as follows:

- **Simple:** systems with very few parts that behave according to very basic laws or rules. A simple system is very easily understood; take for example the six basic machines: the lever, wheel and axle, pulley, inclined plane, wedge, and screw (*Anderson*, 1914). These machines are easily conceptualized and the input can be directly linked to the output via $P_{in} = P_{out}$.

- **Complicated:** systems with many parts that behave according to very basic rules or laws. Complicated systems are typically built off the combination of many simple systems. For example, a grandfather clock is a combination of a pulley system and screw. A cursory investigation of the clock may not reveal exactly how it works, but after a sufficient time studying the mechanics, one can fully understand its operation.

- **Complex:** systems with a large number of subunits that have a high level of interaction. These subunits can also be composed of smaller sub-subunits. These interactions result in a rich, collective behavior that feeds back into the behavior of the individual parts. Complex systems, and specifically the interactions of the building block units often cannot be understood, though it is typically possible to model the behavior of the total system. The outputs of complex systems usually cannot be linearly mapped to the inputs.

- **Random or chaotic:** systems with very few interacting subunits that interact in such a way the overall system behavior cannot be accurately modeled. The outputs of chaotic systems are usually highly non-linear and/or stochastic.

From this prospective, a ship blurs the line between complicated and complex. It is certainly possible to deconstruct the complicated individual subsystems that make up a ship to learn how they operate (*Page*, 2009), but it may not be possible to fully understand how they all operate and interact with each other. This pushes a ship into the realm of a complex system. When a ship is considered in its entirety, from concept design to scrapping, along with the human systems that complement the physical system, it is even more evident that it should be considered a complex system (*Page*, 2010).

Beyond just being a complex system, a ship and a ship design is a system that adapts. Over its lifetime, a ship design will adapt according to both external and internal pressures. In the earliest stage of design, it will adapt according to available technologies and owner requirements. As the ship is constructed and operated, it will be upgraded as new technologies become available or new regulations are put into place. Negative adaptations will also occur: metals will rust, engines will lose power, and the ship will gain weight.

The borderline complicatedness or complexity and adaptability of a ship design makes it a fertile breeding ground for emergent behaviors. Typically, emergent behaviors are considered good things: they produce consciousness and evolution, and bolster the economy. However, in the design environment, emergence or the possibility of emergence is an uncertainty and, as stated earlier, uncertainties are anathema to design. The possibility of emergent behavior is akin to the concept of unknown unknowns, introduced by Rumsfeld (*Federal News Service*, 2002), which have the highest potential for changing the structure of systems (*Taleb*, 2007).

One of the main facets of complex systems theory is the study of how to model complex, adaptive systems and subsequently analyze and evaluate those models. One such method for modeling and evaluating these systems is network theory, which will be introduced in the remainder of this chapter.

3.2 Network theory

Networks, which are also called graphs depending on the context, consist of a collection of points, or nodes, connected by a series of lines, or edges. These collections of nodes and edges can be used to represent a great number of objects and systems that are of interest to physicists, biologists, social scientists, and engineers. Thinking of objects and systems in this way can lead to many useful and new insights (*Newman*, 2004).

Any system or object composed of individual objects that are linked together in someway can be studied using the structure of networks. While this structure does not allow for the study of the individual components (unless another network representing that component's constituent pieces were created) nor the nature of the interactions between components, network theory promotes the study of the pattern of the interactions and connections between components of a system.

The types of objects or systems that can be represented as networks is nearly infinite, some examples include, but are not limited to:

- **Technological networks:** the Internet, the telephone network, the power distribution grid, transportation networks, delivery and distribution networks, and ship distributed systems networks.
- **Social networks:** interpersonal relationships, Facebook, and professional net-

works.

- **Information networks:** the World Wide Web, academic citation networks, patent and legal citations, and peer-to-peer networks like Napster or various torrent clients.

- **Biological networks:** biochemical networks, neural networks, and ecological networks including food webs or food chains.

Due to the flexibility of the network structure to represent any system that can be abstracted to a series of nodes and edges, it has already been shown to be useful in the analysis and evaluation of facility layouts (*Hassan and Hogg*, 1987; *Muther*, 1973; *Shouman et al.*, 2001; *Singh and Sharma*, 2005), ship arrangements (*Gillespie*, 2012), and the design processes (*Parker and Singer*, 2013).

The remainder of this chapter serves to fully introduce network concepts, structures, and measures.

3.3 Simplex networks

The networks described in this subsection are known as simplex networks, because the nodes and edges represent only one type of entity and connection, respectively. Simplex networks are represented mathematically using an adjacency matrix $\mathbf{A}$, which represents the connections between the nodes.

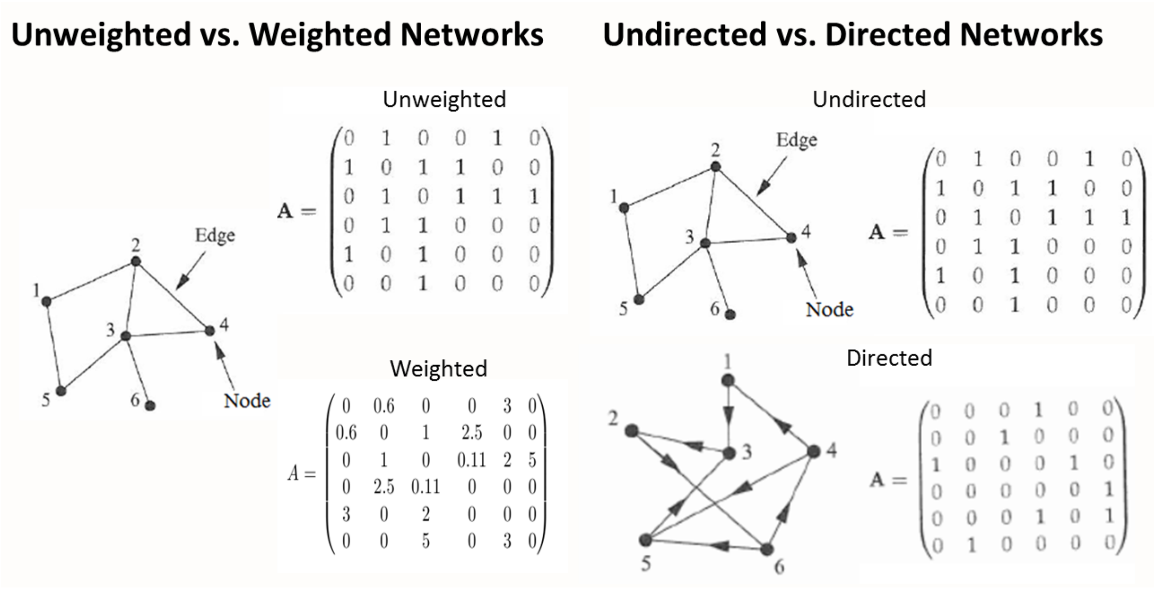
The two most basic kinds of networks are undirected and directed networks. In an undirected network, the adjacency matrix is symmetric and the links between nodes represent any connection between them. For example, a network representing a ship's passageway system would be undirected with the edges representing passageway segments and the nodes representing intersections.

A directed network has an asymmetric adjacency matrix and the edges represent a link from one node to another. A famous case of a directed network is a paper citation network (*Hummon and Dereian*, 1989) which represents papers as nodes with citations linking papers as edges. As it is only possible to cite papers previously published and citation cannot be added once a paper is published this network is directed.

In addition to directedness, networks can either be weighted or unweighted. Unweighted networks are used to show an existence of a connection between nodes, whereas weighted networks capture the intensity of said connection. The previously mentioned friendship network would mostly likely be unweighted, whereas a network used to represent commercial air traffic employs edge weights based on the number of flights between individual airports. A special case of weighted networks are signed networks, where the weights can be both negative and positive to account for adjacency or separation interactions. This type of network has been used for analyzing ship general arrangements (*Gillespie*, 2012). Lastly, there are multiple networks that allow more than one connection between the same set of nodes. Nodes can also be connected to themselves by what is called a self-edge, which is represented by an entry other than 0 on the diagonal of the adjacency matrix. Figure 3.3 displays the visual representations and adjacency matrices for an unweighted, weighted, undirected, and directed network.

3.3.1 Degree

The degree of node i, denoted k_i, is the number of edges connected to it (Equation 3.1). Building on the example of a passageway network presented in the previous section,



$$\mathbf{A} = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

$$A = \begin{pmatrix} 0 & 0.6 & 0 & 0 & 3 & 0 \\ 0.6 & 0 & 1 & 2.5 & 0 & 0 \\ 0 & 1 & 0 & 0.11 & 2 & 5 \\ 0 & 2.5 & 0.11 & 0 & 0 & 0 \\ 3 & 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 5 & 0 & 3 & 0 \end{pmatrix}$$

$$\mathbf{A} = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

$$\mathbf{A} = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}$$

Figure 3.3: Examples of network types and their adjacency matrices.

the degree of an intersection node would be the number of passageway segments that meet at that node. It is reasonable to consider a node with a high degree to be more influential to a network structure than a node with a low degree. The average degree, c, over all the nodes, n, in a network (Equation 3.2), can be used to calculate the density of the network (Equation 3.3). A more dense network suggests a more connected network.

$$k_i = \sum_{j=1}^{n} A_{ij} \tag{3.1}$$

$$c = \frac{1}{n} \sum_{i=1}^{n} k_i \tag{3.2}$$

$$\rho = \frac{c}{n-1} \tag{3.3}$$

The degree distribution over all the nodes of a network can be used to quickly judge the robustness of said network to both random failures and targeted attacks. Exam-

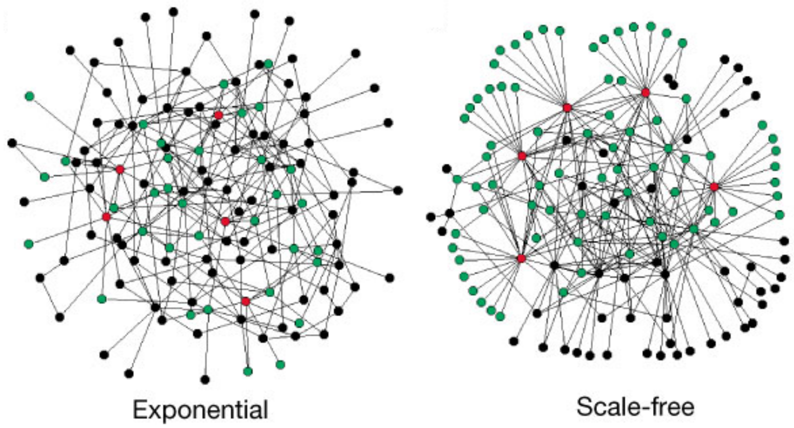


Figure 3.4: An example of networks with exponential and scale free degree distribution (*Albert et al.*, 2000).

ples of random failures in networks are a wire burning out in an electrical system or a modem being accidentally disconnected in a computer network. Targeted attacks are intentional attempts to remove key elements from a network to cause its failure. An example of a targeted attack is the wartime bombing of train tracks to hinder enemy troop and material movements. Networks that have a scale-free degree distribution ($P(k) \sim k^{-\gamma}$) are robust against random failures (*Albert et al.*, 2000) because there is a large amount of spoke nodes with only one edge connected, whereas these networks are vulnerable to targeted attack as there are a large number of critical hub nodes that keep the rest of the network connected. The Internet or the World Wide Web are good examples of such a network (*Estrada*, 2006). Networks where the degree distribution is roughly exponential, like the Erdös-Rényi network (*Erdös and Rényi*, 1960), are more susceptible to random failures. A random failure is more likely to remove a highly connected node, but for this very reason, they are less vulnerable to targeted attack since the overall connectivity of the network is not dependent on any small subset of nodes. An example of an exponential and a scale-free network are shown in Figure 3.4.

3.3.2 Paths

Whereas degree measures are concerned with the edges into and out of one node, path measures are concerned with the edges that connect various nodes. The most basic, and easiest to overlook, path-dependent measure is connectivity, or whether each node is connected to each other node through some path. In a passageway network connectivity is essential, otherwise a segment of passageway and the spaces that it served would not be accessible from the rest of the ship.

After establishing a network, one can analyze the shortest or longest path characteristics between nodes. In an evacuation scenario, the shortest path between a berthing space and a muster station for lifeboats would be of utmost importance. For this research *Dijstrika's Algorithm* (*Dijkstra*, 1959) has been use for identifying shortest paths as it is the fastest known shortest path algorithm. Similar path length analyses have been done by *Nick* (2008), but using a geometric model rather than a network model.

The diameter of a graph is the length of the longest of the shortest paths between any pair of nodes. Diameter is a useful metric for quickly determining the size of a network and is one of the basic network complexity metrics. Continuing with the passageway system example, the diameter of an aircraft carrier's passageway network would be considerably larger than that of a Littoral Combat Ship.

Longest path analyses are vital to the analysis of the critical path problem in scheduling and have been used for assisting the set reduction process in the set based design paradigm (*McKenney*, 2013).

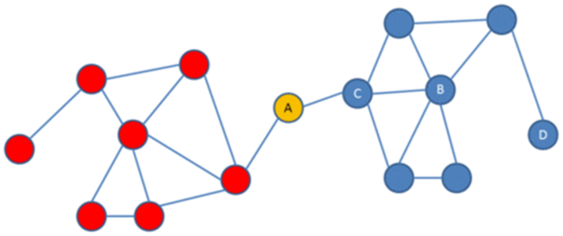


Figure 3.5: A betweenness centrality example. Nodes A and C have the highest betweenness while node D has zero betweenness (*Sellers*, 2011)

3.3.3 Centrality

Centrality is a way of addressing which are the most important nodes within a network. The simplest measure of centrality, mentioned previously, is the degree of a node. A more advanced concept of centrality is known as "betweenness centrality" or simply "betweenness." It is the measure of the extent to which a node lies in the paths between other nodes (*Freeman*, 1977). Betweenness is not concerned with the degree of the nodes. A node with a degree of two could easily have the highest betweenness in a network because it is the only connection between all the other nodes. More generally, it can be said that nodes with high betweenness have the most control over traffic across the network because the most amount of traffic must pass through them. They can be considered choke-points, and their removal from the network would cause the most disruption of traffic flow across a network. For example, ladderways located on the damage control deck, near centerline, amidships would have high betweenness because the most personnel would need to pass through that ladder during the course of ship operations. The betweenness, b_i, of node i is the ratio of the number of shortest paths, n^i_{st}, between node s and node t that cross node i to the total number of shortest paths, g_{st}, between s and t. (Equation 3.4). Figure 3.5 provides an example and discussion of the betweenness of a hypothetical network.

$$b_i = \sum_{st} \frac{n^i_{st}}{g_{st}} \tag{3.4}$$

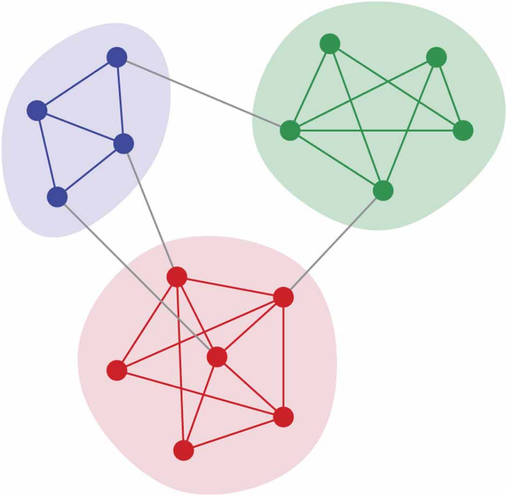


Figure 3.6: A network divided into three communities (*Newman*, 2012).

3.3.4 Communities

It is natural to try and breakdown networks into smaller groups or communities to see how the different groups interact, an example of which is shown in Figure 3.6. There are two ways of going about this: graph partitioning or community detection. Graph partitioning seeks to divide a network into a predetermined number of groups or into groups with a fixed size while minimizing the number of edges between the groups. Partitioning can be used to aid in creating a logical general arrangement (*Gillespie*, 2012).

Community detection, on the other hand, does not start with a predetermined number or size of groups. Instead a subgraph (a smaller part of the network) is considered a community if the intensity of the interactions among nodes in that subgraph is higher than what would be expected in a random graph (*Barigozzi et al.*, 2011). Communities can found via spectral modularity maximization based on the Louvain method (*Blondel et al.*, 2008; *Jutla et al.*, 2011). This method seeks to maximize a quality function (Equation 3.5 for simplex networks (*Lambiotte et al.*, 2008)) by first maximizing modularity in local communities, then creating a new network based on these communities. The optimization occurs in two passes. Each pass contains two phases. First, the modularity is maximized allowing only local changes of communities, then communities found in the first phase are aggregated into large communities to build a new network of communities. The passes are repeated until no increase of modularity is possible. The process is repeated until no increase in the overall quality function is possible. A visualization of this method is presented in Figure 3.7.

$$Q = \frac{1}{2m} \sum_{ij} \left(A_{ij} - \frac{k_i k_j}{2m} \right) \delta(c_i, c_j) \tag{3.5}$$

In Equation 3.5, Q is the quality of modularity, meaning it is a measure of the extent to which like is connected to like in a network, $2m$ is the number of ends of edges in the network, and k_i and k_j are the degree of nodes i and j, respectively. $\delta(c_i, c_j)$ is the Kronecker delta and c_i and c_j are the groups to which nodes i and j belong. If $c_i = c_j$ then $\delta(c_i, c_j) = 1$ otherwise $\delta(c_i, c_j) = 0$. Additionally, this quality function can be used for both unweighted and weighted networks (*Newman*, 2004).

There is no universally agreed upon definition of what it means to have a good division of a network into communities; therefore, many other community detection algorithms exists. An alternative method is based on the concept of edge betweenness (*Newman*, 2010). This method seeks to remove the edge with the highest betweenness, recalculate the betweenness of all the edges, and then repeat this process in an iterative manner until the network of interest has been divided into two/three/four/etc. communities. This method straddles the line between graph partitioning and community detection because it does not give just one division, but many, ranging for coarse to fine, and it is up to the user to decided at which level of granularity to stop the algorithm. The betweenness-based method is quite slow as the calculation time for edge betweenness takes on the order of $\mathcal{O}(mn(m+n))$, with m being number of edges and n being number of nodes.

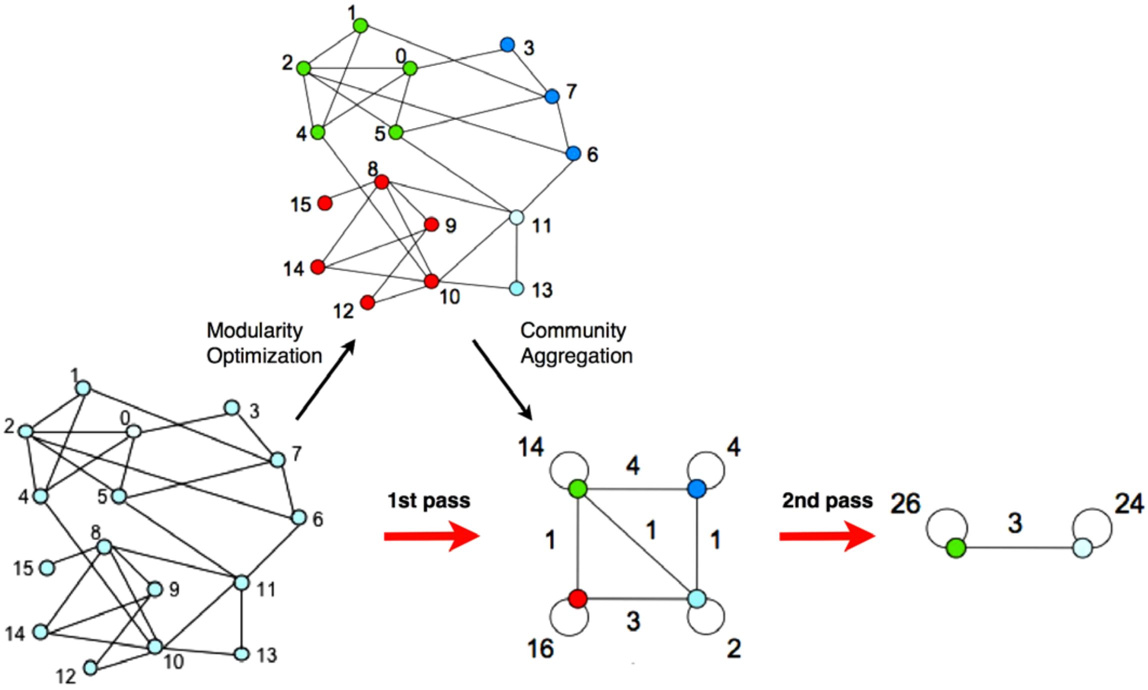


Figure 3.7: A visualization of the Louvain Method (*Blondel et al.*, 2008).

Even though it is slow, it would seem the betweenness-based method would have an advantage over the spectral modularity maximization method because it allows for control over community size based on pre-existing knowledge or intuition, but this can also be achieved using the spectral modularity method by adding a resolution factor, γ, as suggested by *Mucha et al.* (2010). The spectral modularity function then becomes Equation 3.6. γ can be varied to produce communities of a desired size or to see how the community structure of the network changes.

$$Q = \frac{1}{2m} \sum_{ij} \left(A_{ij} - \gamma \frac{k_i k_j}{2m} \right) \delta(c_i, c_j) \tag{3.6}$$

Another method of community detection introduced by *Radicchi et al.* (2004) is also based on the idea of removing edges between communities. *Radicchi et al.* (2004) found that edges connecting communities were unlikely to lay on short loops of other edges; if they were, then there would be other edges connecting two communities, therefore creating a single community. Thus, communities could be found by finding edges that belong to a high number of short loops. While this method is extremely fast ($\mathcal{O}(n^2)$ time) it is only useful for networks with a large amount of short loops, which typically restricts usage it to social networks.

One of the oldest methods, or more precisely class of methods, for community detection is called hierarchical clustering. To use this method, a metric for identifying strongly connected or closely similar nodes is chosen, then the closest or most similar nodes are joined together. These metrics can include, but are not limited to, cosine similarity, correlation coefficients, or Euclidean-distance measures. The fact that different measures can be used gives the method great flexibility in analyzing networks, but also is a hindrance because the answer is dependent on the measure. Choosing the right measure is therefore either a function of experience or experimentation.

Another method of community detection, "The Map Equation" (*Rosvall et al.*, 2009), will be discussed in Chapter IV. Many additional community detection methods have been proposed and a thorough review of these other methods can be found in *Fortunato* (2010) and *Schaeffer* (2007), but due to its simplicity, speed, and flexibility the spectral modularity maximization method for community detection will be used throughout the remainder of this work.

3.3.5 Planar networks

A planar network (or planar graph) is a network that can be drawn in a $\mathbb{R}^2$ plane, such that no two edges intersect (*Fleck*, 2013). Road networks, and presumably most ship system networks, are planar or near planar. Determining if a network is planar or not can be done manually by attempting to draw the network such that no edges overlap, but as the size of the network grows, this becomes no longer practical. For large networks, it is more efficient to search for non-planar subgraphs. *Kuratowski's theorem* states that every non-planar network contains at least one subgraph that is an expansion of the K_5 or $K_{3,3}$ subgraph (Figure 3.8) (*Thomassen*, 1981). These subgraphs are the smallest, non-planar graphs. They are impossible to draw in an two dimensional plane without having overlapping edges.

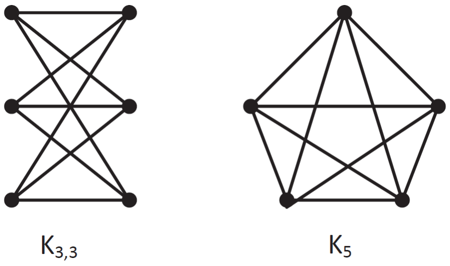


Figure 3.8: The non-planar $K_{3,3}$ and K_5 subgraphs

3.4 Interconnected and interdependent networks

The previously discussed network structures and metrics are useful for isolated networks, but in real systems, networks must interact. This leads to the study of interconnected and interdependent networks. In interconnected networks, the various networks complement one another, whereas in interdependent networks, one network must consume some resource supplied by another network (*Hu et al.*, 2011). Real world networks tend to be a combination of both, but ship system networks tend to be more interdependent than interconnected. For example, an HVAC system is dependent on the electrical system for power to run the fans while the electrical system is dependent on the HVAC system for cooling. Additionally, all distributed systems consume the space provided by the passageway or arrangement systems. An example of an interconnected system aboard a ship would be a LAN based communications systems complemented by sound-powered phones, but other examples of complementary systems are rare.

Broad lessons about the way to structure the interconnectedness and interdependency of ship systems can be taken from the concepts of interconnected and interdependent networks. Increasing the interconnectedness of multiple networks (for example by increasing avenues of communication) has been shown to increase the overall system robustness (*Leicht and D'Souza*, 2009), whereas increasing the interdependence of multiple networks lowers the overall system robustness (*Parshani et al.*, 2010). Additionally, interdependent networks with a broad degree distribution have been demonstrated to be less robust than interdependent networks with a narrow degree distribution, which is the opposite of what occurs in isolated networks (*Gao et al.*, 2011). This occurs since the increase in the spread of degree distribution makes a network more likely to have a hub that relies on a poorly connected node in another network. If a poorly connected node in Network A that is connected to a hub in Network B is removed, then the hub is also removed. This causes failure in the nodes in Network B connected only to the hub, which in turn, could cause more hubs in the Network A to fail. This process has the potential to repeat, causing cascading failures throughout both networks.

A prime example of such a cascade of failures between networks is the September 2003 national blackout in Italy, where the interdependencies between the power network and the Internet network caused a widespread electrical blackout (*Buldyrev et al.*, 2010). The electrical stations relied on control provided by the Internet network and the control stations relied on power supplied by the electrical network, so when one failed the other failed, causing a nationwide blackout. When designing ship systems such a potential for cascading failures should be noted and redundancies and fail-safes should be built into the systems such that the systems can be run independent of each other.

Interconnected and interdependent network analyses tend to be conducted after an event happens, as it is often difficult to manually identify interactions between networks (*Rinaldi et al.*, 2001). This framework is therefore of limited use in the design generation stage. What is needed is a method for determining unforeseen interactions between different networks. This is possible via the multiplex network framework which will be discussed in the next section.

3.5 Multiplex and multislice networks

The networks discussed thus far have all been simplex networks, where each node represents one type of entity and each edge represents a specific type of connection. These networks also can be combined together into what is called a "multislice" or "multiplex network." Typically, multislice networks represent time-dependent changes where each slice of a network is connected only to the slices directly proceeding and following it. Multiplex networks are used to represent networks where edges represent different connections between the same node. Figure 3.9 shows a visual representation of both a multiplex and multislice network. The multiplex structure is more akin to a ship design with its systems represented as networks. Through the previously discussed concept of community detection, unforeseen interactions between systems can be discovered using this multiplex structure.

Community detection in multiplex networks uses the same Louvain method as community detection in simplex networks, but with a different quality function (Equation 3.7).

$$Q = \frac{1}{2\mu} \sum_{ijsr} \left\{ \left(A_{ijs} - \gamma_s \frac{k_{is} k_{js}}{2m_s} \right) \delta_{sr} + \delta_{ij} C_{jsr} \right\} \delta(g_{is}, g_{jr}) \tag{3.7}$$

The multiplex quality function is an expanded form of the simplex function where

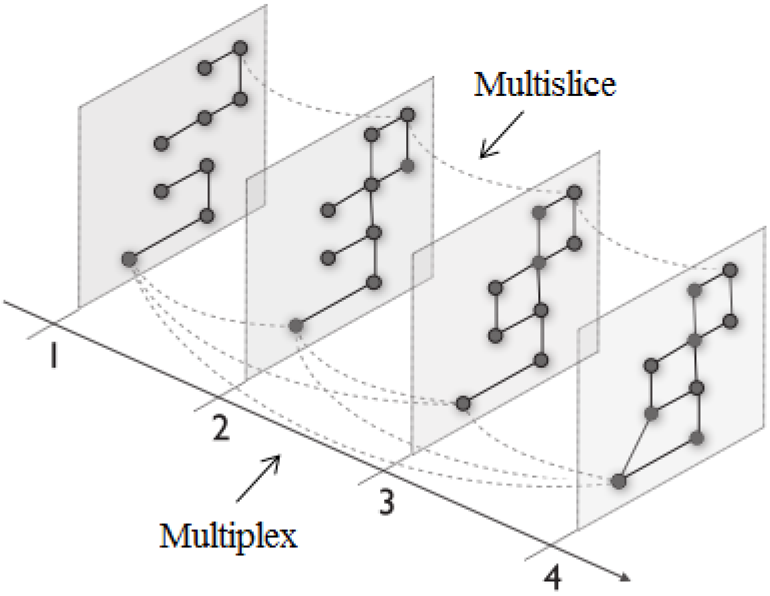


Figure 3.9: The multislice and multiplex network structure. (*Mucha et al.*, 2010).

A_{ijs} is the adjacency matrix for each slice s showing the connections between nodes i and j. C_{jrs} is the interslice coupling matrix that has the connections between node j in slice r and slice s. $C_{jrs} = 0$ if there is no interslice link or $C_{jrs} = \omega$ if there is an interslice link. In a multiplex network, each node is connected to itself in all slices so $C_{jrs} = \omega$ at all times in this analysis. 2μ is the number of ends of edges in both the adjacency matrix $\mathbf{A}$ and interslice coupling matrix $\mathbf{C}$ combined. γ_s is the resolution factor in each slice. k_{is} and k_{js} represent the degree of nodes i and j in slice s and $2m_s$ is the number of ends of edges in slice s. δ is once again the Kronecker delta. $\delta_{sr} = 0$ if the quality function is not comparing the same network slice, $\delta_{ij} = 0$ if the quality function is not comparing the same node across network slices, and $\delta(g_{is}, g_{jr}) = 0$ if the group for node i in slice s is not the same as the group for node j in slice r. Q is still the measure of the extent to which like is connected to like, except now the connections extend across multiple dimensions.

This quality function has been used by *Mucha et al.* (2010) to determine social communities in a multiplex network of college students' interactions. The four interaction networks were Facebook friendships, picture friendships, roommates, and housing group (*Lewis et al.*, 2008). By varying the value of ω, 1640 nodes were grouped into between one and four distinct communities. When $\omega = 0$, each node was placed in four different communities, one for each of the four interaction networks. As ω was increased the number of communities per in node was reduced, and when $\omega = 1$ each node was in one community that spanned all four interaction networks. A similar strategy will be used in this dissertation to identify interactions between ship systems.

In the case of a multislice network, the community detection quality function would be the same, but the value of C_{jrs} would be 0 except, when $r = s \pm 1$ to account for the sequential nature of multislice networks. A multislice analysis of U.S. congressional voting trends was completed in *Mucha et al.* (2010).

3.6 Network theory conclusions

This section has introduced many classical network structures and methods (but is by no meanings comprehensive, see *Newman* (2010) for a much fuller discussion) and has connected them with naval architecture ideas and concepts via analogy, with no discussion of their valid or applicability. The following chapters will build on the network concepts introduced in the preceding pages and not only demonstrate their applicability to naval design but develop new, novel techniques and methods to further extend network theory into the naval distributed systems design domain.

CHAPTER IV

A New Network Complexity Metric

Identification of possible complicated and complex issues in the early design stage is critical in preventing these issues from spiraling and causing a design to not converge. Additionally, when selecting between competing design alternatives that offer the same functionality, the less complex alternative should always be selected (*Knight*, 2014). A survey of complexity methods and metrics based on the physical properties of ships and their structures was presented in Chapter 2, but in order to analyze distributed systems designs using network principles a network-based complexity method is necessary. It was found there was a lack of such a metric for judging network complexity, especially for the planar or near planar networks that are frequently found in ship distributed systems design. To rectify this situation, a metric based on the network concepts of planarity and community detection created for this research and published in *Rigterink and Singer* (2014a). This chapter serves to introduce and validate this metric.

4.1 Network complexity metrics

When dealing with large, complex systems, engineers have sought ways to judge how complex the systems are; both to better understand a given system and to compare it to other complex systems. Sometimes complexity is a good thing. For example, from the complexity of an ecosystem comes its extraordinary robustness, from the complexity of the World Wide Web comes the largest and most powerful sharing of human knowledge and interaction. On the other hand, especially in physical systems, complexity breeds errors. An electrical circuit with wires tangled is more likely to short out than one with well-defined paths and simple connections. A complex manufacturing process is more likely to experience delays and product errors over one that follows a simple, streamlined process. Even in non-physical systems like software, a program with many smaller sub-functions each requiring data from the other ones is more difficult to both initially compile as well as debug than a program that follows a series of logical steps, with independent, modular subroutines. In addition to measuring complexity, making complexity legible so it can be studied and understood is paramount to a designer or engineer.

Many complexity metrics and analogues for complexity have been devised, including topological indices such as the Randić index (*Randić*, 1975; *Liu et al.*, 2005), which seek to determine the complexity of the underlying structure of the network (*Sivakumar and Dehmer*, 2012). Topological indices can be used to give a non-empirical description of the structure of a network which allows for a quantitative analysis of the structure (*Kier and Hall*, 2002). Such indices can then be related to other properties of the entity which the network represents, like the boiling point of substances based on the structure of their molecules (*Randić*, 1975). A further list of such indices and metrics can be found in *Todeschini and Consonni* (2009).

Furthermore, the concept of network dimension can be used to determine the underlying structure and function of a network (*Shanker*, 2013). Dimension is especially useful for considering real networks that are naturally embedded in two- or three-dimensional space but would not be assumed to be planar, like an airline connection network or Internet connection network (*Daqing et al.*, 2011). In cases like these, a network with higher dimension could be said to be more complex than one with lower dimension.

Network complexity also can be judged using an information theory perspective. Simple methods like degree distribution or network diameter can be used to quickly gain insight to the overall structure of a network. More complex metrics like average information on the degree or distance distribution can be used to future elucidate the structure of smaller portions of networks (*Bonchev and Trinajstic*, 1977). Additionally, measures based on centrality also can be used determine the ease or difficulty of information flow across a network (*Bonchev*, 2009). Further information on network complexity measures can be found in *Dehmer and Emmert-Streib* (2009); *Dehmer* (2011); *Dehmer et al.* (2013)

The metrics and measures just mentioned have been shown to be useful for many kinds of networks, but in certain cases they become less adequate because the network being studied is sufficiently different than networks for which the metric was designed. Specifically, in real systems such as road networks or electrical grids where degree distribution is approximately exponential, the centrality of all nodes is roughly equal, and the network is embedded in a two-dimensional space, classical complexity metrics can no longer differentiate between the complexity of two networks.

To rectify this situation, a simple metric for measuring the complexity of these kinds of large networks based on the concepts of graph planarity and network community detection was created. For this metric, a network will initially be determined to be planar or not. If the network is not planar, it will then be run through a community detection algorithm to divide the network into a series of subgraphs. These subgraphs will then be checked for planarity. A network that has more planar communities is considered less complex. The algorithm for calculating this metric runs in $\mathcal{O}(n)$ time where n is the number of communities into which the network is divided.

Work on this metric was also motivated by a quote from *Newman* (2010):

> *What we would really like is some measure of the degree of planarity of a network...If such a measure were to gain currency it might well find occasional use in the study of real world networks.*

Newman suggests such a measure could be based on the number of expansions of the K_5 or $K_{3,3}$ subgraphs (Figure 3.8) contained within a network, which is the direction taken for this metric.

Newman also claims that a planarity metric would be used solely for physical or real world networks, but it will be shown this metric can be used to measure the complexity of non-physical networks. While it is true that non-physical networks are less likely to be planar, that does not mean that nothing can be gained from studying the degree of planarity, especially when it comes to quantifying a networks complexity or where extremely interconnected regions exist.

The belief that planarity and complexity are related is not new and was introduced in *Henry et al.* (1981) where it was claimed that a network or system is more understandable by the user if it is possible to map the flow of information in such a way that no two links overlap. This concept was further expanded in *Kortler et al.* (2009) where a metric for network complexity was developed based on the removing of edges to create the maximum planarization of a network. The edge removal algorithm is computationally intensive ($\mathcal{O}(n! * n)$ time), whereas the new community-based planarity metric runs in linear time, dependent on the number of communities found in the example network.

4.2 The planarity-based complexity metric

For this disseration, a new network complexity metric was created, using the network concepts of community detection and planarity. Figure 4.1 shows a graphical representation of the process required for scoring a networks planarity using this metric. The initial check for the metric is to see if the complete network is planar or not via the Boyer-Myrvold Algorithm (*Boyer and Myrvold*, 1999, 2004). If the network is planar then the metric quits and returns a planarity-complexity value of 0, meaning the network is totally planar. If the network has some non-planar element then the routine continues.

The next step is to determine the network communities. For this work, the spectral modularity maximization method, described in Section 4.3.4, is used. Any other community detection or graph partitioning method could be used, but spectral modularity maximization was used for this work due to its speed as well as its likelihood to return non-planar communities because of its focus on node interdependence. Also, because the size of communities can be tuned using the resolution parameter, γ, modularity maximization allows the user of the algorithm to set community size or number according to his or her beliefs and/or intuitive understanding of the studied network. A comparison of the results using spectral modularity maximization to those found using the map equation (*Rosvall and Bergstrom*, 2008; *Rosvall et al.*, 2009) is presented later in this chapter.

Following community detection, each community is tested for planarity, again using the Boyer-Myrvold algorithm. Following this, the number of non-planar communities are compared to the total number of communities via Equation 4.1 where C is the total number of communities and K is the number of communities that contain a non-planar subgraph. The possible values for this metric range from 0 to 1, inclusive,

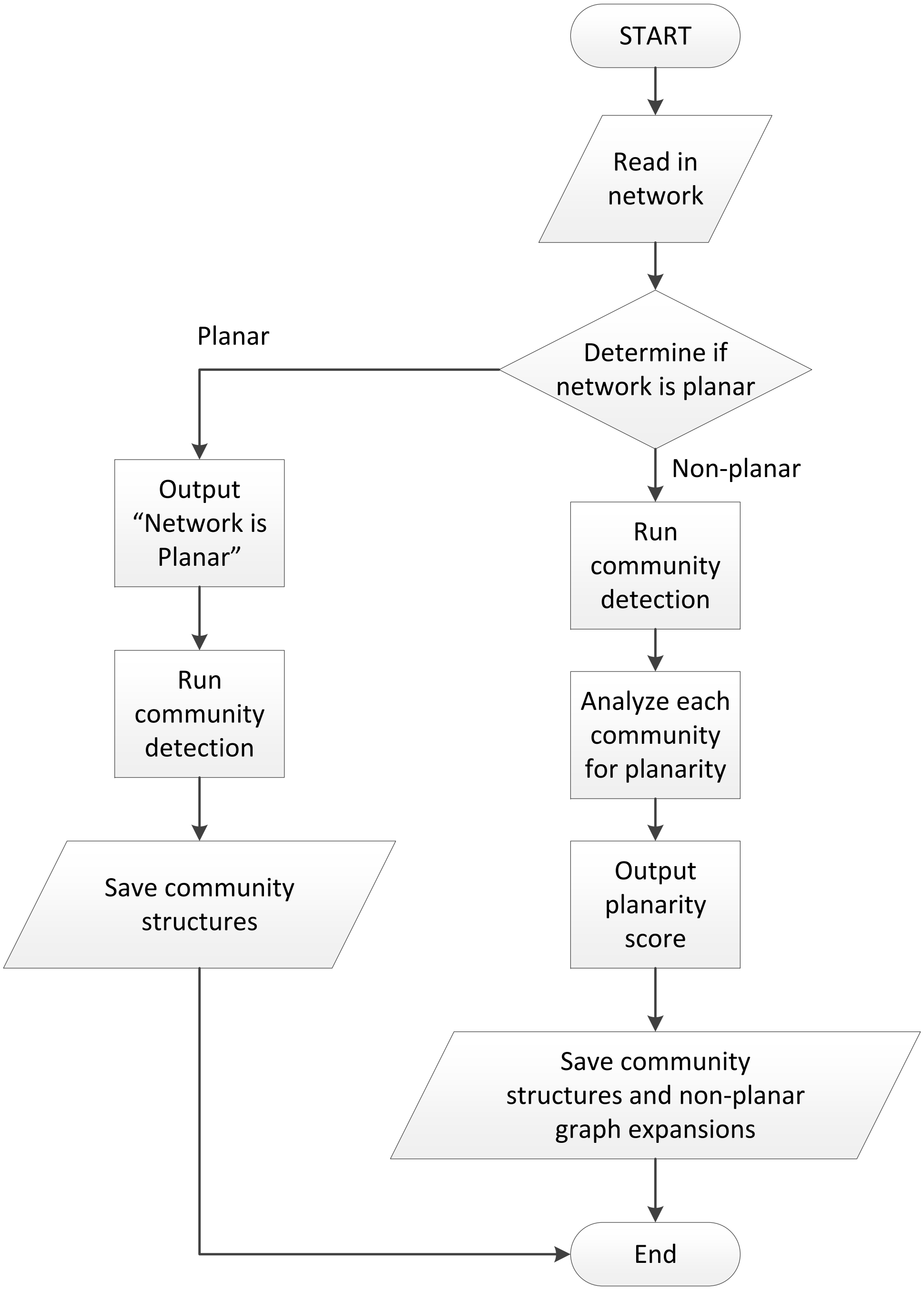


Figure 4.1: Algorithm for computing the planarity-based complexity metric.

with 0 being planar and therefore least complex and 1 meaning each community is non-planar, *i.e.*, the network has maximum complexity.

$$P = \frac{K}{C} \tag{4.1}$$

In addition to calculating the planarity-complexity score of the individual network, the score can be compared to that of a random network that has the same density as the original network using Equation 4.2. Using this ratio may be more meaningful when comparing different networks in certain situations; for example, when the two networks being compared vary greatly in structure. Additionally, this ratio reduces the impact of the resolution factor upon the complexity score because the complexity scores of both networks are dependent on the same factor.

$$P_{ratio} = \frac{P}{P_{rand}} \tag{4.2}$$

If $P_{ratio} < 1$ then the studied network is less complex than what would be expected at random; whereas if $P_{ratio} > 1$ then the studied network is more complex than what would be expected at random.

The network to be analyzed can be either weighted or unweighted. Typically, when considering planarity, directed networks are converted to undirected networks in order to show all possible links by simply adding the transpose of the the adjacency matrix to itself, thus, direction is ignored (*Girvan and Newman*, 2002). Another possibility is to create two separate networks, one for inward pointing edges and one for outward pointing edges and then run the metric on the respective symmetric projections of those networks. Additionally, if it were desired to study the network in its completely directed form, a community detection technique utilizing teleportation could be used

(*McKenzie*, 2012; *Lambiotte and Rosvall*, 2012).

Three network structures will be analyzed using this method in this chapter. Two of the networks are representations of real systems: the road network of the State of Minnesota and the electrical power grid for the Western United States. The third network is a representation of the interactions between variables in a standard set of ship design equations. The three networks range in size, structure, and purpose and were chosen to show the versatility of the community-based planarity-complexity metric. A fourth use of the planarity-complexity metric will be shown in Chapter 7 as part of a ship design example.

4.3 State of Minnesota road system

The network representation of the State of Minnesota's road system is an undirected, unweighted network comprised of 2,642 nodes and 6,606 edges. The network is hypothesized to be highly planar as the instances of roads crossing over each other without intersecting should be rare. An analysis of the planarity of the this network has been undertaken in *Gleich* (2008), with an emphasis on finding those nodes which cause the non-planarity.

The goal of using the planarity-complexity metric is to identify communities of the network where non-planarity exists to allow for further study of those areas to determine if there is an error in the creation of the network, or if there is a truly complex section of road present. In an effort to minimize the size of communities the resolution factor, γ, was increased by 0.5 until the quality of modularity fell below 0.85. This threshold was chosen to show proof of concept and has no other relation to the network. Once this threshold was reached the algorithm continued.

The Minnesota road system network was broken down in to 78 communities with a quality of 0.84, only one of which was non-planar for a complexity score of 0.013. This means the network is nearly planar and therefore not very complex, which is what is desired in a road system. Figure 4.2 shows the total network with the non-planar community highlighted. The non-planar community is located on the north side of the city of Minneapolis, near where Interstates 94 and 694 intersect, adjacent fo the Mississippi River. Roads cross over or under other ones in this area without intersecting, causing the non-planarity, rather than an error in the creation of the network.

Comparing the road system network to a random network with the same density, $P_{rand=}0.031$ and $P_{ratio} = 0.42$, the road network is less complex than what is expected at random.

4.4 Western United States power grid

The final physical system to be analyzed in this paper is the Western United States power grid (*Watts and Strogatz*, 1998). The complexities of the power grid have long been known and have been responsible for large scale blackouts across the world (*Andersson et al.*, 2005), therefore this network is assumed to be highly non-planar. The goal of analyzing this type of network is to find the highly complex communities so those managing the power system know where cascading problems are most likely to happen. Cascading failures are more likely in these highly interconnected communities because there are many different paths for a fault to propagate through.

The network representation of this system is also an undirected, unweighted network with 4,941 nodes and 13,188 edges. For this test case, the community detection algorithm was run with values of γ ranging from 1 to 7. The upper bound on gamma

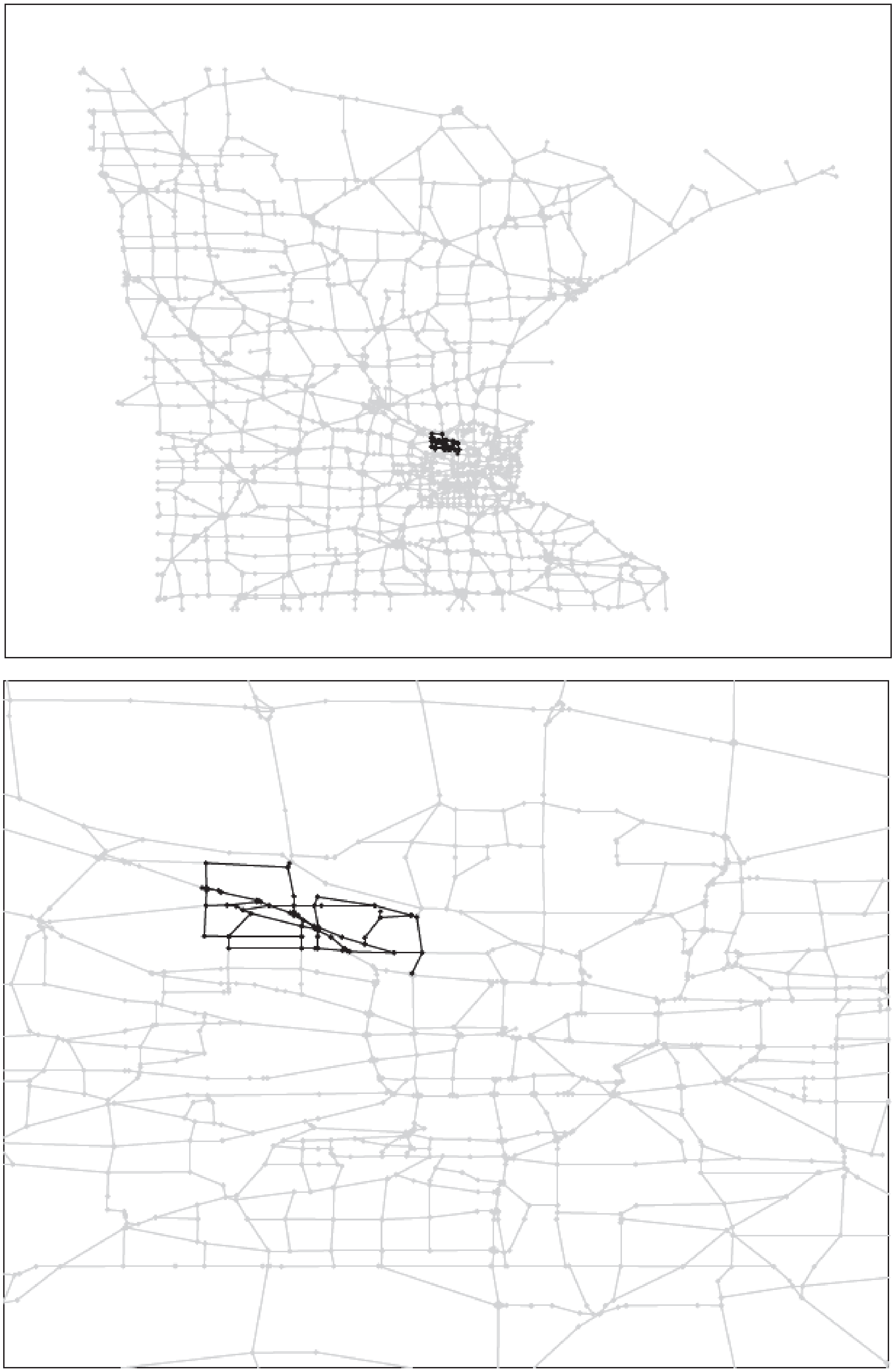


Figure 4.2: The road system of the State of Minnesota. The non-planar community is highlighted (Network files from *Gleich* (2008))

Table 4.1: Statistics for planarity-complexity of the Western United States Power Grid network using different resolution factors.

γ	Q	C	$\frac{C}{C_{\gamma=1}}$	P	Relative Time	P_{rand}	P_{ratio}
1	0.94	41	1.00	0.56	1.00	0.047	11.91
2	0.91	59	1.43	0.42	1.67	0.012	35.00
3	0.90	80	1.95	0.30	2.24	0.002	150.00
4	0.88	96	2.34	0.21	2.51	0.000	undefined
5	0.87	111	2.70	0.19	2.99	0.000	undefined
6	0.86	124	3.02	0.15	3.13	0.000	undefined
7	0.85	137	3.34	0.14	3.27	0.000	undefined

was when the quality of modularity fell below threshold of 0.85. This was done to show how the relative complexity of a network can change based on the parameters it is compared against. Additionally, this can be used to show the relative speed of the algorithm. Table 4.1 displays the resolution factor, number of communities, normalized number of communities, planarity-complexity score, and normalized algorithm run times. The number of communities and algorithm run times have been normalized to case of $\gamma = 1$.

It can be seen that the the algorithm runs in $\mathcal{O}(n)$ (linear) time dependent on the number of communities created from the initial network. This is expected as the algorithm is based on the Boyer-Myrvold planarity algorithm, which also runs in $\mathcal{O}(n)$ time, and the planarity-complexity algorithm simply calls the Boyer-Myrvold algorithm C+1 times (one initial call to check if the starting network is planar). The planarity-complexity score of the network decreases as the number of communities increases which also was expected. As the communities get smaller, there is less of a chance of there being non-planar graph expansions present. Also of note, P_{rand} for all cases was less than P for each gamma case, meaning the actual networks are more complex than would be expected at random. In a system with numerous of localized structures, such as a power grid, this is expected.

4.5 The Watson and Gilfillan ship design equations

Lastly, this metric will be tested on a network representation of a classic set of naval architecture design method (*Watson and Gilfillan*, 1977; *Parker and Singer*, 2013) to prove a planarity metric can be useful for analyzing more than just physical systems. In this network, the design variables are nodes, and an edge is created if two variables are used in the same equation. These equations also have been analyzed using a tripartite network structure and various centrality metrics to determine the most important variables (*Parker and Singer*, 2013). The 28 nodes in this network were separated into three communities, none of which were planar ($P = 1$). It was found that there were complex, non-planar interactions between:

- Length, beam (vessel width), depth (total vessel height), draft (height of vessel below the water), and displacement (vessel weight)
- Coefficient of thrust (propeller power output), effective power (the power necessary to move the ship through the water), engine maximum continuous rating (the maximum amount of power an engine can produce continuously in normal conditions), machinery weight, and machinery cost
- Deckhouse height, ship type, overall steel weight, ship type based steel weight, and total ship cost

These complex, highly dependent interactions are intuitive to a naval architect and agree with the analysis presented in *Parker and Singer* (2013).

For a random network with the same density, $P_{rand} = 0.75$, meaning this equation network is more complex than what is to be expected at random. This is expected, again pointing out the highly dependent and interrelated nature of the ship design equations.

In general, the ratio of the planarity-complexity score for a real network to that of a randomly generated network is not useful for the analysis of individual real networks. This is because most real networks are sparse, or sparse enough, that complex, non-planar communities are rare in random networks with the same density as the real network. The complexity ratio of real networks to random networks is more interesting when real networks are considered holistically, as a class of networks. The fact that P_{ratio} is typically greater than one proves the assertions made by *Miller and Page* (2007),*Page* (2010), and *Rickles et al.* (2007) that real systems are, in general, more complex than their random counterparts.

4.6 Validation of metric

Weyuker (1988) stated nine criteria for judging the effectiveness of a complexity metric for software systems . These have been modified slightly to better apply in the context of this work, in a style similar to *Cardoso* (2005); *Kortler et al.* (2009). The criteria are described in the following list and arguments as to how well the developed metric meets each criterion are presented. Since these criteria have been adapted from criteria used for measuring the complexity of scripts in computer programs some of the criteria seem either inconsequential or were not obtainable. Each criterion also includes a discussion of how it relates to the naval domain.

1. **Each different network may be complex.** This is a fundamental assumption in the study of complex systems and networks. The purpose of the developed metric is not to determine if a network is complex, but rather the degree of its complexity.

2. **A metric cannot measure all networks as being equally complex.** As evident from the case studies, the complexity of different networks varies based

on both the community structure as well as the degree of planarity.

3. **There are only a finite number of networks of the same complexity.** To meet this criterion, a metric must be sufficiently sensitive so as to not group all networks into a small number of complexity classes. While there is an infinite amount of possible networks if the number of nodes and edges are allowed to vary, there is only a finite number of networks that can be created for a set number of nodes and edges, so at worse for a given network size this finite amount of permutations would have the same complexity.

4. **The complexity of a network depends on its implementation; even if two networks solve the same problem, they can have different complexities.** This criterion speaks to the actual system that is being modeled by a network. It is not hard to imagine two potential computer programs, power grids, or roadway systems that serve the same purpose and solve the same problem but have vastly different complexities when abstracted to a network. Additionally, it is possible to represent the same system differently in network space and these representations could also have different complexities.

5. **If a network of a given complexity is created by joining two networks, the complexity of the resulting network is a function of the interactions between smaller networks.** If two networks were joined together, the added edges could create new, non-planar communities causing the higher joint complexity. But on the other hand, the networks could be simply analyzed together, but with no connection between the two. As explained in Property 9, such an analysis would result in a lower complexity.

6. **A permuted version of a network can have different complexity.** If the edges of a network were permuted to give a different structure, but the

global characteristics of the network (degree distribution, diameter, etc.) were unchanged, the complexity of the network could change.

7. **If a network is a straight renaming of another network, its complexity should be the same as the original network.** As explained in the previous point, a network with nodes renamed or reordered will not have its fundamental structure changed, and therefore a renaming of a network produces the same complexity as the original.

8. **The complexity of two networks joined together may be greater than the sum of their individual complexities**. Meeting this criterion is subject to what edges are added connecting the original networks. In the event that two networks are analyzed together without adding any edges between them, the joint complexity will not be higher than the sum. But in the case that two completely planar networks were joined together with nodes in such a way that a non-planar subgraph was created, then the complexity of the resulting network would be higher than the sum of the planarity-complexity score of the original networks.

9. **The complexity of two networks joined together is greater than the complexity of either network considered separately.** This criterion is not met for the case of simply analyzing two networks together without adding edges between them. The process for calculating the planarity-complexity score in such an instances is presented in Equation 4.3.

$$P_{A+B} = \frac{K_A + K_B}{C_A + C_B} \tag{4.3}$$

This means a network with complexity $P_A = \frac{1}{10} = 0.1$ combined with a network

> with complexity $P_B = \frac{9}{10} = 0.9$ has a complexity of $P_{A+B} = \frac{1+9}{20} = 0.5$ which is a lower complexity than Network B.
>
> If the two networks were joined together with additional edges it would be possible to increase the complexity score of the new network because new communities would be formed and a greater percentage of those could be non-planar than in either of the original networks, but as this is not assured to happen it cannot be claimed the metric meets this final criterion. This presents a possible extension to this metric looking at the density of non-planar subgraphs in specific partitions of the overall network. This could potentially be done by running the community detection algorithm recursively. Due to the low computational expense of this algorithm, a recursive analysis would be practical.

By meeting eight of these nine criteria, it is believed this planarity-based metric is useful for the study of networks that represent both real and abstract systems.

In addition to the criteria present by Weyuker, it is of interest how this new metric compares to other similar metrics. To this end, the complexity scores calculated for the test cases will be compared to their scores on the Randić connectivity index (*Randić*, 1975; *Liu et al.*, 2005). While the Randić index is typically used in the study of molecular bonds, it can be used for any network to give a non-empirical description of the structure to allow for a quantitative analysis of the structure (*Kier and Hall*, 2002). For this work, the first degree Randić index (*Todeschini and Consonni*, 2009), χ^1, will be calculated using Equation 4.4 where A is the total number of nodes, a_{ij} represents the existence of an edge between node i and node j, and δ_i and δ_j are the degrees of node i and node j, respectively. Due to the fact that the calculation of the Randić index includes a summation, it will invariably be larger for networks with more nodes. To counteract this, the calculated Randić index for each network must be normalized by the number of nodes in the network. This normalized value, χ^1_{norm}, will allow for easier comparison across networks.

$$\chi^1 = \sum_{i=1}^{A-1} \sum_{j=i+1}^{A} a_{ij}(\delta_i * \delta_j)^{-\frac{1}{2}} \tag{4.4}$$

The first four networks presented in Table 4.2 are provided to give points of reference for how simple planar and non-planar networks score on the Randić index. The first example network is a simple "circle network" with six nodes and six edges, the next, more complex example is a 2-methylpentane molecule (Figure 4.3) (*Todeschini and Consonni*, 2009), then the final two example networks are the $K_{3,3}$ and K_5 non-planar networks presented earlier in Figure 3.8. The Randić index value for each graph was calculated using the algorithm presented in *Iranmanesh and Alizadeh* (2010). These cases display that the greater the Randić index, the simpler a network appears. It also is interesting to note that the K_5 network has a lower Randić index than the $K_{3,3}$ network, and therefore could be considered more complex. This demonstrates the Randić index's bias towards considering denser networks to be more complex.

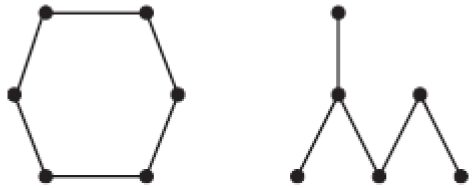


Figure 4.3: Circle (left) and 2-methylpentane (right) networks

Table 4.2: A comparison of the planarity-complexity score and Randić connectivity index for various networks.

Network	P	χ^1	χ^1_{norm}
Circle Network	–	3.41	0.57
2-methylpentane molecule	–	2.77	0.46
$K_{3,3}$	–	2.00	0.33
K_5	–	1.20	0.20
Road System	0.013	1,286	0.49
Power Grid ($\gamma = 1$)	0.560	2,333	0.45
Design Equations	1	12.64	0.45

The normalized Randić indices for the four test cases fall between that of the simple circle network and those of the K_5 and $K_{3,3}$ subgraphs which shows that such an index can be used as a measure of complexity of interactions, which is expected, as all topological indices are meant to quickly give some idea of the structure of a network in a quantitative manner. The issue with using a topological index for the type of networks studied in this work is that their structures are all very similar to each other. Therefore the normalized Randić indices are all very close to each other (a spread of just 0.04) which violates, or nearly violates, the second criterion presented by Weyuker (a metric cannot measure all networks as being equally complex). The unnormalized Randić index values are almost meaningless for comparison due to the differing size of the networks. For this reason, it is believed the presented planarity-complexity metric is superior to a topological index for studying and analyzing real networks.

As noted previously, the implementation of the algorithm used for calculating the complexity metric in this work utilized modularity maximization as a means for determining communities. The main criticism of this technique is that it is resolution dependent and relies on the choice of gamma. Though it is believed this can be beneficial as it allows for tuning of the analysis dependent on the size of communities desired, investigating the robustness of the metric using a different community detection scheme is necessary. Therefore, the test cases were reran using the map equation (*Rosvall and Bergstrom*, 2008) to determine the network communities. The map equation for undirected networks, weighted networks is presented in Equation 4.5. To use the map equation for community detection, the description length of the partitions, $L(M)$, is minimized by modifying the community membership of the nodes. The description length of a network with n nodes divided via m partitions is based on the relative weight, w_α which is the sum of the weights of all the edges connected to each node α, divided by twice the total weight of all edges in the network.

The relative weight of each module i is $w_i = \sum_{\alpha \in i} w_\alpha$. $w_{i\curvearrowright}$ is the relative weight of links exiting module i, and $w_{\curvearrowright} = \sum_{i=1}^{m} w_{i\curvearrowright}$ is the total relative weight of all edges between modules. The symbol $\curvearrowright$ denotes terms relating to links between modules, and this nomenclature is taken directly from *Rosvall and Bergstrom* (2008). The minimization algorithm attempts to reduced the relative weight of the edges between modules while increasing the relative weight of each module.

$$\begin{aligned} L(M) = w_{\curvearrowright} \log w_{\curvearrowright} &+ \sum_{i=1}^{m} (w_{i\curvearrowright} + w_i) \log (w_{i\curvearrowright} + w_i) \\ &- 2 \sum_{i=1}^{m} w_{i\curvearrowright} \log w_{i\curvearrowright} - \sum_{\alpha=1}^{n} w_\alpha \log w_\alpha \end{aligned} \tag{4.5}$$

The map equation differs from modularity maximization because it determines community membership from pairwise connections and the way the network was formed as opposed to focusing on the interdependence of edges and the dynamics of the already formed network. By considering a random walker as real flow across a network, the map equation is able to capture the minimum entropy necessary to describe the trajectory of said random walker for a given network partition. This value can then be minimized over all possible partitions to provide a set of network communities (*Rosvall et al.*, 2009). Table 4.3 presents the comparison of the planarity-based complexity scores for the four test cases using modularity maximization and the map equation.

Table 4.3: A comparison of the planarity-complexity score using modularity maximization and The Map Equation.

Network	$C_{modularity}$	$P_{modularity}$	C_{map}	P_{map}
Road System	78	0.013	246	0
Power Grid ($\gamma = 1$)	41	0.560	437	0.023
Design Equations	3	1	1	1

It can be seen that, for the test cases with the larger networks, the map equation partitions the networks into significantly more communities. More communities means there are less nodes per community (an average of just over 10 nodes per community for the road system network and roughly 11 nodes per community for the power system network) meaning a particular community is less likely to be non-planar, similar to the trends shown in Table 4.1. For the smaller test cases, the map equation returns similar results to those found via modularity maximization, though for the design equations case (the smallest of the four), the map equation does not partition the network at all meaning very little information about the network structure or interactions is gained.

Due to its focus on flow and pairwise connections, the map equation is less likely to create non-planar communities, especially those with the expansion of the $K_{3,3}$ subgraph, because such a subgraph naturally has a higher level of entropy than a circular subgraph like that shown in Figure 4.3. This behavior was shown when using the map equation to partition the road system network and the power grid network. For this reason, it is believed the map equation is not suitable as the community detection method for this algorithm, and modularity maximization (or a similar method) is preferred.

4.7 Planarity-complexity metric conclusions

This chapter introduced a new network-complexity metric based on the network concepts of community detection and planarity. This metric was created to fill a void in network metrics for determining the complexity of the planar and near planar systems. This gap existed because planar networks typically do not receive much attention because the planarity criteria is not easily overcome using calculus. The lack of analytical tools and results discouraged the study of planar networks and the lack of study of planar networks led to a lack of analytical tools (*Rosvall et al.*, 2009). The new metric in this dissertation contributes a non-calculus-based metric to encourage the expanded study of planar networks.

Additionally, planar networks have not received interest because many believe they are trivial in both topological and geometrical structure (*Rosvall et al.*, 2009) and metrics created for their study would only find "occasional use" (*Newman*, 2010). This chapter showed that planar or near planar networks are quite prevalent, in both technological/real networks and information/equation networks and an analysis of their planarity can lead to useful insights into their structure and complexity.

Due to its speed, versatility, and effectiveness this novel planarity-complexity metric not only provides network researchers a new tool for quantifying the complexity of networks, but also encourages a renewed interest into the study of planar and near-planar networks.

From a naval architect's perspective, this new complexity method is useful for not only scoring system complexity, but also identifying where this complexity comes from. Knowing which parts of a system contribute the most to system complexity would allow the naval architect to manage risk by allocating more time to those parts of the design or incorporating larger margins into those parts of the design. It is impossible to remove complexity from naval design, but this new method provides naval architects a new way in which to make that complexity legible.

CHAPTER V

Ship Distributed Systems Design Analysis Using Network Theory

This chapter serves to the validate the network concepts presented in the previous chapters for use in the naval architecture domain. The chapter begins with a analysis of individual systems by comparison of a modular ship systems design philosophy with a more traditional design philosophy. The focus then moves onto multislice networks and the effect changes throughout time have on individual systems. The chapter closes with a discussion of a new way of analyzing system interactions in both a static case and throughout time.

The structure of this chapter mimics that of the ship design process and demonstrates how network structures and metrics are applicable to each stage of this process. First, a prototype system is designed independent of other systems. This system can be analyzed a simplex network using basic metrics like connectivity and degree distribution. Next, the prototype is redesigned and improved. The evolution from iteration to iteration is analyzed and evaluated using multislice community detection. Once the individual system designs have converged their interactions must be analyzed, which is possible using multiplex community detection. Lastly, the evolution of system interactions in both design and operation must be tested and understood, which is possible using a new time-dependent, multiplex structure and quality of modularity function for community detection which were both developed for this dissertation, and are introduced in this chapter.

5.1 Analysis of individual distributed systems

In this section, an analysis using the previously described network structures and metrics will be carried out to compare the potential systems survivability of a traditionally designed ship to that of a modularly designed ship, typified by the MEKO (Mehrzweck-Kombination or multi-purpose-combination) concept developed by the Blohm and Voss shipyard. Following the introduction of the MEKO concept, a series of distributed systems designed using both the MEKO paradigm and a traditional paradigm will be represented via networks and the previously introduced network concepts will be used to evaluate said networks.

5.1.1 The MEKO Concept

The MEKO design concept is based around three key elements; modularity, survivability, and reduction of the ship's radar signature (*Blohm and Voss*, 2003). Only the first two elements will be discussed in this work. The ships using this system are specifically designed to increase the flexibility of installation and removal of machinery, weapon, and electronic systems through the use of standardized modules, conceptualized in Figure 5.1. There are currently seventy ships built or contracted to be built using this design philosophy. These ships are operated by twelve countries. For this work, the focus will be on the distributed systems, i.e., ventilation, firefighting, and power distribution with networks representing these systems derived from *Dicker* (1986). The networks developed in this section are modeled on the hy-

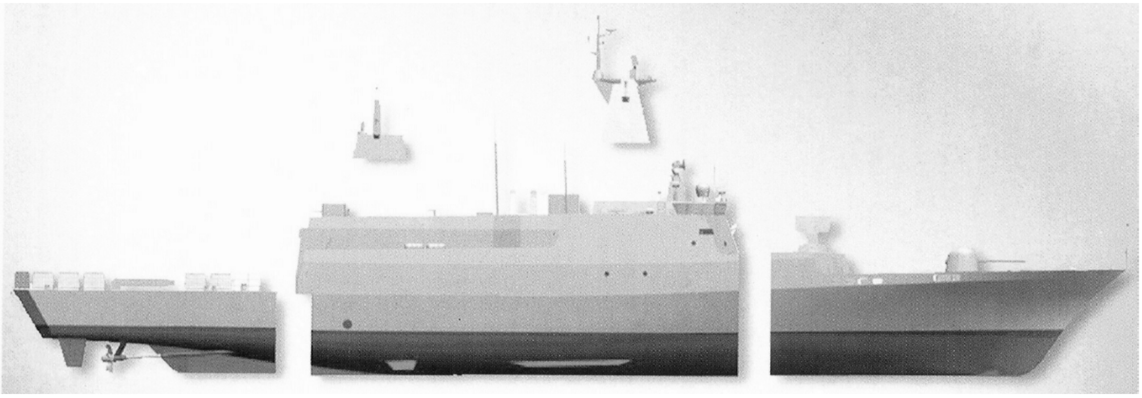


Figure 5.1: An artist's conceptualization of the modular MEKO concept (*Blohm and Voss*, 2003).

pothetical ship design shown in Figures 5.2, 5.4 and 5.6. In general, the advantages of the MEKO system are still debated and typically manifest themselves in the building stage due to the increased modularity of the designs, which lends itself to block construction techniques (*Jacobi*, 2003). In the remainder of this section, network concepts will be used to gain more insight into the differences between the MEKO and traditional distributed systems design philosophies.

5.1.2 Ventilation

A MEKO-style design attempts to split a ship into series of ventilation zones based primary around watertight bulkheads (Figure 5.2). By having multiple, modular zones, it is possible to shutdown just the affected zones in the event of a fire, smoke, or other toxic fume event. For example, in the event of a fire in an auxiliary machinery room, the ventilation in just that room can be turned off, while cooling can still be supplied to an adjacent electronic equipment room. In a conventionally designed ventilation system, the entire HVAC system would need to be shutdown and closed off to prevent the propagation of such a fire through the ducts. According to *Dicker* (1986) the MEKO design, using a Blohm & Voss containerized ventilation unit, costs 75%

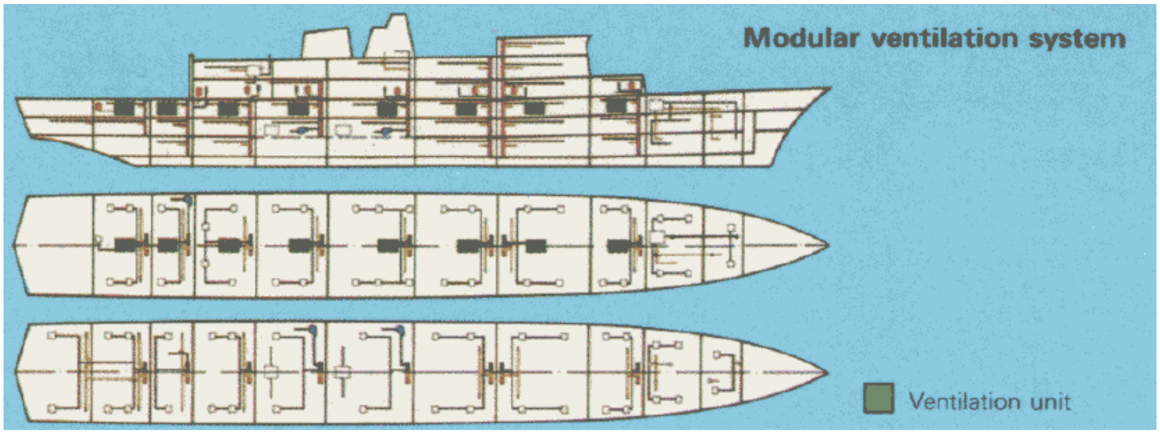
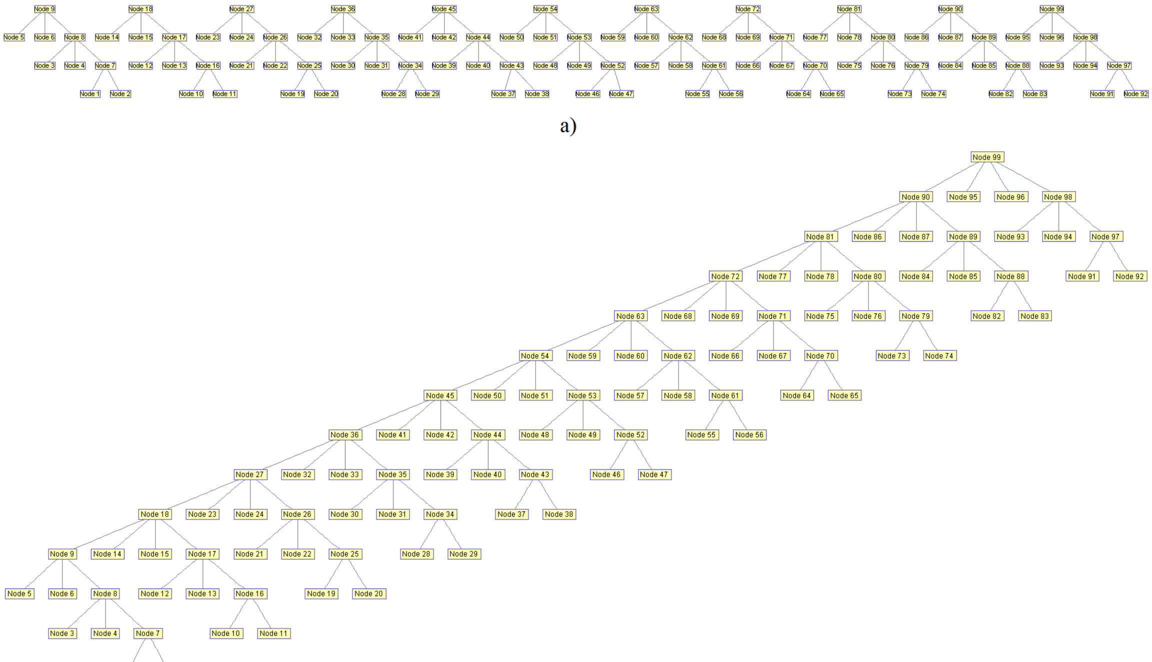


Figure 5.2: The MEKO modular ventilation system (*Dicker*, 1986).

of the traditional system and has a lower center of gravity, though it requires every bulkhead penetration for pipes, cableways, etc. to be smoke and nuclear, biological, and chemical gas-tight.

Figure 5.3 shows two representative ventilation system networks, one modeled after the MEKO design philosophy and the other after a conventional design. The key difference between these two networks is the lack of connectivity in the case of the MEKO design. As discussed previously, this means different parts of the ventilation system can be shutdown independently as opposed to the traditional system where shutting down one of the key duct intersections could potentially shutdown the entire system. Other network metrics are not necessary for analyzing this type of system as modularity is the primary goal of the MEKO design. By the network analysis, the MEKO system is the preferable system for damage control. This analysis can be used in the trade-off analysis between survivability and the extra costs or perceived costs of installing and maintaining the containerized ventilation units in each ventilation zone as well as making each bulkhead penetration between ventilation zones smoke and nuclear, biological, and chemical gas-tight.



a)

b)

Figure 5.3: A network representation of a (a) MEKO style and (b) traditional ventilation system.

5.1.3 Firefighting

The firefighting system in the MEKO concept follows the same modular concept as the ventilation system, but does connect all the compartments with one large fire main, though this main is fed via an independent fire pump located in its own sea chest within each fire-zone in an effort to maintain some modularity. The fire main is located deep within the ship, on centerline, as opposed to outboard on the main deck as with a traditional design. Aside from the location of the mains, the MEKO and traditional firefighting designs are actually quite similar to each other, and have a network structure very similar to that of the traditional ventilation system, as shown in Figure 5.5. The main difference is the traditional system will have two copies of the system shown in Figure 5.5; one on the port side and one on the starboard side. From a network perspective, there is no difference between the MEKO system and

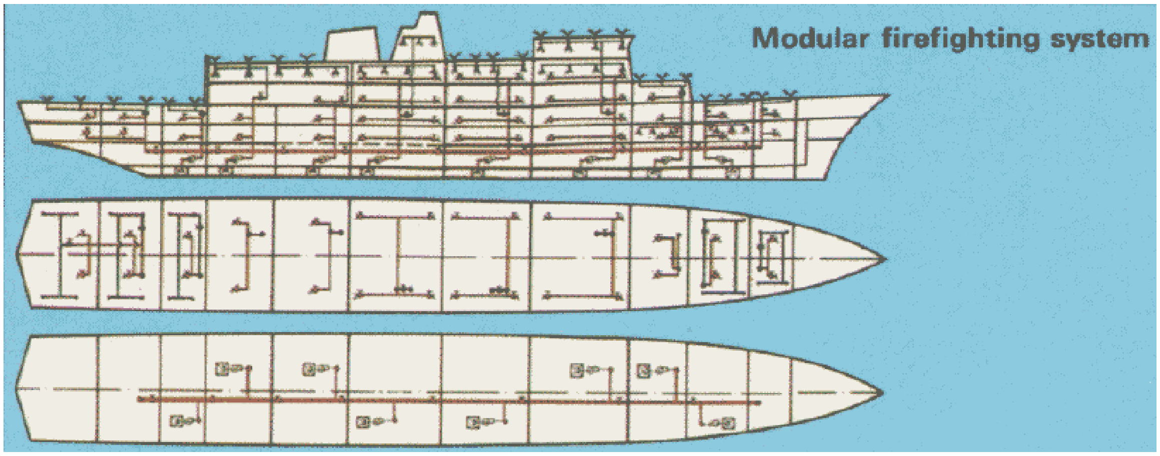


Figure 5.4: The MEKO modular firefighting system (*Dicker*, 1986).

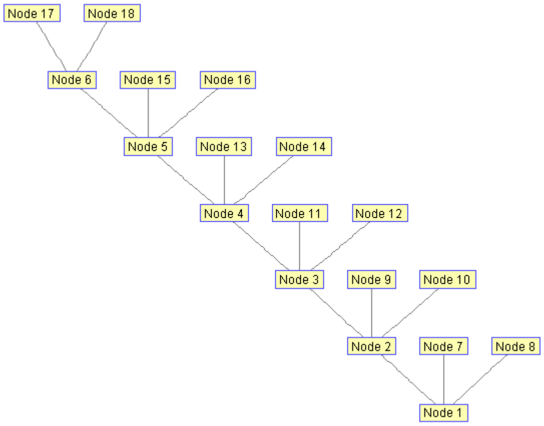


Figure 5.5: A network representation of a firefighting main and hydrant feeders.

the traditional system, but by adding the context of the design the MEKO system will perform better in damage situations due to the redundant and independent fire pumps being located in areas less susceptible to battle damage.

5.1.4 Power distribution

The MEKO-style power distribution system (Figure 5.6) is similar to the firefighting system. Instead of having the main power distribution line underneath the main deck, as is common in a radial system (Figure 5.7), it is relocated to the tank top, just inside

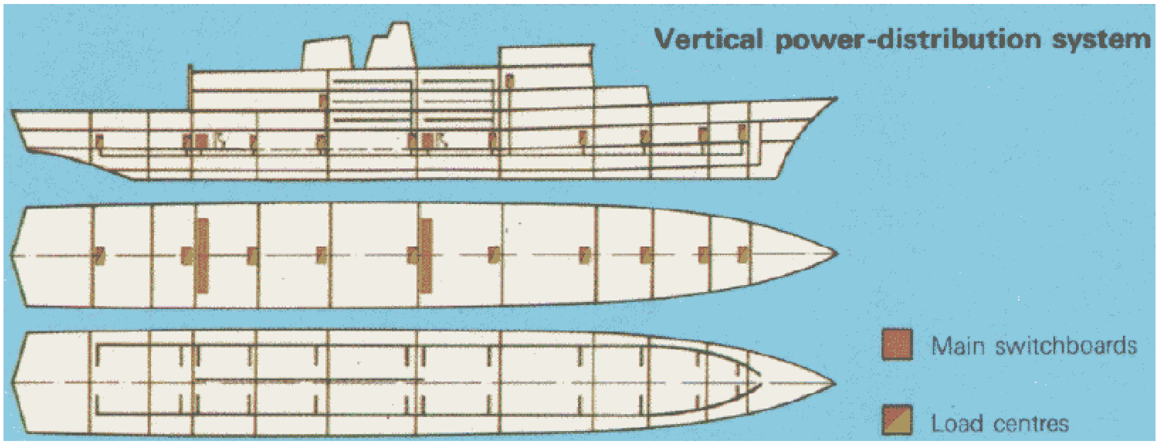


Figure 5.6: The MEKO modular power distribution system (*Dicker*, 1986).

the wing tanks where it is less susceptible to battle damage. Additionally, instead of one main line down the center of the ship, there are two lines: one port and one starboard. These lines are connected to load centers on centerline on the third deck by vertical feeder lines. These load centers then act as the local distribution point for all electrical power inside the watertight compartment. Blohm and Voss claim this setup results in a 20% reduction in weight and cost compared to the traditional system.

Figure 5.8 shows the network representations of a MEKO-style and traditional power distribution system (shown in Figure 5.7) and Table 5.1 shows the properties of the networks. These are undirected, unweighted networks, as there is no additional information available upon which to derive the edge weight values.

The network analysis shows the MEKO design philosophy succeeds in more evenly distributing the power system throughout the ship, as can by seen by comparing the network densities of 0.073 from the MEKO design versus 0.176 for the radial design. Additionally, the MEKO design succeeded in creating a more modular system, with 5 communities instead of 4. The MEKO design also had a higher quality of modularity compared to the radial design (0.598 compared to 0.367, respectively), meaning the

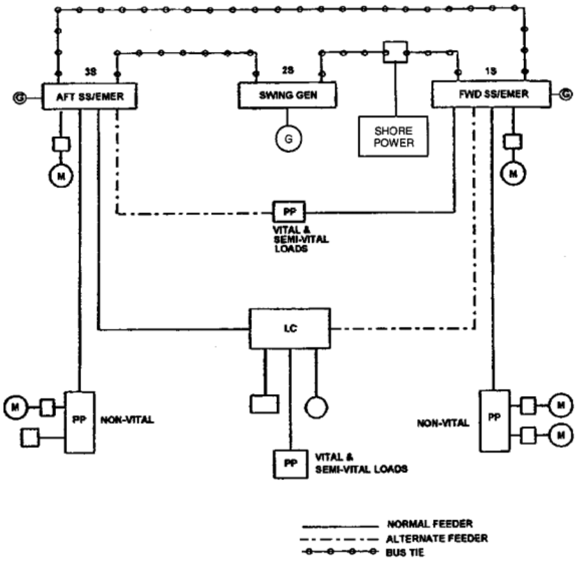


Figure 5.7: Combatant ship service radial distribution with dual purpose generators (*Naval Sea Systems Command*, 2005).

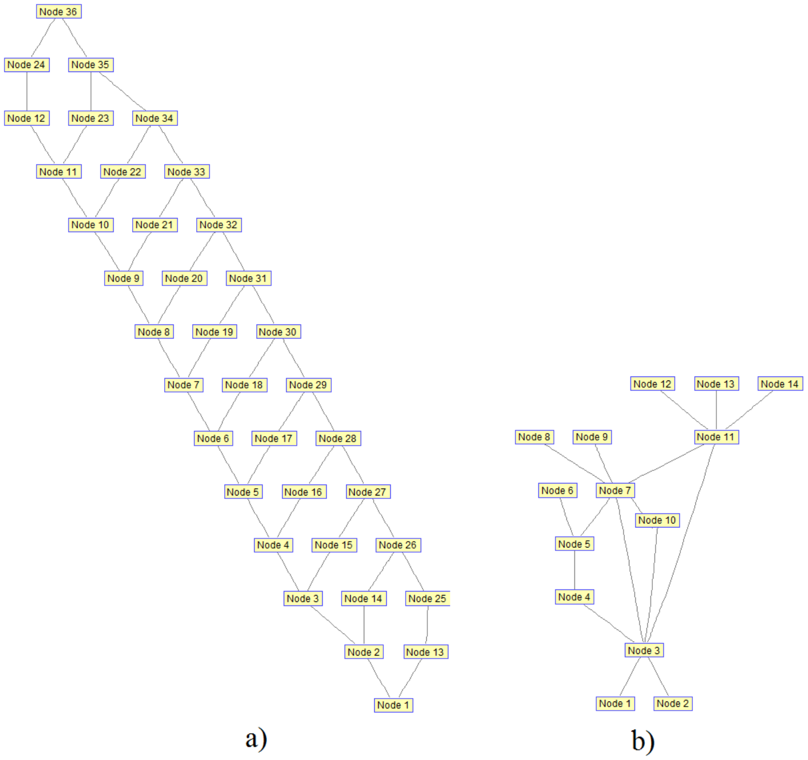


Figure 5.8: A network representation of the (a) MEKO and (b) radial power distribution system.

Table 5.1: Characteristics of MEKO and radial power distribution networks.

Network Property	MEKO	Radial
Nodes	36	14
Edges	92	32
Average Degree	2.556	2.286
Density	0.073	0.176
Diameter	26	8
Max. Betweenness	317.98	76.00
Mean Betweenness	147.33	17.00
No. of Communities	5	4
Quality	0.598	0.367

communities that were created are better defined in the case of the MEKO design. These more modular power distribution communities agree with the MEKO design paradigm and are well suited for block construction techniques.

The MEKO system has a structure akin to a small world network, which means there is a strong probability of maintaining network connectivity in the event of random failures or targeted attacks (*Gong and Zhang*, 2009), a key concern in distributed systems design. In fact, when adding the edges and nodes connected to the load centers, the MEKO structure takes on the form of an n-Star network, which has been shown to be one of the most robust network structures to targeted attacks and random failures (*Sawai*, 2013). Contrarily, the radial system is more similar to a scale-free network, which is extremely susceptible to targeted attacks. The differences between zonal and radial power distribution systems will be further investigated in Section 5.2.2.

5.1.5 Individual distributed systems conclusions

Unlike the ventilation and firefighting systems, it is possible to definitively state that MEKO modular or zonal power distribution system has advantages over the traditional, radial power distribution system, which is why the zonal system is becoming the standard for large naval ships. (*Naval Sea Systems Command*, 2005; *Doerry*, 2006a). For the ventilation and firefighting systems, the advantages of one system over the other are less clear, beyond the costs savings claimed by Blohm & Voss, though modular versions of ventilation and firefighting systems, along with other distributed systems, are too becoming more popular (*Petry and Rumburg*, 1993; *Zhang and Sui*, 2002), especially as distributed systems needs are becoming both larger and more localized (*Frank and Helmick*, 2007), as is evidenced by the growth in installed air conditioning plant capacity on new build navy ships compared to the legacy ships as shown in Figure 5.9.

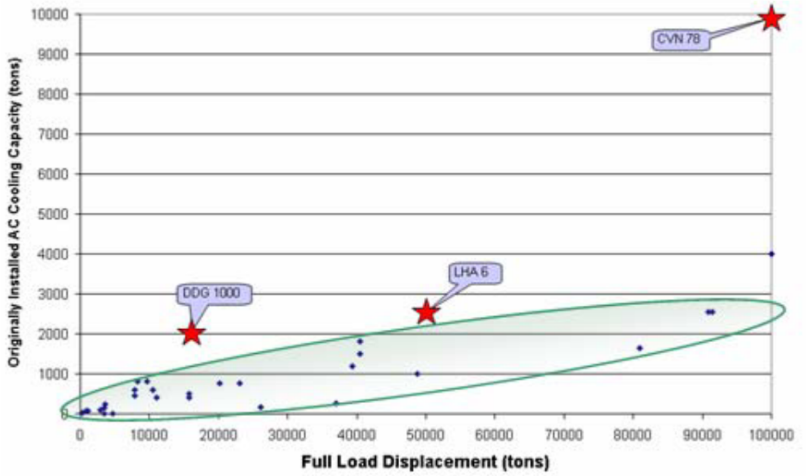


Figure 5.9: Installed AC plant capacity on new build ships (stars) versus legacy ships (circled) (*Frank and Helmick*, 2007).

5.2 Design evolution analysis

Over the course of a design, a system will go through countless iterations. Each iteration requires the same analyses, which is both time consuming and potentially expensive. If two iterations are similar to each other, and the analyses return the same results, the time and money spent on those analyses is wasted. A method for checking if two designs are so similar that additional analysis is not necessary is essential for removing redundant analysis from the design process. Additionally, such an analysis can be used to determine if the change has a positive, negative, or neutral impact on the overall design. Currently, there is no method for quantifying this impact due to the limited amount of design information available in early stage design. This is possible using a sequential, multislice network and related community detection techniques, but first the validity of multislice community detection for design and physical systems must be verified. That is the purposed of this section.

In the ensuing subsections, multislice network structures will be used to investigate two instances of design evolution. First, the focus will be on the design process, where the differing impact of design changes in the early design stage versus the later design stage will be investigated. The focus will then shift back to physical systems, where a further investigation of the zonal and radial power distributions systems will be conducted.

5.2.1 The effect of design changes

As shown in Figure 5.10, design decisions later in the design process are associated with a great increase in cost than those made at the beginning of the process (*Bragança et al.*, 2014; *Rehman and Yan*, 2008). Unfortunately, as mentioned numerous times throughout this dissertation already, it is not always possible to make the correct decision early in the design process because of a lack of information about the design. In a perfect world, the design process and tools would be modular to a point that a change in one design variable would not affect any other variable. Alas, this is not the case and almost all variables, especially in the naval design realm are coupled. Since it is infeasible to remove this coupling, the next best thing would be to understand the coupling and how design variables become more inter-related as the design process progresses. This is possible with multi-resolution, multislice community detection analysis, in the same vein as that done by *Mucha et al.* (2010) for the Zachary Karate Club Network.

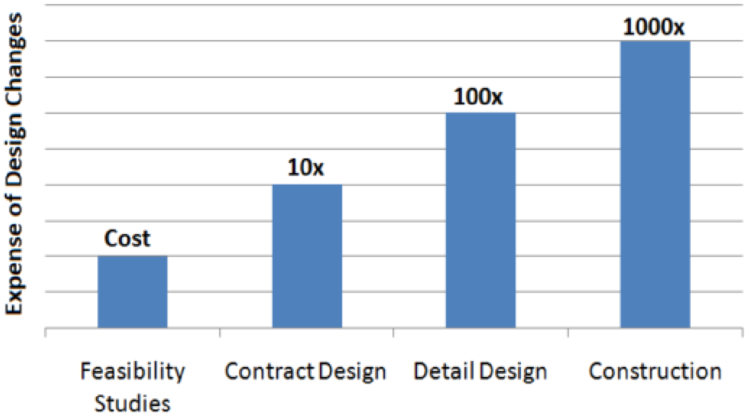
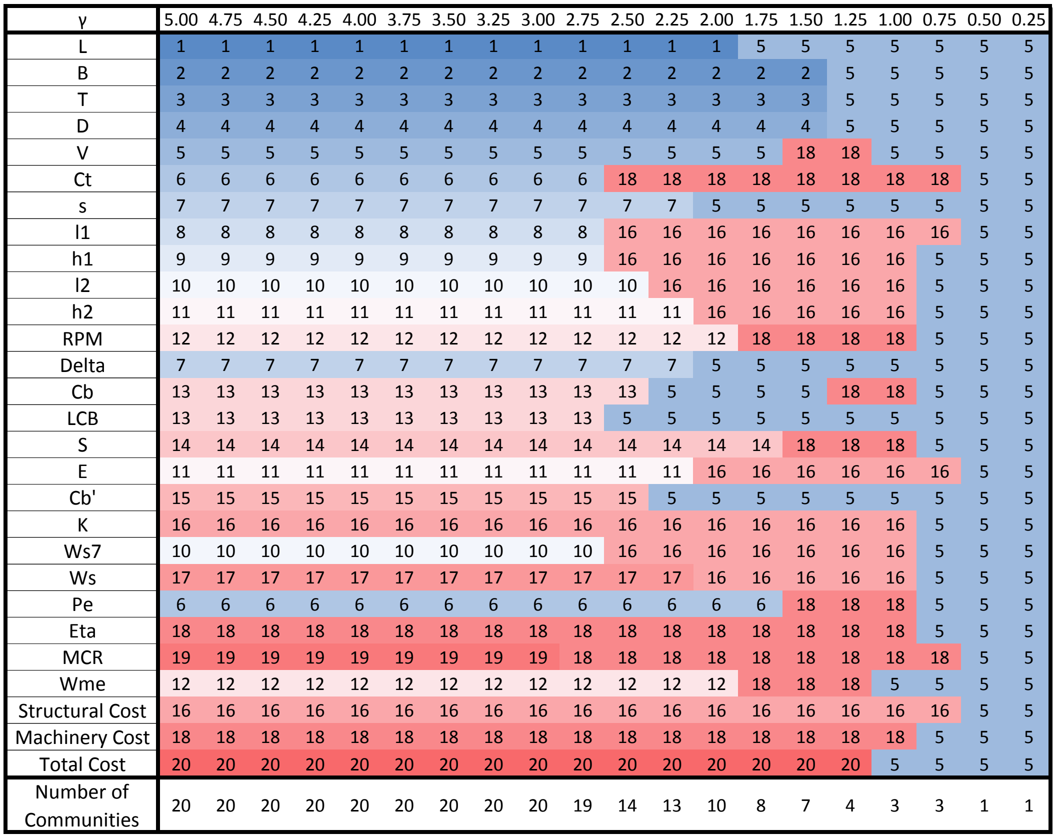


Figure 5.10: Cost of design changes during different naval ship design phases (*McKenney*, 2013).

To show this process, the Watson and Gilfillan ship design equations (*Watson and Gilfillan*, 1977) will be used as an analog for the entirety of the design process. A network representation of the variables in these equations (*Parker and Singer*, 2013) will be used to construct a multislice network of 20 slices. In each slice, nodes represent variables and edges represent two variables being in the same equation.

For the multislice community detection algorithm, the resolution factor, γ, in each slice is reduced, causing the likelihood of a community linkage to increase thereby increasing the size of communities. This mimics the design process, where in the early stage variables are less coupled together, *i.e.*, communities of variables are smaller, and then, as the process progresses, the coupling of variables increases, meaning the communities should grow in size and decrease in number.

The resulting community structures are shown in Figure 5.11. The numbers in the shaded boxes serve to identify to which community each variable in each slice belongs. The communities for each network slice are calculated using a different resolution



γ	5.00	4.75	4.50	4.25	4.00	3.75	3.50	3.25	3.00	2.75	2.50	2.25	2.00	1.75	1.50	1.25	1.00	0.75	0.50	0.25
L	1	1	1	1	1	1	1	1	1	1	1	1	1	5	5	5	5	5	5	5
B	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	5	5	5	5	5
T	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	5	5	5	5	5
D	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	5	5	5	5	5
V	5	5	5	5	5	5	5	5	5	5	5	5	5	5	18	18	5	5	5	5
Ct	6	6	6	6	6	6	6	6	6	6	18	18	18	18	18	18	18	18	5	5
s	7	7	7	7	7	7	7	7	7	7	7	7	5	5	5	5	5	5	5	5
l1	8	8	8	8	8	8	8	8	8	8	16	16	16	16	16	16	16	16	5	5
h1	9	9	9	9	9	9	9	9	9	9	16	16	16	16	16	16	16	5	5	5
l2	10	10	10	10	10	10	10	10	10	10	10	16	16	16	16	16	16	5	5	5
h2	11	11	11	11	11	11	11	11	11	11	11	11	16	16	16	16	16	5	5	5
RPM	12	12	12	12	12	12	12	12	12	12	12	12	12	18	18	18	18	5	5	5
Delta	7	7	7	7	7	7	7	7	7	7	7	7	5	5	5	5	5	5	5	5
Cb	13	13	13	13	13	13	13	13	13	13	13	5	5	5	5	18	18	5	5	5
LCB	13	13	13	13	13	13	13	13	13	13	5	5	5	5	5	5	5	5	5	5
S	14	14	14	14	14	14	14	14	14	14	14	14	14	14	18	18	18	5	5	5
E	11	11	11	11	11	11	11	11	11	11	11	11	16	16	16	16	16	16	5	5
Cb'	15	15	15	15	15	15	15	15	15	15	15	5	5	5	5	5	5	5	5	5
K	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	5	5	5
Ws7	10	10	10	10	10	10	10	10	10	10	16	16	16	16	16	16	16	5	5	5
Ws	17	17	17	17	17	17	17	17	17	17	17	17	16	16	16	16	16	5	5	5
Pe	6	6	6	6	6	6	6	6	6	6	6	6	6	6	18	18	18	5	5	5
Eta	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	5	5	5
MCR	19	19	19	19	19	19	19	19	19	18	18	18	18	18	18	18	18	18	5	5
Wme	12	12	12	12	12	12	12	12	12	12	12	12	12	18	18	18	5	5	5	5
Structural Cost	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	16	5	5
Machinery Cost	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	18	5	5	5
Total Cost	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	20	5	5	5	5
Number of Communities	20	20	20	20	20	20	20	20	20	19	14	13	10	8	7	4	3	3	1	1

Figure 5.11: A multiple resolution network community analysis of the Watson and Gilfillan ship design equations.

factor, γ, as indicated in the top row. Decreasing γ leads to a decrease in the number of communities as shown in the bottom row. In this example, the interslice coupling factor was held constant at $\omega = 1$.

This analysis returns intuitive community structures and patterns and leads to conclusions similar to those found using the planarity-complexity metric in Chapter 3. In agreement with findings of *Parker and Singer* (2013), the design length is independent of any other variables until the time in the design that it is fixed, and all the other variables are related to it ($\gamma = 1.75$). Machinery cost and efficiency (Eta) are in same community (first community eighteen, then community five) throughout the entire design process, which is expected as the more efficient machinery tends to be of higher quality and higher cost. As the design process progresses, the machinery community expands to include design speed (V), coefficient of thrust (Ct), engine RPM, block coefficient (Cb), service margin (S), and machinery weight (Wme) signifying that these variables become more coupled to each other as the design progresses than they do to other variables like beam (B) or draft (T). Eventually, all the variables are placed in the same community signifying any change in one variable will necessarily effect the entirety of the design. The way in which the communities grow and group certain variables together follows the logical division of work in a ship design and agrees with the separation of task that was depicted in Figure 2.4.

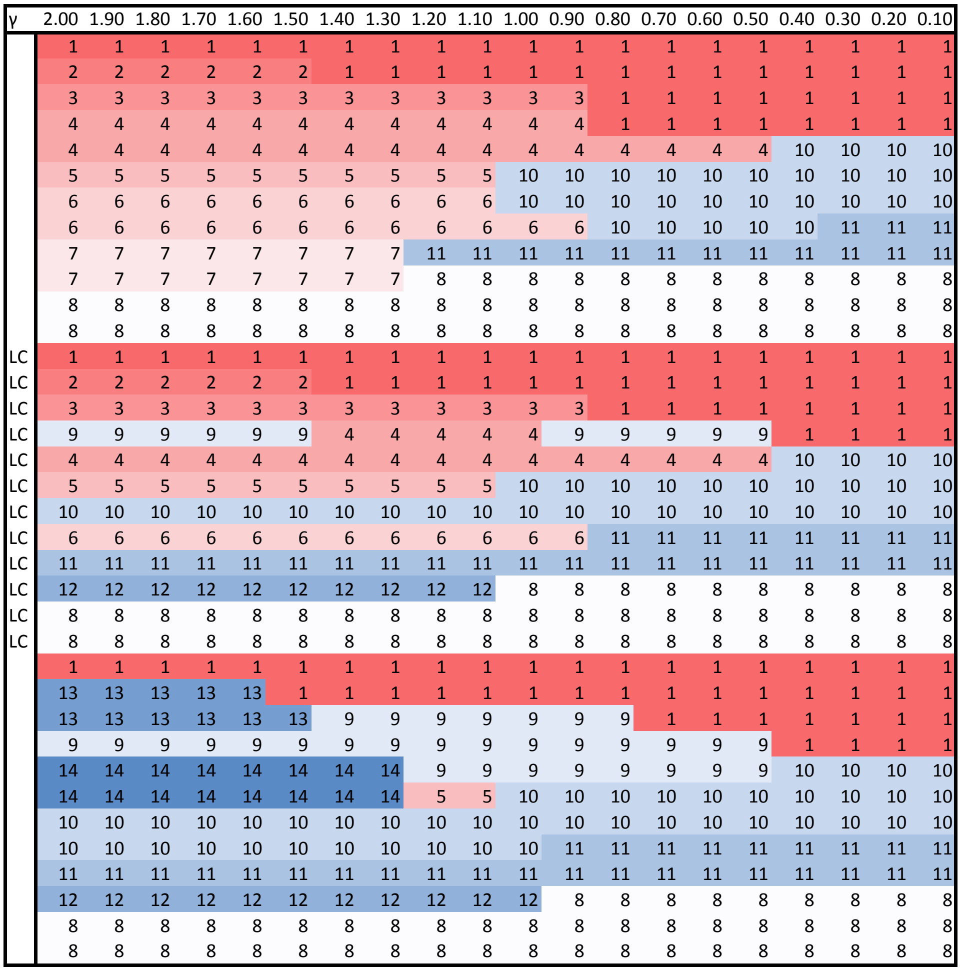
When the trends shown in the table are considered holistically, again the expected trends are found. Early in the design stage, when $5 \geq \gamma \geq 3$, the most communities were found corresponding to the most flexibility in design decisions. This region was also relatively static. When $\gamma \leq 2.75$, the number of communities decreased rapidly, denoting a tipping point in the design process where previously independent design decisions become increasing coupled and changing one parameter becomes increasingly expensive as it affects all the other parameters. When $\gamma \leq 0.50$, there is only one community present, meaning any parameter change affects the entirety of a design, which mirrors the real design process.

This is only a representative example of how this analysis would be used when analyzing an entire design process. In a full process analysis, different steps in the design process would be represented using different network structures. Additionally, network models of the design process in the later design stages would likely have more nodes than network models of the earlier design process. This can be handled by also including those nodes in every network slice, but not connecting them to any other nodes in that slice. This more in-depth analysis, utilizing network models with increasing fidelity, would return the same trends found in the Watson and Gilfillan example, but with increased information about the interactions between pieces of design information, proportionate to the increased fidelity of the added network models.

5.2.2 Zonal versus radial shipboard power distribution systems revisited

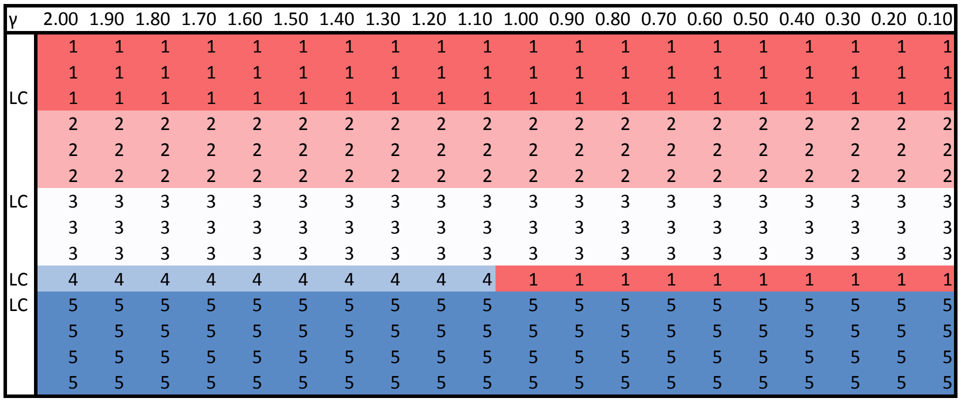
In Section 6.1, some of the advantages of the zonal power distribution system over the radial power distribution system were discussed. In this section, these two design philosophies will be further analyzed using the multiple resolution, multislice community detection method just presented. The MEKO zonal power distribution system network and the radial power distribution system network (Figure 5.8) will be used to create a twenty-slice multislice network. Over the twenty slices, the resolution factor, γ, will be decreased, in even steps, from 2 to 0.10. The interslice coupling factor, ω, will be kept constant. Figures 5.12 and 5.13 show the effects this has on the community structure, with a special emphasis on the nodes representing load centers.

As in the design decision case, as the value of γ is decreased, the number of communities for each network is reduced (from 14 to 4 for the zonal system and 5 to 4 for the radial system); nodes that were originally in separate communities are combined together. In the case of the radial system, only one node changes its group affiliation, and this node is a load center. It is expected that a load center would be the most likely to join another community as the load centers should be well connected to the rest of the network and therefore affected the most by the changing resolution factor. The issue with the radial system is this does not happen enough. In the event of a failure or some other unforeseen power spike at one load center, it is desired that the rest of the system could be adjusted to maintain service. Such an occurrence is modeled by decreasing of the resolution factor which in turn increases the community size.



γ	2.00	1.90	1.80	1.70	1.60	1.50	1.40	1.30	1.20	1.10	1.00	0.90	0.80	0.70	0.60	0.50	0.40	0.30	0.20	0.10
	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	3	3	3	3	3	3	3	3	3	3	3	3	1	1	1	1	1	1	1	1
	4	4	4	4	4	4	4	4	4	4	4	4	1	1	1	1	1	1	1	1
	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	10	10	10	10
	5	5	5	5	5	5	5	5	5	5	10	10	10	10	10	10	10	10	10	10
	6	6	6	6	6	6	6	6	6	6	10	10	10	10	10	10	10	10	10	10
	6	6	6	6	6	6	6	6	6	6	6	6	10	10	10	10	10	11	11	11
	7	7	7	7	7	7	7	7	11	11	11	11	11	11	11	11	11	11	11	11
	7	7	7	7	7	7	7	7	8	8	8	8	8	8	8	8	8	8	8	8
	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
LC	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
LC	2	2	2	2	2	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1
LC	3	3	3	3	3	3	3	3	3	3	3	3	1	1	1	1	1	1	1	1
LC	9	9	9	9	9	9	4	4	4	4	4	9	9	9	9	9	1	1	1	1
LC	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	4	10	10	10	10
LC	5	5	5	5	5	5	5	5	5	5	10	10	10	10	10	10	10	10	10	10
LC	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10
LC	6	6	6	6	6	6	6	6	6	6	6	6	11	11	11	11	11	11	11	11
LC	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11
LC	12	12	12	12	12	12	12	12	12	12	8	8	8	8	8	8	8	8	8	8
LC	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
LC	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	13	13	13	13	13	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	13	13	13	13	13	13	9	9	9	9	9	9	9	1	1	1	1	1	1	1
	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	9	1	1	1	1
	14	14	14	14	14	14	14	14	9	9	9	9	9	9	9	9	10	10	10	10
	14	14	14	14	14	14	14	14	5	5	10	10	10	10	10	10	10	10	10	10
	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10	10
	10	10	10	10	10	10	10	10	10	10	10	11	11	11	11	11	11	11	11	11
	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11	11
	12	12	12	12	12	12	12	12	12	12	12	8	8	8	8	8	8	8	8	8
	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8
	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8	8

Figure 5.12: Zonal power distribution system community structure under multiple resolutions. LC denotes load centers.



γ	2.00	1.90	1.80	1.70	1.60	1.50	1.40	1.30	1.20	1.10	1.00	0.90	0.80	0.70	0.60	0.50	0.40	0.30	0.20	0.10
	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
LC	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2
LC	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3	3
LC	4	4	4	4	4	4	4	4	4	4	1	1	1	1	1	1	1	1	1	1
LC	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5
	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5	5

Figure 5.13: Radial power distribution system community structure under multiple resolutions. LC denotes load centers.

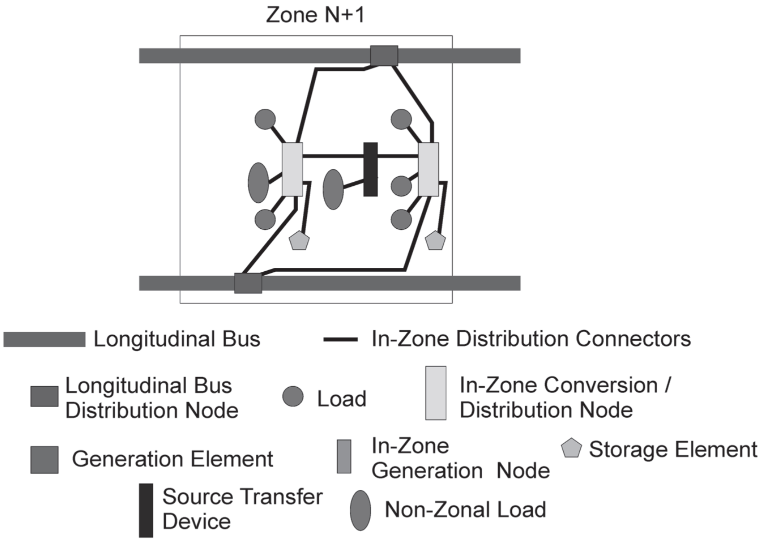


Figure 5.14: A sub-zone electrical distribution system design for handling non-zonal loads.

In the case of the zonal power distribution system, the nodes become more likely to be coupled as the resolution factor is decreased, just like they would if there were a sudden large power demand (for example, firing a rail gun or other high energy weapon). The structure of the network allows for the communities to easily reform and combine to provide extra power to meet this peak, and hopefully transient, demand. This flexibility of communities also improved system survivability as one load center can take over for a neighboring non-functioning load center. If two load centers and their respective nodes were placed within each watertight or damage control compartment, then this could be done with the minimum amount of penetrations in the watertight bulkheads. This two electrical zone per compartment design, termed “sub-zones” was introduced in *Doerry* (2006b), and an example of the design for dealing with non-zonal loads is shown in Figure 5.14.

5.2.3 Multislice community detection conclusions

One of the goals of this dissertation is to use multislice network structures and community detection as a way of tracking design evolution. Towards this goal, it was necessary to validate this method for known design problems. The previous two subsections showed the applicability of mutlislice networks to the naval design problem. It was believed multiple resolution community detection could be used to analyze the evolution of network communities as nodes become more connected, and this was confirmed by using the Watson and Gilfillan as a proxy for the ship design process and conducting a community detection experiment. The communities that emerged agreed with the commonly accepted division of analysis in the naval architecture design process. This confirmed that multislice community detection was valid approach for evaluating a design process.

Next, the acceptability of multislice networks and community detection was tested for physical systems. The evolution of communities (or lack thereof) for the respective zonal and radial networks corresponded with design philosophies of the two systems. For the zonal system, as the local nodes became more coupled (representing either a peak load or possible failure) the communities grew representing the flexibility of power distribution within the system. Contrarily, the communities for the radial power distribution network barely changed which agrees with the static nature of the distribution system. Not only did multiple resolution community detection uncover the inherent nature of the two systems, it also demonstrated why the zonal power distribution structure, thanks to its flexibility and reconfigurability, has started to become the norm in ship design.

For these reasons, it is believed that multislice network structures and community detection are valuable methods for the analysis of sequential iterations of ship systems designs. These methods will be used in the next chapter in a representative ship design.

5.3 Analysis of system interactions

For the remainder of this chapter the focus will shift from the analysis of systems in solitude, to the analysis of interactions between systems. For the final two studies of this chapter, the case of the USS Yorktown going dead in the water in 1997 will be investigated using both a static multiplex structure and new time-dependent multiplex structure.

These two structures will be used to investigate two different failure modes that caused the propulsion failure. The static multiplex structure, along with community detection, will be used to test the interdependency of the distributed systems required to operate the engines. The time-dependent multiplex structure, developed for this dissertation, will be used to represent the cascading failure that caused the ship to lose power. These two experiments are done to show the applicability of the multiplex and time-dependent multiplex structure to physical systems design. These two structures will also be used in the next chapter to aid in ship design case study.

5.3.1 USS Yorktown (CG-48)

The USS Yorktown was the second ship of the Ticonderoga Class Guided-Missile Cruisers. It was contracted to Ingalls Shipbuilding, in Pascagoula, Mississippi in 1980 and commissioned in 1984. It was decommissioned in 2004. The ship was 173 meters long, 16.8 meters at maximum beam, and had a draft of 10.2 meters. It displaced approximately 9,600 tons. As a guided-missile cruiser, the Yorktown was designed to use the Aegis system, a suite of high powered radars and computers for tracking and guiding weapons (*Global Security*, n.d.). On top of the complexity of the Aegis system, the Yorktown was chosen as the test bed of the US Navy's Smart Ship program (*Evers*, 1997). The program reduced manning by roughly ten percent by integrating many redundant tasks such as engine room and bridge monitoring into a centralized unit, as can be seen in Figure 5.15.

Unfortunately, integrating all the systems and managing them from a central control station led to an unforeseen, extreme event. On September 21, 1997, an error in data entry into the main computer caused a fault in the ship's main control system (*Slabodkin*, 1998; *Wired News*, 1998). This fault spread to the LAN switches causing them to turn off. When the LAN switches turned off, the engine controllers also switched off. Finally, when the engine controllers switched off, the engines themselves switched off, causing the ship to go dead in the water off the coast of Cape Charles, Virginia.

The crew was not able to restart the vessel and it was towed into port. The original fault occurred at the operating system level when a crew member mis-entered a command causing a divide by zero error, this error then cascaded causing a one billion dollar warship (*United States Navy*, 2013) to become inoperable. According to reports, the ship had suffered similar issues in the past (*Slabodkin*, 1998).

This extreme event is an example of the kind of complex, emergent behavior known as a "dragon-king" (*Sornette*, 2009) that designers should seek to avoid. The complete loss of propulsion power due to miss-entered data is such an outlier in the normal course of naval operations that it could not be predicted using traditional means, but using the combination of multiplex and time-dependent multiplex structures a designer may have been able to foresee the potential for failure caused by the decision to reduce shipboard manning.

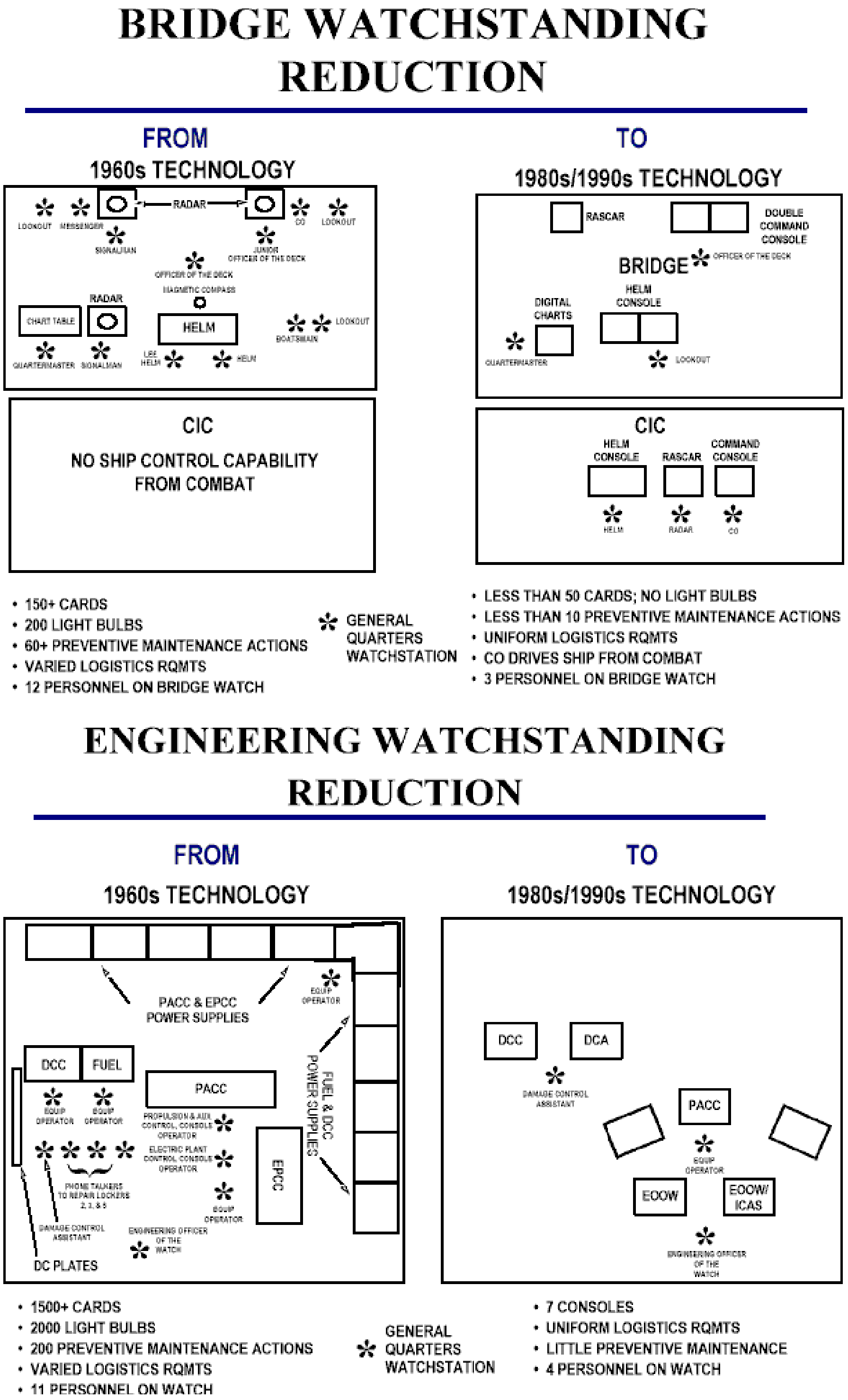


Figure 5.15: Two examples of measures taken by the Smart Ship program to reduce watch-standing requirements (*Global Security*, n.d.).

5.3.2 System interactions using multiplex community detection

Representative network models of the LAN switch, engine controller, and engine networks are presented in Figure 5.16. Each network is an undirected, unweighted network. These networks were created based on intuition and are intended as representative models for proving the method, not for analyzing the exact system structure present on the USS Yorktown. These three networks were then combined into a multiplex network structure like the one shown in Figure 3.9. By varying the interslice connection strength, ω, the degree of interdependence of the three systems can be investigated. In the case of the USS Yorktown, the three systems were highly interdependent which caused the sudden loss of operability. It is desired to find the value of ω where every node in each of the three systems is in the same community, this signifies a set of systems are completely interdependent and a fault at one level of the system would cause a total system shutdown. This is done by increasing the value of ω from 0 to investigate how the community structure changed. Table 5.2 shows some properties of multiplex community structure with a discussion to follow.

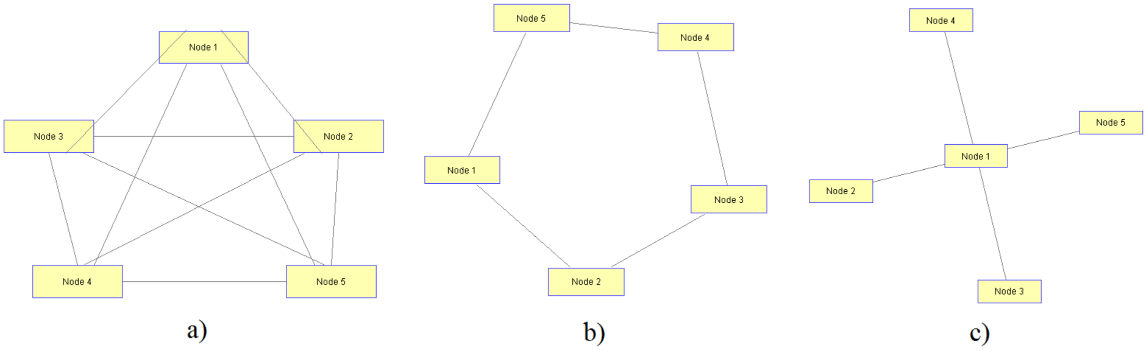


Figure 5.16: Representative (a) LAN switch, (b) engine controller, (c) and engine system networks of the USS Yorktown.

It can be seen that the interslice connection strength did not need to be increased greatly from 0 as the individual system networks were rather simple, but the expected trends were seen, the number of communities decreased while the quality of

Table 5.2: USS Yorktown multiplex community properties

ω	0	0.025	0.05	0.075	0.1
% of nodes in multiple communities	100	60	60	0	0
# of communities	4	2	2	1	1
Quality	0.0211	0.0323	0.043	0.0559	0.0732

the communities increased. This decreased the number of nodes that were in different communities, meaning the networks were more interconnected. The fact that one community was created for $\omega \geq 0.075$ indicates that the individual systems were highly interconnected by design which agrees with the fact that failure propagated so quickly in the case of the USS Yorktown. This type of analysis can help a design see what potential low occurrence events will could lead to a catastrophic failure.

5.4 Analysis of system interactions through time

After analyzing static multiplex networks and time dependent simplex networks, the next step is to combine the two into a time-dependent multiplex structure, as shown in Figure 5.17. By analyzing the communities for this type of structure, the manner in which the entirety of the ship systems design evolves including the interactions between systems can be investigated. To discern these communities, a new modularity equation (Equation 5.1), developed in this dissertation, is necessary as another linkage term is required. This equation is described in the following paragraphs.

$$Q = \frac{1}{2\mu} \sum_{ijsrxy} \left\{ \left(A_{ijxy} - \gamma_{sx} \frac{k_{isx} k_{jsx}}{2m_{sx}} \right) \delta_{sr}\delta_{xy} + \delta_{ij}\delta_{xy} C_{jxsr} + \delta_{ij}\delta_{sr} E_{jsxy} \right\} \delta(g_{isx}, g_{jry}) \tag{5.1}$$

Equation 5.1 is an expansion of the multiplex/multislice spectral modularity function (Equation 3.7) that accounts for both the static all-to-all interslice connections at one

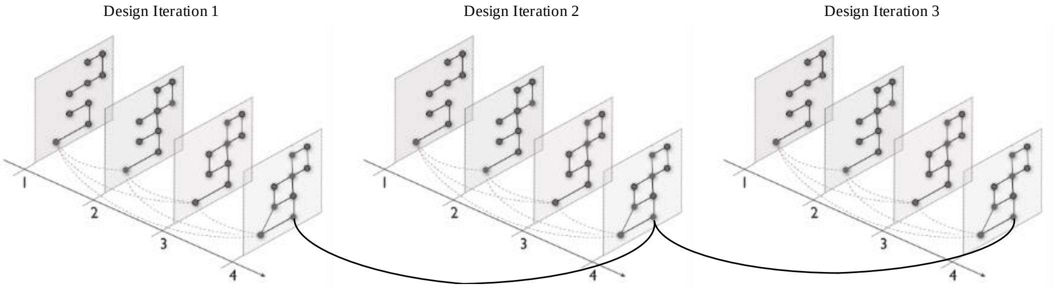


Figure 5.17: The time-dependent multiplex network structure.

time step, as well as the time dependent connections between different time steps. Adding the time dependencies requires the addition of a new “interplex” connection term, E_{jsxy}, which signifies the connection between node j in slice s in time step x to the same node j in slice s in time step y. As this is a sequential structure, $E_{jsxy} = \zeta$ when $x = y \pm 1$ and $E_{jsxy} = 0$ when $x \neq y \pm 1$. This is consistent with valuing of ω in the multislice community detection.

While it would be possible to use the original multiplex/multislice modularity function to achieve the same ends, the C matrix would become significantly more difficult to handle. Additionally, it is believed that the developed equation is more intuitive and easier to conceptualize, especially when paired with Figure 5.17. The addition of this term also means the multiplex interaction weights can be varied for each time step, if desired.

For the USS Yorktown case, the ship did not instantly lose propulsion power, though it did happen very quickly. Time-dependent multiplex community detection will be used to investigate the interaction between system interdependence and the speed of propagation of failures. While the optimal would be to have no interdependence and no propagation of failure, this is a utopia point that is not reachable. Rather, systems should be constructed in such a way that either interdependence is at a minimum to reduce failures spreading from one system to another, or failures propagate slowly so that the crew has time to resolve the issue before it causes a system-wide catastrophe. To model this, the already introduced USS Yorktown multiplex network will be used to construct a time-dependent multiplex network, and then the strength of interslice connections and the strength of time-dependent connections will be varied.

Tables 5.3 to 5.5 show the number of communities, the percentage of nodes in different communities, and the quality of communities of the USS Yorktown time-dependent multiplex network as ω is varied to change the strength of interslice interactions as a measure of system interconnectedness and as ζ is varied as a measure of the speed that a fault can propagate. Here, the greater the value of ω, the strong the interslice connections, meaning failure is more likely to propagate between systems. Higher values of ζ correspond to stronger time-dependent connections, meaning each sequential multiplex has a stronger connection to its adjacent multiplex. This corresponds to a larger delta between time steps, therefore there is a longer time for failure to propagate between times steps. The larger the value of ζ that is required to form communities between time steps, the slower a hypothetical failure would propagate.

As was seen in the static multiplex example, very low values of ω caused the instances of nodes in different systems to form into the same community, once again pointing to the high interdependence of the systems. It is interesting that the time coupling factor ζ had little affect on the number of communities (Table 5.3), this is attributed to the size and simplicity of the tested networks. One must look at the additional information provided by the fraction of nodes in different communities and quality (Tables 5.4 and 5.5) to gain further insights into the network structures.

In Table 5.4, the fraction of nodes grouped into different communities does not smoothly decrease as either ω or ζ increase while the other is held constant. This will be further investigated by focusing on the separate cases of $\omega = 0.050$ and $\zeta = 0.100$.

Table 5.3: Number of communities of the USS Yorktown time-dependent multiplex network for combinations of ω and ζ.

		ζ								
		0.000	0.010	0.025	0.050	0.075	0.100	0.250	0.500	1.000
	0.000	12	4	4	4	4	4	4	4	4
	0.010	6	2	2	2	2	2	2	2	2
	0.025	6	2	2	2	2	2	2	2	2
	0.050	6	2	2	2	2	2	2	2	2
ω	0.075	4	2	1	1	1	1	1	1	1
	0.100	4	1	1	1	1	1	1	1	1
	0.250	3	1	1	1	1	1	1	1	1
	0.500	3	1	1	1	1	1	1	1	1
	1.000	3	1	1	1	1	1	1	1	1

Table 5.4: Fraction of nodes in USS Yorktown time-dependent multiplex network grouped into different communities for combinations of ω and ζ.

		ζ								
		0.000	0.010	0.025	0.050	0.075	0.100	0.250	0.500	1.000
	0.000	1	1	1	1	1	1	1	1	1
	0.010	1	1	0.6	0.6	0.6	0.4	0.6	0.4	0.4
	0.025	1	1	0.6	1	0.4	0.4	0.4	0.4	0.4
	0.050	1	1	0.6	1	0.8	0.8	0.4	0.6	0.4
ω	0.075	1	0.6	0	0	0	0	0	0	0
	0.100	1	0	0	0	0	0	0	0	0
	0.250	1	0	0	0	0	0	0	0	0
	0.500	1	0	0	0	0	0	0	0	0
	1.000	1	0	0	0	0	0	0	0	0

Table 5.5: Quality of modularity of communities of the USS Yorktown time-dependent multiplex network for combinations of ω and ζ.

		ζ								
		0.000	0.010	0.025	0.050	0.075	0.100	0.250	0.500	1.000
	0.000	0.021	0.025	0.032	0.043	0.056	0.062	0.135	0.225	0.359
	0.010	0.026	0.030	0.038	0.049	0.058	0.075	0.138	0.228	0.361
	0.025	0.032	0.036	0.045	0.053	0.071	0.082	0.145	0.233	0.363
	0.050	0.045	0.049	0.050	0.064	0.076	0.086	0.155	0.236	0.368
ω	0.075	0.058	0.065	0.067	0.079	0.090	0.101	0.160	0.244	0.369
	0.100	0.073	0.078	0.084	0.095	0.106	0.116	0.174	0.255	0.377
	0.250	0.165	0.168	0.174	0.183	0.191	0.200	0.248	0.315	0.420
	0.500	0.283	0.286	0.290	0.296	0.303	0.309	0.345	0.397	0.479
	1.000	0.441	0.443	0.445	0.449	0.453	0.457	0.479	0.513	0.568

For $\omega = 0.050$, when $\zeta = 0$, every instance of every node is in a different community, one for each time step. When $\zeta = 0.025$, two of the nodes become grouped together across time steps, but then in the next step ($\zeta = 0.050$) each node is in different communities again. This points to complex nature of these systems interactions. Also, by consulting Table 5.5, it can be seen that the quality of community structures for these low values of ω and ζ is also very low, meaning the community structure is both ill-defined and fluid. The same trends can be seen when reading down the column for $\zeta = 0.100$. The faction of nodes first decreases as ω increases, but then jumps at $\omega = .050$ before falling to 0 when all the nodes become coupled. Again, it can be see that the quality is quite low in this unstable region, but then rises quickly once $\omega \geq 0.075$ and the intersystem coupling factor becomes dominant.

In the case of the USS Yorktown, if the systems could have been designed to be slightly more autonomous, then the ship may not have lost engine power. It does not appear that building additional robustness into each system independently to slow down the propagation of failure in the that system would have prevented the event. This is supported by the concepts introduced in Section 3.4 that highly interdependent networks have reduced robustness to both random and targeted failures because the fault can jump between networks, spreading through weakly connected nodes back to the key highly connected nodes.

Even if the time-dependent multiplex analysis does not expressly show through which path a fault will propagate, it can serve as a lead indicator for the existence of complex relationships do exist between the constituent subsystems, with a high chance of emergent behavior. Systems that showed these lead indicators could then be subjected to additional analyses, while the analysis of systems that did not display these phenomena could be reserved for a later time, thus increasing the speed of the design process while sufficiently managing design risk.

5.5 Distributed systems analysis conclusions

This chapter took the network concepts from Chapter 4 and showed the applicability to many, separate naval architecture and design examples. The examples in this chapter are the first instance of the use of multislice and multiplex network structures and community detection methods in not only naval design, but in physical systems in general. Additionally, this chapter introduced the time-dependent multiplex structure and related community detection method for the first time. The next chapter will use the principles validated above and additional network concepts in the process of the conceptual design of the distributed systems of a hypothetical naval combatant.

CHAPTER VI

A Ship Design Example

In this chapter, the network methods demonstrated individually in the previous chapter will be used through the course of hypothetical ship design to demonstrate their value as a new method for evaluating distributed system designs during the course of a ship design process. A nominal warship's general arrangements will be used to create network representations of the ship's passageway, electrical, and firefighting systems. While the general arrangements are complete, the distributed system design is not and has been approximated. The ship is a 108 meter anti-piracy corvette featuring a single main passageway (*Kemink et al.*, 2009) supporting 69 crew members. The general arrangements are included as Figure 6.1. Much of the chapter was presented in *Rigterink et al.* (2013b).

6.1 Individual ship systems

In this section, three ship distributed systems will be represented as networks. These networks will then be analyzed using the network metrics introduced previously in this document. The three distributed systems to be analyzed are the passageway system, the electrical system, and the firefighting system.

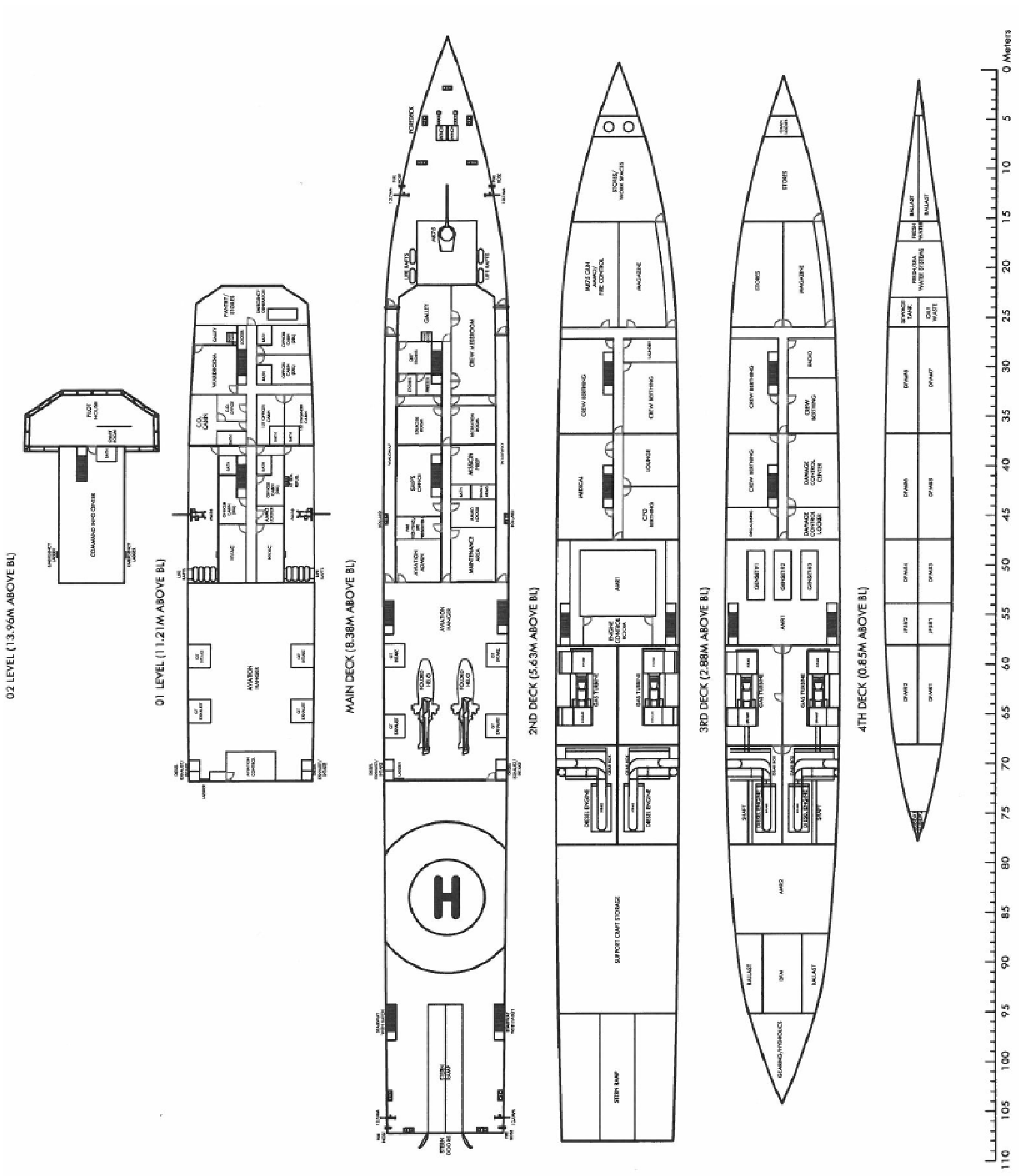


Figure 6.1: The general arrangements of the Cougar-108 (*Kemink et al.*, 2009).

6.1.1 Passage system

The passageway network of a ship is nearly identical to that of the road network of a city; as such, many of the techniques for modeling and analyzing passages have been borrowed from the urban planning and traffic engineering disciplines. Road or passage networks are called spatial networks because nodes represent something in a physical space and edges are equipped with some sort of physical metric. Typically the space is a two-dimensional space and the metric is euclidean distance (*Barthélemy*, 2011).

There are two ways of representing such systems as a network: either passage segments are denoted as edges and intersections are nodes or the reverse where the intersections are treated as edges connecting the nodes representing roads. For this analysis, the former representation where the passage segments are edges and the intersections of the segments are nodes was used as it is a more intuitive structure when dealing with personnel movement and evacuation. A node is placed at each doorway from a space or stairwell as well as anywhere that two passages intersect, nodes are then connected by edges representing the connecting passages. Edge weights are equal to the distance, in meters, between the intersections. The 02 level was not included in the passageway network as it only had two functional spaces.

Figure 6.2 displays a small section of the ship with the passage nodes and edges shown which leads to the creation of the network shown in Figure 6.3. This representation was also chosen because it can easily be adapted into a discrete event simulation (*Teknomo and Fernandez*, 2012) where the movement of agents throughout the passage system can be visualized. The goal of this work is not simulation but rather an easy-to-use metric for comparing multiple designs. Table 6.1 displays some key properties of the passage system networks of the example ship.

It is not possible to tell if a network is sparse or dense by only considering one network

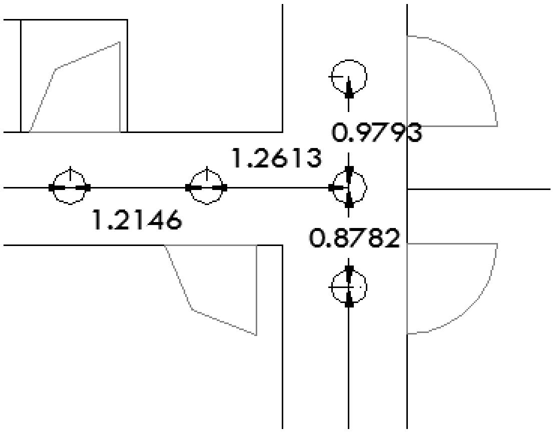


Figure 6.2: Passageway network abstraction.

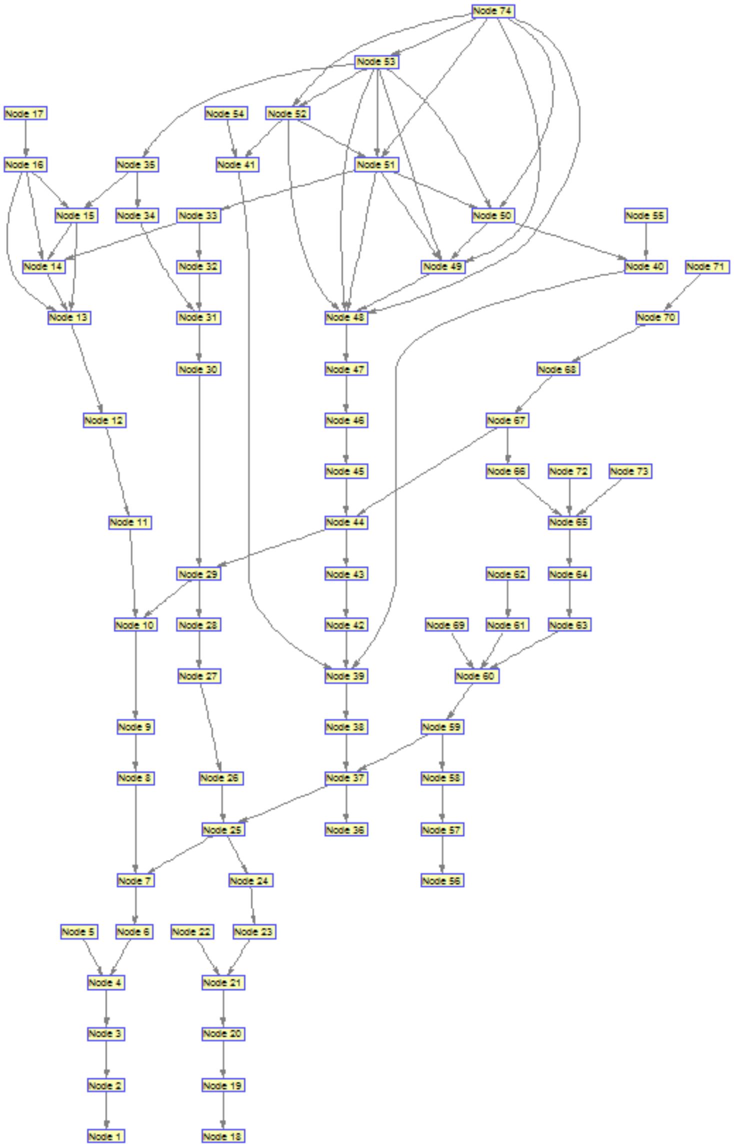


Figure 6.3: Passageway network structure.

Table 6.1: Characteristics of the ship passage network.

Network Property	Value
Nodes	74
Edges	194
Average Degree	2.62
Density	0.0359
Diameter	70.6
No. of Communities	11
Quality	0.802

on its own, rather one must calculate ρ as the number of nodes goes to infinity for a network type. A cursory study of other ship passageway networks shows that the average degree stays roughly the same while the number of nodes increases meaning the density decreases. This leads to the belief that ship passageway networks can be considered sparse. This is beneficial because it reduces the run time of network algorithms and models (*Newman*, 2010). This sparsity of the network also leads to a much higher quality of modularity (Q=0.802) for the passageway next than what was found for the distributed systems in previous sections. Due to the relative lack of edges in the network, splitting the network into many subgraphs without edges running between the subgraphs is much cleaner than in higher density networks.

6.1.1.1 Personnel movement analysis

The ease of personnel movement of paramount to the successful operation of a naval vessel. In the course of operations, many different large-scale personnel movement scenarios occur including the call to general quarters or the dispatching of damage control parties throughout the ship. Less likely scenarios, such as abandon ship events, also need to be planned for. Analyses of these events are typically conducted using discrete event simulations. Simulations are often incongruent with early stage design due to both their high information requirements and time consuming nature. Rather than attempt to model every possible scenario using a discrete event simulation, a network-based approach that instead gives information about personnel movement trends would be preferable for early stage design analysis. A series of these network methods, with a focus on evacuation, will be introduced in this subsection.

When planning evacuation routes, the first thing to look for is the shortest path from any space to an evacuation point. This is achieved by using the shortest path principle. Of the 60 spaces on the example ship, the furthest space from an evacuation point is the forward storeroom on the 3rd deck which is 36.6m. Assuming a slower than average walking speed of 1 m/s, a crew member could reach the nearest egress point in 36.6 seconds which is within norms (*International Maritime Organization*, 2007). Berthing spaces were a maximum of 22.5m from an evacuation point.

Betweenness centrality is a useful metric when looking at the overall flow of the personnel through a passageway system (*Altshuler et al.*, 2011). It is hypothesized that ladders that connect the main deck to other decks will have the highest betweenness as they are the only thing connecting decks to each other. This is confirmed as true. Unsurprisingly, passageways at the fore and aft ends of the ship have the lowest betweenness score out of all the passageways. If functional spaces were included in the network, they would always have a zero betweenness as they are “dead end” nodes that connect only to a passageway node.

While betweenness can give an overall picture of where choke-points could occur if personnel moved about the ship randomly, it does not give as much of an understanding of what could happen in real events such as a catastrophic damage to the ship causing it to need to be evacuated. To this end, the concept of a “goal node” was explored. In the evacuation example, a goal node would be an egress point. In the example of a computer program network, the final returned output of a function could be considered a goal node. To easily analyze choke-points in the probable flow of people or information in either example, a new network measure termed “goal betweenness", e_{ij}, was created (Equation 6.1). This metric is found using the product of the shortest path from the node of interest, i to the goal node j, defined as l_{ij} and the number of shortest paths connecting all other nodes to the goal node that pass through the node of interest, n^i_{kj}. This metric can also be non-dimensionalized using the network diameter, D (Equation 6.2), which is useful when comparing networks representing different systems. This metric could be further tailored to specific cases by adding an exponent λ to the distance term as shown in Equation 6.3. If the effect of distance was to be discounted compared to the effects of the many paths running through a node then $\lambda < 1$, if the effect of distance was considered more important than the number of paths then $\lambda > 1$. For the following analysis $\lambda = 1$, therefore Equation 6.1 is used.

$$e_{ij} = l_{ij} \sum_k n^i_{kj} \tag{6.1}$$

$$e_{ij,D} = \frac{l_{ij}}{D} \sum_k n^i_{kj} \tag{6.2}$$

$$e_{ij,\lambda} = (l_{ij})^{\lambda} \sum_k n^i_{kj} \tag{6.3}$$

In the ship evacuation example, if a node was 16 meters from an exit and five other shortest paths to that exit also came through that node then the evacuation betweenness score is 80. This score can then be analyzed by a designer to find any chokepoints that may be unacceptably far from an egress point.

The node with the highest goal betweenness on the example vessel was the entrance to the forward stairway on the second deck. There are sixteen escape paths that come through this node and it is 14.1m from an evacuation point resulting in a goal betweenness score of 225. Looking over the general arrangement, it can be seen that this node is in the escape route for four crew berths and therefore it is very likely to get congested. To rectify this problem, a crew berth could be moved, a more in-depth evacuation plan could be created to reroute some of the crew members, or a wide stairway could be used to minimize congestion.

6.1.1.2 Passageway system communities

For the community detection algorithm, it was necessary to take the inverse of each element of the adjacency matrix. This was done so that nodes that were closer to each other, which had lower value for the edge weights connecting them, appeared to have stronger community connections than those nodes which were spread further apart. Elements in the matrix that were initially 0, representing a lack of connection between two nodes, were kept as 0.

The quality function always returns values less than or equal to 1, so the score of 0.802 for the passageway system is quite high, meaning the communities are well defined. The eleven groups were split in an intuitive fashion according to the proximity to a stairway. The groupings are shown using an inboard profile in Figure 6.4, where the shades of gray serve only to denote the extent of each community. These groupings could be used for evacuation simulations where instead of having to simulate many individual agents moving through the entire passage system, less agents representing a group of crew members could move through a simplified network based on these passage system groups. This would significantly reduce the cost and time of the simulations, making them more useful in the early design stage. Additionally, the groups could be used for the placement of distributed systems hubs, fan rooms, and damage control lockers as well as production decisions like where to split blocks and modules.

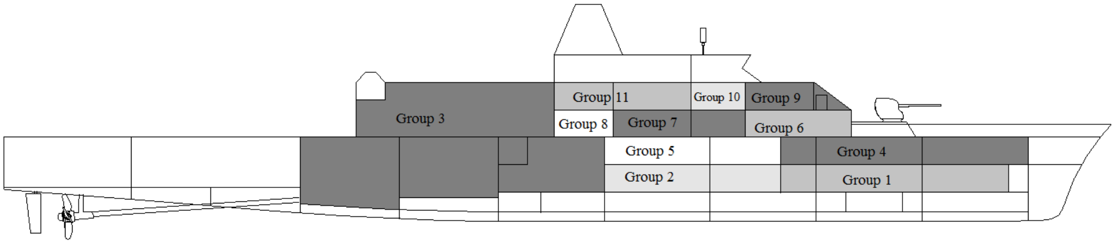
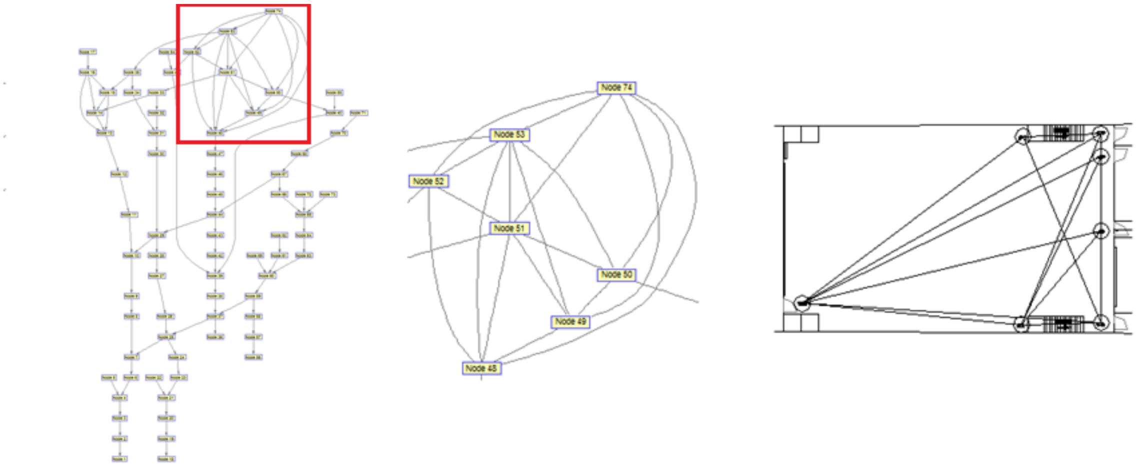


Figure 6.4: Passageway communities for example ship.

6.1.1.3 Passageway system complexity analysis

Using the eleven communities found in the previous section it was found that one community contained a non-planar K_5 subgraph, as shown in Figure 6.5. This leads to a planarity-complexity value of $P = \frac{1}{11} = 0.091$, meaning the graph is mostly planar. A random graph with the same density as the passageway system has a planarity-complexity score of $P_{rand} = 0$ which means P_{ratio} is undefined. Thus the naval ship passageway network is more complex than what would be expected at random for such a sparse network. This is primarily due to the higher than expected clustering of nodes in certain parts of the network.

The non-planar part of the passageway system network is located in the hangar on the main deck which includes three egress points, two stairways to lower decks, and a doorway into the central passageway on the main deck. Such an area could become congested with sailors in an evacuation scenario and there is potential for confusion as to which is the best avenue for escape. In evacuation scenarios, it is best to minimize the creation of large, confused groups and the potential for "herding behavior" which can lead to further overcrowding and overall slower escape (*Helbing and Johansson*, 2009). Being able to capture the location of the non-planar community is an advantage of this method as it allows the analyst to further study the more complex areas.



(a) (b) (c)

Figure 6.5: The non-planar community of the ship passageway system. Shown as part of the full network (a), in isolation (b), and as the physical system (c).

This situation is easily rectified by modifying the hangar area to force personnel flow in specific directions. A majority of the time, the presence of a helicopter or other aircraft in the hangar would remove many of edges and force personnel to follow straight lines rather than crossing diagonally across the hangar. It would also be possible to force this behavior if the helicopter was not present by installing guides in the form or chains or ropes or simply painted walkways on the ground. Forcing to groups to spread out and follow a non-intuitive escape path has been shown to actually improve evacuation times and reduce the potential for stampeding, trampling, and crushing incidents (*Page*, 2009). The hangar's physical layout with the barrier and the resulting network structure are shown in Figure 6.6.

6.1.2 Electrical system

The electrical system network was created by placing a node in each space, at each passage intersection, and at each passage dead end. Longer passageways also had a node place at their midpoint. Additionally, nodes were placed where major deck

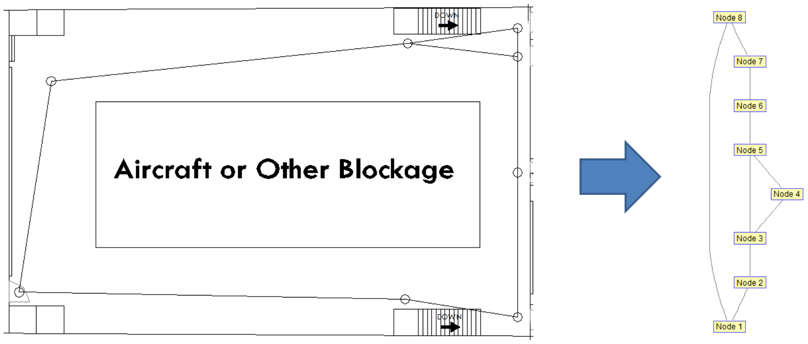


Figure 6.6: Hangar with blockage.

machinery and weapons systems are located. The network assumes three voltage levels which are used for assigning edge weights: a 120v system used for hotel loads, a 240v system for high powered electronics like radars, and a 480v system for machinery. The edge weights are the distance between connected nodes multiplied by a factor corresponding to the voltage of the link to account for different wiring requirements (Table 6.2). The electrical system uses separate adjacency matrices for each deck. The 02 level was once again not considered due to its size. Table 6.3 shows the characteristics for each deck's electrical network. Again, it can be seen that each network has low density, making these networks prime candidates for analyses using network metrics.

Table 6.2: Electrical system edge weight factors.

Voltage	Factor
120v to 120v	1.0
120v to 240v	1.5
120v to 480v	2.0
240v to 240v	2.0
240v to 480v	2.5
480v to 480v	3.0

Table 6.3: Characteristics of example ship electrical networks.

Deck	Nodes	Edges	Ave. Degree	Density
01 Level	40	74	1.85	0.047
Main Deck	35	60	1.71	0.050
Second Deck	38	74	1.95	0.052
Third Deck	38	80	2.11	0.057

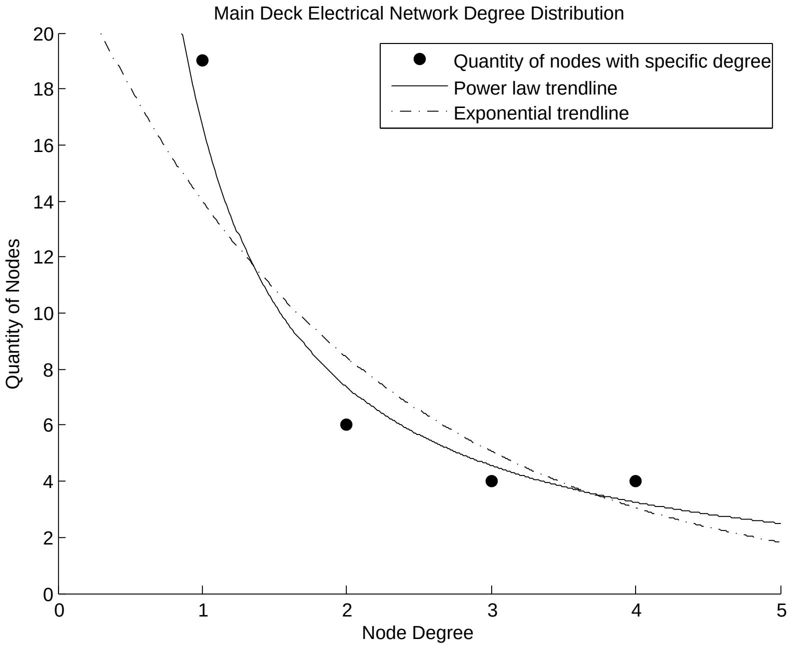
Table 6.4: Electrical system network degree distribution R^2 values.

Deck	Power Law R^2	Exponential R^2
01 Level	0.7622	0.6909
Main Deck	0.9267	0.7924
Second Deck	0.7665	0.6840
Third Deck	0.7454	0.6323

6.1.2.1 Electrical system robustness

The network structure of each deck followed a hub and spoke pattern akin to that of an airline network or the Internet. The degree distribution of the main deck can be seen in Figure 6.7 along with best fit lines created based on a power law equation and an exponential equation. It can be seen that the electrical network degree distribution more closely fits a power law trendline. The R^2 values for each deck's electrical network degree distribution compared to a power law trendline and and exponential trendline are provided in Table 6.4, and they agree with the analysis presented for Figure 6.7.

In one sense this is an attractive feature as scale-free networks are more resilient to random failures. Unfortunately, a targeted attack that can destroy one of the main hubs will be more affective at causing the entire network to fail (*Zhao and Xu*, 2009). Additionally, the hub and spoke structure means the hub nodes have very high betweenness whereas the spoke nodes have low or zero betweenness, meaning not only could the destruction of a hub node render many other nodes inoperable, it could very likely create a discontinuity in the network. With this information it



Trendline Type	Equation	R^2
Power law	$y = 16.685x^{-1.18}$	0.9267
Exponential	$y = 23.27e^{-0.508x}$	0.7924

Figure 6.7: The degree distribution of the main deck electrical network.

is clear that these high betweenness hub nodes should at least be ruggedized and protected more than other nodes, if not designed out completely.

Another method for analyzing the robustness of this system would be by way of a "knockout" experiment. Edges or nodes could be randomly removed and the effect on the connectivity of the network could be studied. Such an experiment would return the same overall results (the fact that center hub nodes are more important to the network as a whole), but take longer to complete, which is why it was deemed unnecessary for this research.

6.1.2.2 Electrical system communities

The groups for each deck were split around these major hubs (Figure 6.8) as is expected. The groups were in the same space across every deck which means switchboards and circuit breakers can be placed in roughly the same location throughout the ship and the ship can be easily split up into electrical zones. The community structure for each deck's network is shown in Table 6.5. This community structure is nearly identical to that of the firefighting system which predisposes the two systems to high levels of interdependency which will reduce the robustness of each system

The electrical system network used for the network is understood to be rather simplified and better approximates a commercial ship's electrical system than that of a military vessel. In this example, it quickly became evident from the power law type degree distribution due to the hub and spoke nature of the system and the similarity of community structure with the firefighting system that the electrical system design was inadequate for the ship's mission.

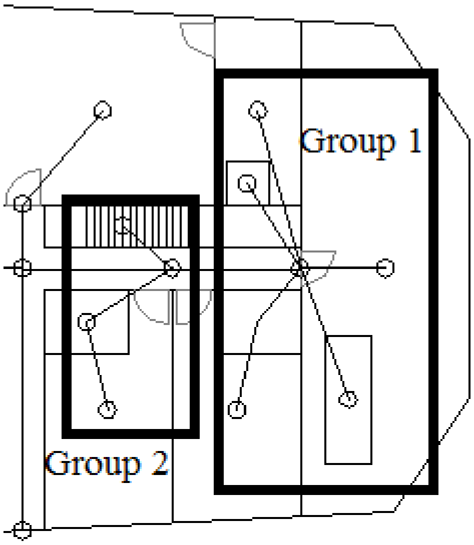


Figure 6.8: 01 level electrical system network structure and groups.

Table 6.5: Electrical system network communities.

Deck	No. of Communities	Quality
01 Level	6	0.749
Main Deck	8	0.739
Second Deck	8	0.722
Third Deck	6	0.688

6.1.3 Firefighting system

The firefighting system network representation used the same nodes and edges as the electrical system with different weighting factors on the edges. The edge weights for the firefighting system are the distances between nodes multiplied by a risk factor. Each node was assigned one of three levels of risk: high, medium, and low, based on a combination of the nodes likelihood of damage and the nodes importance to ship's operation. High risk spaces include engine spaces, damage controls spaces, and magazines. Medium risk spaces include galleys and the steering gear room. Berthing and general storage spaces are considered low risk. Table 6.6 shows the weighting factors for the firefighting system. The firefighting system will be driven by the location of critical mission equipment and major machinery. The initial iteration of the firefighting system design has the same network structure as the electrical system, but with different edge weights. Refer to Table 6.7 for the a summary of the characteristics of each deck's firefighting system network.

Table 6.6: Firefighting system edge weight factors.

Risk	Factor
Low to low	1.0
Low to medium	1.5
Low to high	2.0
Medium to medium	2.0
Medium to high	2.5
High to high	3.0

Table 6.7: Characteristics of example ship firefighting networks.

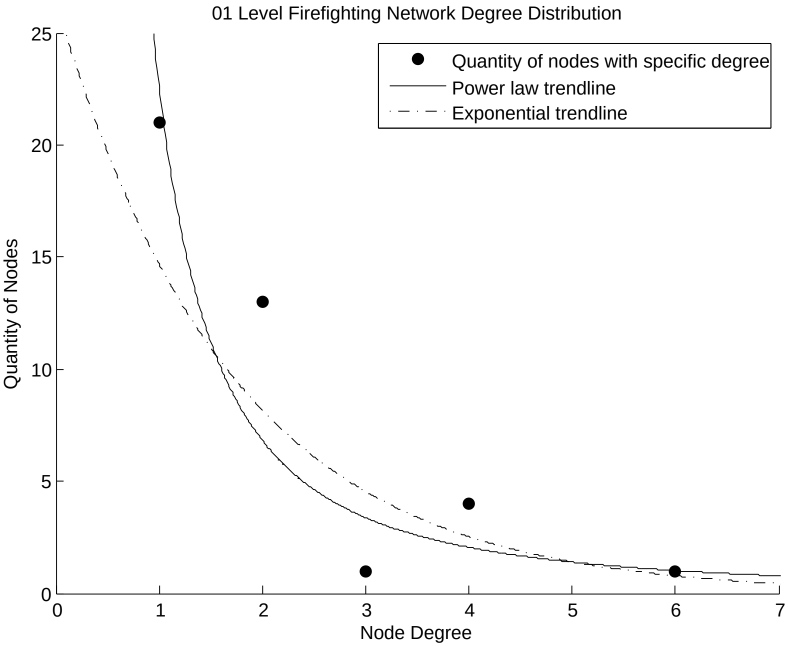
Deck	Nodes	Edges	Ave. Degree	Density
01 Level	40	74	1.85	0.047
Main Deck	35	60	1.71	0.050
Second Deck	38	74	1.95	0.052
Third Deck	38	80	2.11	0.057

6.1.3.1 Firefighting system robustness

Since the firefighting system network is a re-weighting of the electrical system network the number of nodes and edges as well as the average degree and density are the same as those reported in Table 6.3. The betweenness scores for the nodes are also similar: the hub nodes have high betweenness while the spoke nodes have low to zero betweenness. Table 6.8 shows the R^2 values of the degree distributions for each deck's firefighting system network as compared to a power law and exponential trendline. The firefighting system network has the same robustness issues as the electrical system in that it is susceptible to targeted attacks due to its scale-free degree distribution, as is shown by the R^2 values in Table 6.8 being higher for the power law distribution than the exponential distribution. For the 01 Level, the degree distribution is less similar to a power law than it is for the main deck (Figure 6.9), which again can be seen because the R^2 value for the power law distribution for the firefighting system is 0.7056 compared to 0.9267 for the power law distribution for the electrical system (Figure 6.7).

Table 6.8: Firefighting system network degree distribution R^2 values.

Deck	Power Law R^2	Exponential R^2
01 Level	0.7056	0.6359
Main Deck	0.9429	0.8285
Second Deck	0.7665	0.6840
Third Deck	0.6556	0.5583



Trendline Type	Equation	R^2
Power law	$y = 22.638x^{-1.731}$	0.7056
Exponential	$y = 26.392e^{-0.586x}$	0.6359

Figure 6.9: The degree distribution of the 01 Level firefighting network.

6.1.3.2 Firefighting system communities

The electrical system and firefighting system on the 01 Level have same community structure with only a minor difference in quality due to the low voltage spaces on that level also being low fire risk spaces and high voltage spaces being high fire risk spaces. The same can be said about the communities on the second deck. The main deck has the largest difference between communities and quality due the deck machinery and weapons systems outside of the deck house, which do not receive as high of a fire risk factor, almost detaching them from the rest of the main deck's firefighting system. On the third deck, the firefighting system has an extra community because it breaks the port and starboard gas turbine spaces and the aft auxiliary machinery room into three different groups whereas the starboard gas turbine space and aft auxiliary are in the same electrical community. This similarity of communities suggests a high level of interdependence or interconnectedness between the two systems. This idea will be investigated further in Section 7.3. Due to the power law degree distribution and similarity of community structure to the electrical system, the firefighting system also required a redesign.

Table 6.9: Firefighting system network communities.

Deck	No. of Communities	Quality
01 Level	6	0.746
Main Deck	9	0.728
Second Deck	8	0.729
Third Deck	7	0.685

6.2 Ship system design evolution

This section will detail two more iterations of the electrical and firefighting system design and introduce the multislice network structure and multislice community detection as a method for elucidating the similarities and differences between different design iterations.

6.2.1 Electrical and firefighting system redesign

Based on the analyses presented in Sections 7.1.2 and 7.1.3, it is desirable to redesign the example ship's electrical and firefighting systems to increase the robustness of each system to targeted attack. This was done for the second deck only. The first revision attempted to separate the main electrical trunk and main firefighting trunk onto different sides of the ship to reduce interactions while the second revision attempted to add redundancies into both systems.

Table 6.10 shows the community structure of the three different iterations of both the electrical and firefighting systems, taken individually. It can be seen that the first and second iterations of the electrical network have similar numbers of communities and values for quality meaning they are in fact similar systems, with only the main service line moved. The third iteration of the electrical system has fewer communities and lower quality than the other two iterations because of the additional edges (82 in iteration three as opposed to 74 in iteration one, an increase of roughly 10%) added for redundancy. These additional edges further connected the nodes in the network, making dividing the network into rigid communities more difficult. The firefighting system's community structure did not follow the same trend, as adding redundancy by simply adding another pipe connecting sprinkler heads is not feasible.

Table 6.10: Community structure of redesigned system networks.

	Electrical		Firefighting	
Iteration	No. Communities	Quality	No. Communities	Quality
1	8	0.722	8	0.729
2	8	0.742	7	0.788
3	7	0.686	7	0.740

Table 6.11 shows the R^2 values for the degree distribution of all the second deck design iterations when compared to a power law and exponential distribution. The new iterations have degree distributions more similar to an exponential distribution than a power law distribution, which was desired in an effort to increase robustness to target attacks.

Table 6.11: R^2 values for degree distribution of second deck electrical and firefighting networks.

	Electrical		Firefighting	
Iteration	Power Law R^2	Exponential R^2	Power Law R^2	Exponential R^2
1	0.7665	0.6840	0.7665	0.6840
2	0.5053	0.6954	0.1524	0.2742
3	0.6887	0.9082	0.4878	0.6461

The especially low R^2 values for the second iteration of the firefighting system (0.1524 for a power law distribution and 0.2742 for an exponential distribution) simply mean the degree distribution for the iterations was not well described by either distribution.

This is of less concern than the difference between the R^2 values for each distribution, shown in Table 6.12, where it can be seen that over the course of the design evolutions the degree distributions became more like an exponential distribution than a power law distribution. This is shown by the delta between the R^2 values going from -0.0825 for both the electrical and firefighting systems (distribution is more like a power law distribution) to 0.2195 and 0.1583 for the electrical and firefighting systems, respectively (the degree distributions are more like an exponential distribution).

Table 6.12: Difference between exponential and power law R^2 values.

Iteration	Electrical Exponential R^2 - Power Law R^2	Firefighting Exponential R^2 - Power Law R^2
1	-0.0825	-0.0825
2	0.1901	0.1228
3	0.2195	0.1583

For this work, no node locations were changed. If modified node locations were desired, dummy nodes would need to be to be added to each network slice. These nodes would not be connected to any other nodes in slices in which they are not used. The passageway system was not changed.

6.2.2 Design evolution analysis using a multislice network structure

A mutlislice network structure was then created for each system where the inter-slice linkages were sequential, i.e., a node in iteration one was only connected to itself interaction two. Through multislice community detection using Equation 3.7, the similarities and differences between different design iterations can be explored. In the case of a multislice network, $C_{jsr} = \omega$ if, and only if, $s = r \pm 1$, and $C_{jsr} = 0$ otherwise. This enforces the sequential nature of the mutli-slice network structure. By varying the strength of the interslice connections, ω, a designer can determine how different one design iteration is from another.

First, every combination of two of the three design iterations were placed into a two slice multislice structure in an effort to determine how different each design was from the other designs of the same system. Tables 6.13 and 6.14 show the ω values where the two tested designs decoupled (when at least one node was in the a different community between network slices). Table 6.13 contains shows these ω values for the electrical system and Table 6.14 shows these ω values for the firefighting system. The quality of modularity is also included. The resolution factor was kept constant at $\gamma = 1$.

Table 6.13: Electrical system design iteration decoupling ω values.

Combination	$\omega_{decouple}$	Quality
One and Two	0.174	0.798
One and Three	0.174	0.750
Two and Three	0.129	0.756

Table 6.14: Firefighting system design iteration decoupling ω values.

Combination	$\omega_{decouple}$	Quality
One and Two	0.206	0.785
One and Three	0.129	0.772
Two and Three	0.124	0.769

For this analysis, the greater the value of $\omega_{decouple}$ the less similar the two tested designs. By this measure, the first and second iteration of the firefighting system design are the most different, which is supported by the widely different R^2 values for their degree distributions. This also agrees with the intent of the redesign stated previously. The second and third iterations of both the electrical and firefighting systems were the most similar designs of their respective systems, which also reaffirms that the goals of the redesign were met. The third design iteration of the respective systems sought solely to add redundancy to the second design iteration, meaning a majority of the network structure was intentionally left identical between the two networks, which the analysis also identified.

While knowing at what level of interaction different designs become decoupled for one node is useful, it is also important to know how different the designs are after in a more holistic sense. In Figures 6.10 and 6.11, the number of nodes that are in different communities for 20 values of ω ranging from 0.01 to 0.20 are shown along with the quality of the network communities. For all combinations in both systems the expected trends arise: the percentage of nodes in different communities between slices generally decreases and the quality of modularity increases as the strength of the interslice connections, ω, increases.

Additionally, it can be seen that the first and second iteration of the electrical system design are very different as evidenced by the long, nearly flat line between $\omega = .03$ and $\omega = 0.2$. The other two combinations for the electrical system designs show much closer similarity. This coincides with the prior analyses. Alternatively, the firefighting system combinations show a much larger amount of difference between all the design iterations, which is shown by the nearly steady increase in the number of nodes in two communities as ω decreases.

The previous analyses were not based on a true multislice structure because there were only two slices for each structure. In the next set of analyses, two three-slice multislice structures will be created, one for each the electrical and firefighting system. The slices will be in the order that the design iterations were created. Community detection will again be used, but this time in an effort to discern the design drivers for each system. In this analysis ω is reduced until a node is in different communities between two slices. The higher the value of ω when this occurs, the higher the variation between design iterations. In Table 6.15, the multislice community structure of both the electrical and firefighting systems are shown along with the value of ω that first causes a node to be a member of different communities in different network slices.

It can be inferred from Table 6.15 that the firefighting system has more variation between each design iteration than the electrical system because of the higher value

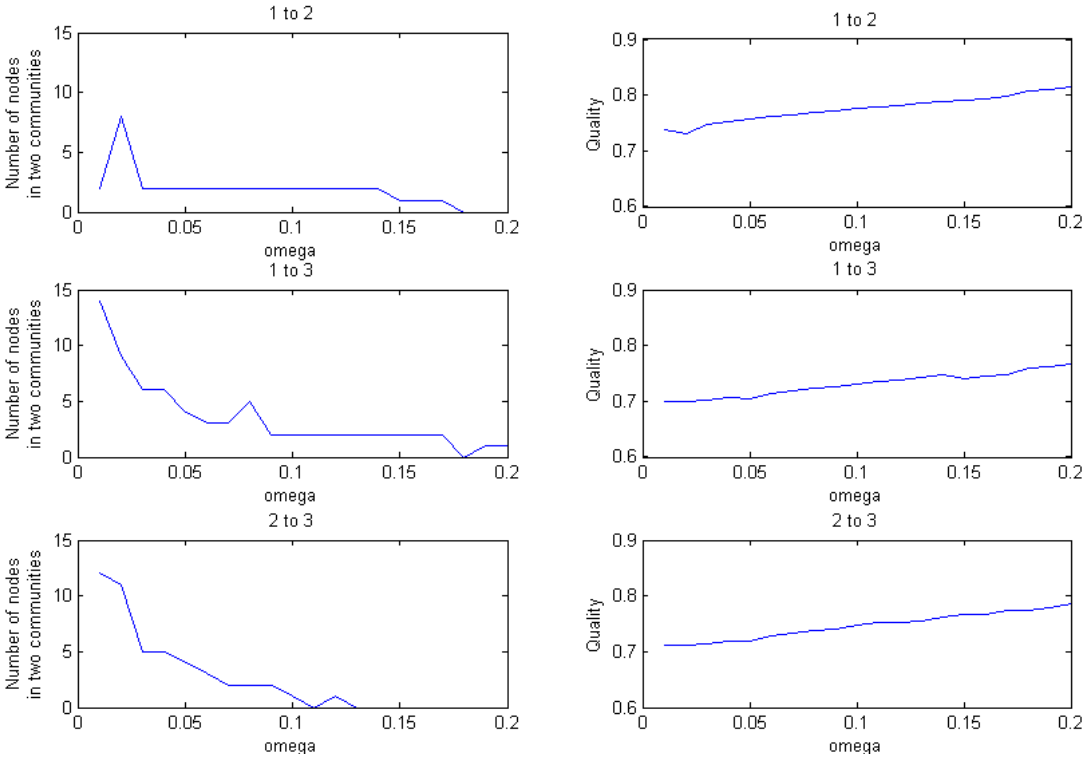


Figure 6.10: Difference between design iterations of electrical system

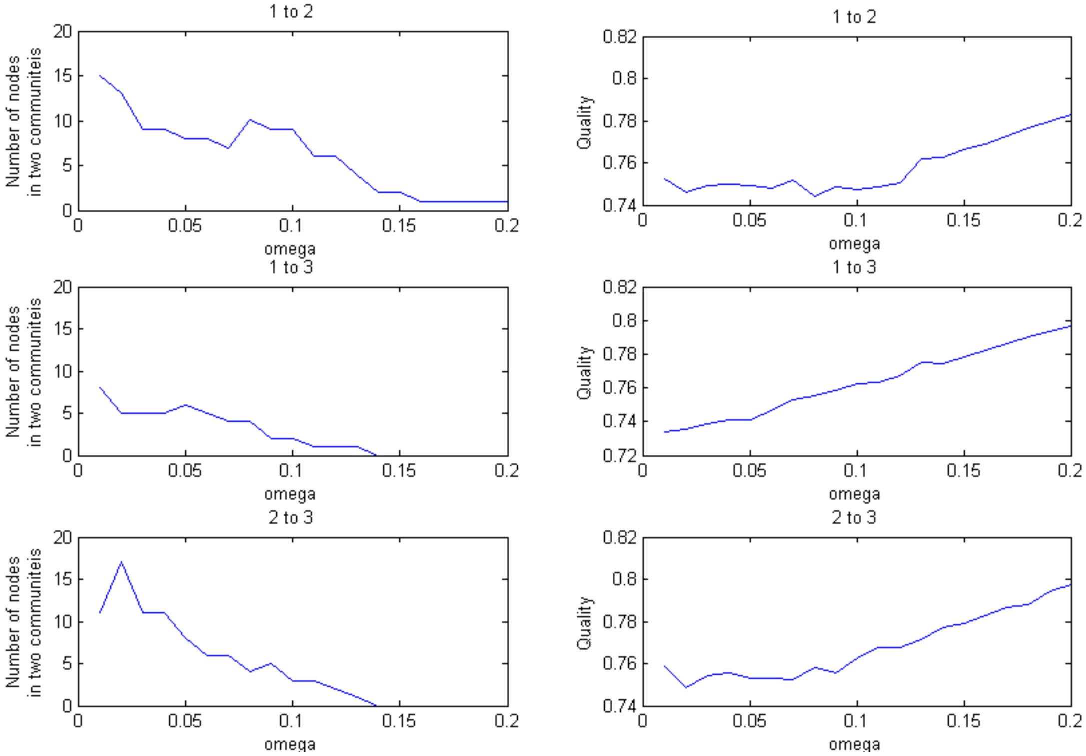


Figure 6.11: Difference between design iterations of firefighting system

Table 6.15: Multislice network communities.

System	ω	No. of Communities	Quality
Electrical	0.182	8	0.794
Firefighting	0.199	7	0.807

of ω. This means the communities within a slice are more well defined than those created using interslice links which is also supported by the higher value for the quality of modularity. It is interesting to note in both cases the node that split into different communities in different designs was in the same community for the second and third iteration for both systems, though it was not the same node. This shows that the second two iterations may be more similar than was originally thought. The value of ω had to be lowered to 0.160 and 0.130, for the electrical and firefighting systems respectively, to separate a node into two communities for the second and third design iterations. In this case, the electrical system's second and third design iterations were more different than the firefighting system's, which is also expected as the third iteration for the electrical system had many more edges than the previous iteration which made the community memberships much weaker.

6.3 Aggregation of ship systems

In this example, there are three interaction networks, the previously discussed electrical system and fire fighting systems as well as a modified passageway network that uses the same nodes as the other two networks. When $\omega = 0$, meaning there was no interslice connection, each individual was placed in three communities, one community for each network slice. For $\omega \geq 1$, each individual was placed in one community which was the same across all three networks in the multiplex. This caused the number of communities on each deck to decrease compared to the number of communities found when each network was analyzed individually, while the quality of the communities increased, suggesting a much more defined community structures. The number of communities and quality of modularity for each deck can be seen in Table 6.16. This decrease in communities and increase in quality is expected and caused by overweighting the all-to-all multiplex connections. The highest edge weights in the individual networks is less than one (once the distances have been inverted), so having an inter-slice connection of one or higher will always lead to community structures where each individual is only in one community that bridges all network slices.

Table 6.16: Multiplex network communities.

Deck	No. of Communities	Quality
01 Level	6	0.930
Main Deck	6	0.946
Second Deck	7	0.954
Third Deck	7	0.953

6.3.1 First design iteration

By varying ω between 0 and 1, more information about the level of interdependency and interconnectedness of the multiplex networks can be found. In Table, 6.17 the first value of ω that caused the same individual to be in different communities in different networks is shown, along with the number of communities and their quality. The values of ω can be interpreted as an interdependency metric with lower values of meaning there is higher interdependency across the network slices. In Table 6.17, it can be seen that the Second Deck has the most inter-system coupling whereas the Main Deck has the least. It can also be seen that the quality of modularity decreased as ω decreased, which is expected because the strength of the interslice connections is reduced. It should also be noted that the passageway instance of each node was the typically the node that would go into a separate group than the other two instances of that node, meaning the electrical and firefighting systems are still highly coupled. For the Third Deck, there was no value of ω other than 0 caused the a node to be placed in different communities. The resolution factor was set as $\gamma = 1$.

Table 6.17: System coupling factor for each deck.

Deck	ω	No. of Communities	Quality
01 Level	0.111	6	0.783
Main Deck	0.202	6	0.843
Second Deck	0.040	8	0.755
Third Deck	−−	−−	−−

This process will be repeated for the second and third design iteration to determine if the goal of reducing system interconnectedness was achieved.

6.3.2 Second and third design iteration

As mentioned previously, the first revision attempted to separate the main electrical trunk and main firefighting trunk onto different sides of the ship to reduce interactions. Table 6.17 shows the first value of ω that caused the same individual to be in different communities in different networks, along with the number of communities and their quality, for all three design iterations of the second deck.

Table 6.18: Second deck system coupling factor for each design iteration.

Design Iteration	ω	No. of Communities	Quality
1st	0.040	8	0.755
2nd	0.131	7	0.780
3rd	0.122	7	0.783

Of the three iterations, the second design has the least interdependencies between the three networks which is in agreement with the goals of the redesign. The third iteration inevitably has tighter interactions than the second iteration due to the added redundancy which leads to an increased number of edges within each network slice. Interestingly, the quality of modularity increases in each iteration, meaning the communities are more well defined, which is contrary to what was expected. It was expected that the communities would be less well defined as the interactions decreased because nodes would have stonger ties to multiple potential communities. This is explained by the decrease in the number of communities from 8 to 7 between the first and second design iteration. The reduction in the number of communities caused the communities to grow, thus encompassing more of the potential links. When ω is reduced to 0.103 and 0.020 for the second and third iteration respectively, 8 communities are found and the quality has been reduced to 0.775 and 0.708, respectively.

6.4 System design interaction evolution

After analyzing static multiplex networks and time dependent simplex networks, the next step is to combine the two into a time-dependent multiplex structure, like was shown in Figure 5.17. By analyzing the communities for this type of structure, the designer can see how the entirety of the ship systems design evolves including the interactions between systems. Additionally, communities that form out of the same nodes throughout time could signal a part of the system that has matured or is no longer being changed. To discern these communities, a new modularity equation (Equation 5.1) is necessary as another linkage term is required. This equation is described in Chapter V.

A time-dependent multiplex network was created using the three iterations of the electrical and firefighting systems and the passageway system (which was not changed). Rather than trying to tune both the interslice (ω) and interplex (ζ) connection terms to find the first instance of a node being in different communities in different slices/plexes, the values of ω and ζ were varied to explore trends in community structures. Table 6.19 shows the tested combinations and the fraction of nodes that were in multiple communities, Table 6.20 shows the number of communities for each combination, and Table 6.21 shows the quality of modularity of the communities for each combination.

As was expected, when either connection weight was set to 0 all of the nodes were split into different communities. When $\omega = 0$, the community detection algorithm saw three different multislice networks and when $\zeta = 0$ the community detection algorithm saw three different multiplex structures. When both ω and $\zeta = 0$, each network was analyzed separately. As the two factors were increased, the percentage of nodes in multiple communities decreased, as did the number of communities, while the quality of communities increased, agreeing with the trends found in the earlier sections. It should also be noted that the effects of changing ω and ζ are not symmetric. Nodes are more tightly coupled in the time domain than they are in the spatial domain, which makes sense from an evolutionary design perspective as there is only so much that can be changed from one iteration to another, especially in a small design such as the one presented here, where no node locations were changed.

A closer investigation of the case where ω and $\zeta = 0.05$ revealed that 14 of the 38 nodes were grouped into multiple communities. Of these 14 cases, only two nodes were grouped into more than two communities. These two spaces are the auxiliary machinery room and engine control room which were crucial factors in the design of the electrical and firefighting system as they are integral spaces for the operation of the ship. It is interesting to note that typically the engine control room is not considered in early stage design because it is relatively small and the equipment in the space is typically easy to install. However, when the amount of connections required to make the engine control room functional are taken into account, it becomes clear that it is a very important space from the systems perspective. The use of time-dependent multiplex community detection reveal this fact. This proves the usefulness of the time-dependent multiplex structure for analyzing system interactions over time and elucidating design drivers of interdependent and interconnected systems.

Table 6.19: Fraction of nodes grouped into different communities for different combinations of ω and ζ.

		ζ						
		0.00	0.01	0.05	0.10	0.25	0.50	1.00
	0.00	1.000	1.000	1.000	1.000	1.000	1.000	1.000
	0.01	1.000	0.500	0.500	0.342	0.263	0.184	0.289
	0.05	1.000	0.553	0.368	0.237	0.053	0.000	0.000
ω	0.1	1.000	0.316	0.211	0.105	0.026	0.000	0.000
	0.25	1.000	0.289	0.053	0.053	0.000	0.000	0.000
	0.50	1.000	0.289	0.053	0.053	0.000	0.000	0.000
	1.00	1.000	0.237	0.053	0.053	0.000	0.000	0.000

Table 6.20: Number of communities for different combinations of ω and ζ.

		ζ						
		0.00	0.01	0.05	0.10	0.25	0.50	1.00
	0.00	103	35	36	34	34	34	34
	0.01	25	9	9	7	7	7	7
	0.05	22	8	8	7	7	7	7
ω	0.1	22	7	7	7	7	7	7
	0.25	21	8	7	7	7	7	7
	0.50	22	8	7	7	7	7	7
	1.00	21	8	7	7	7	7	7

Table 6.21: Quality of modularity for different combinations of ω and ζ.

		ζ						
		0.00	0.01	0.05	0.10	0.25	0.50	1.00
	0.00	0.717	0.720	0.738	0.763	0.827	0.883	0.928
	0.01	0.717	0.721	0.737	0.764	0.825	0.881	0.927
	0.05	0.747	0.746	0.751	0.779	0.836	0.887	0.928
ω	0.1	0.783	0.783	0.787	0.800	0.855	0.897	0.933
	0.25	0.860	0.859	0.861	0.866	0.889	0.917	0.942
	0.50	0.910	0.909	0.910	0.910	0.925	0.937	0.953
	1.00	0.948	0.947	0.947	0.946	0.953	0.958	0.965

6.5 Ship design process conclusions

This section introduces a new set of network-based methods that can be used to better understand the implications of distributed systems design decisions. Starting with individual systems, a series of new methods for determine robustness without the need for simulations were presented. Next, the evolution of these system designs was tracked using a multislice network structure, which allows a designer to understand the implications of changes between design iterations. Then the systems were analyzed together in a multiplex network structure, allowing a designer to understand the interdependencies between multiple systems that would not be evident in a visual analysis. Lastly, the novel time-dependent multiplex network structure was used to identify design drivers and analyze the evolution of interactions between systems as a ship design progresses. These novel methods do not require complex 3D CAD models or simulations. Therefore, they can be used by a single naval architect to gain insight into the implications of design decisions in the earliest design stages which should result in improved naval distributed systems designs.

CHAPTER VII

Conclusions

This final chapter is divided into two parts: the first reiterates the novel contributions of this research and the work done to support those claim; and the second part presents potential areas for future research in network-centric ship systems research and design.

7.1 Review of intellectual contributions

The goal of this research was to create a new method for bring ship design information that is traditionally only available in the latest design stages as early in the design process as possible. This dissertation has shown the value of network theory and its principles to naval design, as well as developed novel techniques and methods to extend network theory for the study of naval distributed system design. The previous chapters presented results that are a combination of novel applications of network concepts to distributed systems design along with development and application of new network methods and structures.

The methods introduced in this dissertation are valuable not only on there own. When used together they provide a series of uncorrelated models that can aid a designer in better understanding the big picture of a design. When exploring a large design space it is important to have multiple models and sensors negate the effect of biases and preconceptions.

The impetus for this was the recognition of the need to shift from the 3D CAD paradigm of creating and analyzing distributed systems design in the later design stages and focusing on methods that can be used in the early design stage to bridge the gap between rules of thumb and detailed geometric models. Chapter 2 begins with a discussion of the current process of ship design and how the naval architecture and marine engineering realms are considered two separate disciplines in the early stage and then are combined together in 3D models late in the design when modifications are most costly. This chapter exposed the need for a new system that was less costly, both computationally and monetarily, and that would allow naval architects and systems design to work more closely and easily share ideas as the design process progresses.

Also important to this research was the recognition of the inherit difference between distributed systems and other ship systems, which requires a new medium for storing, displaying, and analyzing distributed systems information. The latter portion of Chapter 2 surveyed the tools used for early stage design for non-distributed systems and found that they were not applicable for distributed systems design and a new way of modeling and analyzing distributed systems in the early design stage was necessary. From this realization came the study of complex systems theory and network theory and the realization that it was well suited to distributed systems design. While network theory was well suited for this role as shown in Chapter 3, there were many gaps between the two disciplines that needed to be bridged. The novel contributions bridging these gaps are presented in the remainder of this section.

7.1.1 Contributions to single system analysis

Creating a new network complexity measure (Equation 7.1) based on the network concepts of planarity and communities. Chapter 4 introduces a complexity metric designed specifically for dealing with planar or near-planar networks, the class to which most ship systems networks belong. This metric fills a void in analysis of planar networks, a topic that is usually ignored in network theory.

$$P = \frac{K}{C} \tag{7.1}$$

Identifying and applying network metrics for determining potential choke-points within a ship's passageway system. Rather than relying on discrete event simulations, the network concepts of shortest paths and betweenness centrality were shown to be useful for determine choke-points within a passageway system. Additionally, the complexity metric introduced in Chapter 4 was used to identify areas of potential confusion during an evacuation scenario.

Creating a new family of betweenness measures, "goal betweenness," (Equation 7.2) which added the concept of a goal node to previously developed betweenness measures. This new metric, introduced in Section 6.1.1.1, adds another level of fidelity to betweenness centrality. Instead of considering all possible information traffic across a network, the traffic is given a goal node and choke-points and their distances from this goal are calculated. The relative importance of the distance component and the path components can be tuned by changing the exponent λ. This metric was used to analyze the passageway system of a ship and located a potential bottleneck far from an egress point, which may call for a redesign.

$$e_{ij,\lambda} = (l_{ij})^{\lambda} \sum_{k} n^{i}_{kj} \tag{7.2}$$

Identifying and applying network methods for analyzing system robustness. The network concept of degree distribution was leveraged for calculating the robustness to targeted attack and random failure of hypothetical ship systems and used to redesign the systems in an effort to make them more robust to targeted attack.

Abstracting additional system complexity information into the edge weights of distributed system networks. Rather than relying solely on euclidean distance to weight connections between nodes, the complexity of the edges connecting the nodes was accounted for using the voltage and fire risk factors (Sections 6.1.2 and 6.1.3, respectively).

Grouping physical system communities into communities and using the community structures as a predictors of system interactions. The community structures for the electrical and firefighting systems were compared and found to be very similar, and it was predicted that they would have a high level of interdependence. This prediction was later confirmed using multiplex community detection.

7.1.2 Contributions to multiple system analysis

Realizing that multiplex and multislice network structures are applicable to more than just social networks. Prior to the dissertation, mutliplex and multislice systems had been used solely for analyzing social networks, this dissertation proved their applicability to physical networks.

Applying multiple resolution, multislice analysis to analyze change prop-

agation in a design process. In Section 5.2.1, multiple resolution, multislice community detection was used verify that changes later in a design have a greater effect on the entirety of the design than changes made earlier.

Verifying the advantages of zonal power distribution systems over radial power distribution systems. By analyzing both their individual structures (Section 5.1.4) and how the community structure changes in a multiple resolution, multislice community analysis (Section 5.2.2), it was shown that the zonal power distribution system is more robust to both random failures and targeted attacks and more adept at handling power demand spikes caused by damage or transient loads.

Tracking distributed systems design evolution using a multislice network structure. Section 6.2.2 introduced the multislice community detection as a method for determining the amount of change between sequential design iterations. The interslice weighting factor ω was introduced as a metric for the degree of change between iterations.

Analyzing distributed systems interactions using a multiplex network structure. The interslice connection weight factor ω was introduced as a metric (Section 5.3.2 and Section 6.3) for determining the degree of interconnected between multiple networks. Previously, ω had just been varied to see how community structure changed, but this dissertation associated it with physical property.

Creating the time-dependent multiplex network structure and using it to analyze the evolution of distributed system interactions over sequential iterations. The multiplex and multislice network structures were combined together to create the time-dependent multiplex structure. Community detection, done using Equation 7.3, on this new network structure used a tool for determining how system interactions changed over time. This tool was used to investigate the case of the USS

Yorktown losing propulsion power and the changes in system interactions over the course of an example ship design.

$$Q = \frac{1}{2\mu} \sum_{ijsrxy} \left\{ \left(A_{ijxy} - \gamma_{sx} \frac{k_{isx} k_{jsx}}{2m_{sx}} \right) \delta_{sr} \delta_{xy} + \delta_{ij} \delta_{xy} C_{jxsr} + \delta_{ij} \delta_{sr} E_{jsxy} \right\} \delta(g_{isx}, g_{jry}) \tag{7.3}$$

7.2 Future topics of interest

Expanding upon the planarity-complexity metric. It is believed the planarity-complexity metric can be expanded to assess the planarity of the secondary network created after community detection. Once the network is split into communities, these communities can then be recast as nodes and an edge can be create between these community-nodes, if an edge exists between nodes in different communities. This edge could be weighted based on the number of edges connecting the communities or left unweighted. The overall planarity of the new network could then be analyzed to determine the global network complexity.

Additionally, a betweenness analysis could be used to determine if the local non-planar community existed on the periphery of the total network, and therefore the complexity would be less of an issue; or, if the complex community was a central part of the network which would have a far greater effect on network complexity.

Using community detection to create reduced order personnel movement models. As was shown in Section 6.1, the passageway system of a ship can be reduced from a large number of nodes to a smaller number of communities. This smaller community network could then be used as a proxy of the larger model for discrete event simulations to analyze personnel movement (*Rigterink et al.*, 2014), as shown in Figure 7.1. Discrete event simulation computation times increase super-linearly dependent on their size (*Fishman*, 2001), so any decrease in the size of the model makes simulation more attractive for early stage design. The reduced order model could be useful relative to other reduced order models of alternative designs.

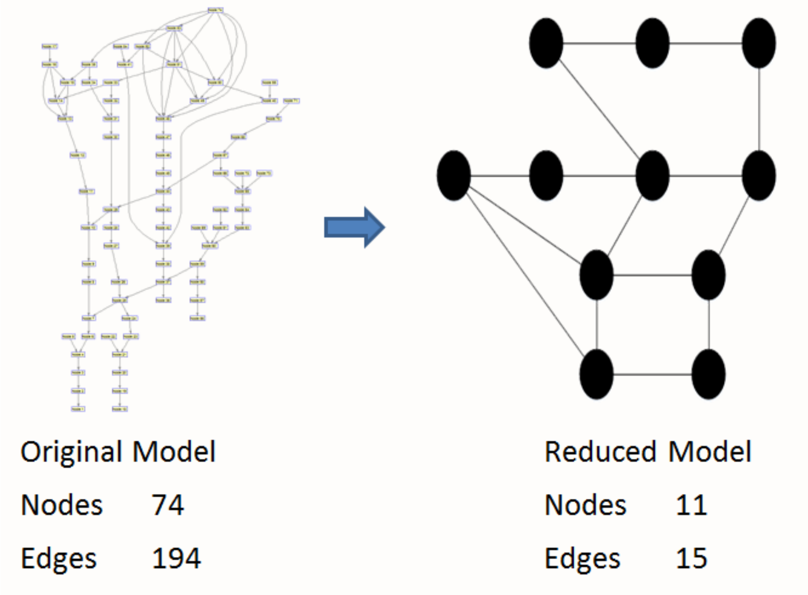
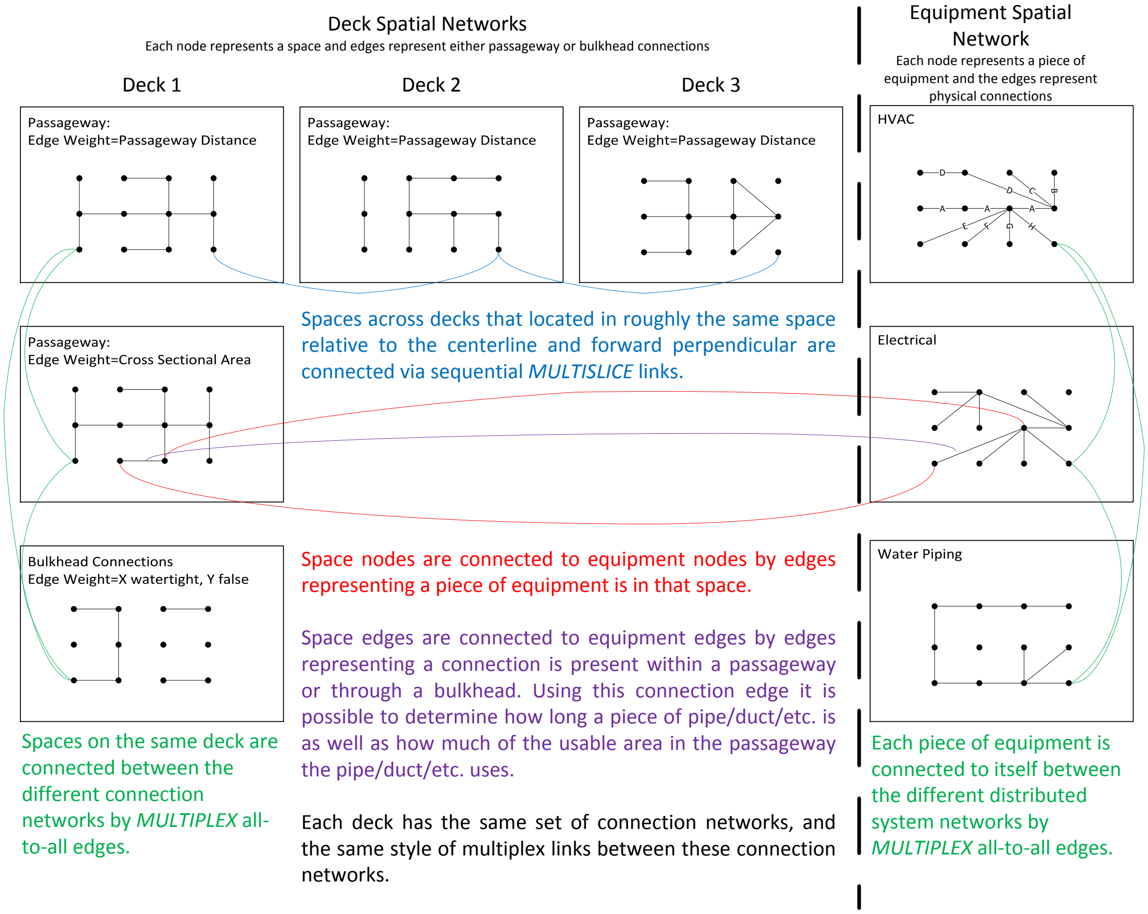


Figure 7.1: The reduction of passageway network size using community detection.

Applying percolation to personnel movement models. Percolation models could be applied to distributed systems networks in an effort to model hazards or damage. Percolation is an attractive alternative to discrete event simulation because it substantially faster. While percolation does not give insights into the results of specific events, it does reveal the patterns and average response of a system to various perturbations. This is desirable at the early design stage because specific conditions and damage probabilities are unknown, so it is more important to understand the reaction of the system to any and all scenarios.

Creating a hybrid multiplex/multislice model for sizing ship distributed systems. In this dissertation, the focus was on individual system robustness and the interactions between systems, no effort was expended on sizing systems. It is believed that network structure utilizing multiplex and multislice concepts like the one presented in Figure 7.2 could be useful for solving this problem.



Spaces across decks that located in roughly the same space relative to the centerline and forward perpendicular are connected via sequential *MULTISLICE* links.

Space nodes are connected to equipment nodes by edges representing a piece of equipment is in that space.

Space edges are connected to equipment edges by edges representing a connection is present within a passageway or through a bulkhead. Using this connection edge it is possible to determine how long a piece of pipe/duct/etc. is as well as how much of the usable area in the passageway the pipe/duct/etc. uses.

Each deck has the same set of connection networks, and the same style of multiplex links between these connection networks.

Spaces on the same deck are connected between the different connection networks by *MULTIPLEX* all-to-all edges.

Each piece of equipment is connected to itself between the different distributed system networks by *MULTIPLEX* all-to-all edges.

Figure 7.2: A schematic of a potential hybrid multiplex/multislice model for distributed system sizing (*Rigterink and Singer*, 2014b).

Developing a process for creating and using the “informational dual” of a network distributed system network for comparing different types of systems. *Rosvall et al.* (2005) introduced the concept of an ”informational dual” for planar street networks where named roads are represented as nodes and intersections between roads are edges (Figure 7.3) rather than the opposite where road segments are edges and intersections are nodes (the representation used as the basis for the passageway system network analyzed in Section 6.1.1). This dual representation stores information the same way humans travel on roads, memorizing which streets to turn on rather than every road segment they must pass through. Storing information in this way reduces the amount of information required to navigate even vast distances.

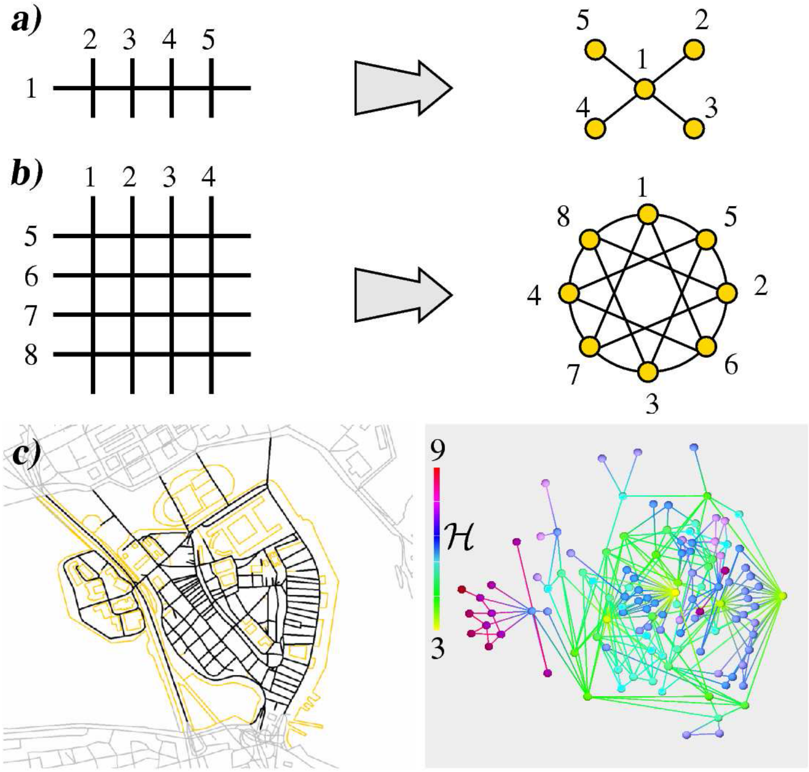


Figure 7.3: The spatial primal and informational dual of various street networks. Examples shown are of a hypothetical small city (a), a hypothetical planned community (b), and “Gamla stan”, a district of Stockholm, Sweden (c).

Masucci et al. (2009) expanded upon this representation and introduced a series of metrics for analyzing the organization and design philosophy of a system based on the informational dual representation of spatial graphs. It is believed that these metrics could be applied to distributed systems design as to either analyze completed designs or possibly to create an optimal informational dual to tailor all systems off of. Additionally, the informational dual representation allows for a comparison of networks representing different systems in the same space, which can be difficult or impossible to achieve otherwise.

There is a need for a method to quickly convert the spatial networks used in this dissertation into informational networks as well as more study into what insights these duals can yield.

* * *

The ideas for future research presented in this section are merely a sampling of the potential of complex systems theory and network theory. Complex systems theory and network theory have shown themselves to be extremely flexible, so the extent to which these concepts can be applied to the naval design problem, or any other problem, is limited solely by the creativity of future researchers.

BIBLIOGRAPHY

BIBLIOGRAPHY

Albert, R., H. Jeong, and A. Barabasi (2000), Error and attack tolerance of complex networks, *Nature*, *406*(6794), 378–82, doi:10.1038/35019019.

Alion Science and Technology (2011), Early Manpower Assessment Tool.

Altshuler, Y., R. Puzis, Y. Elovici, S. Bekhor, and A. Sandy (2011), Augmented Betweenness Centrality for Mobility Prediction in Transportation Networks, in *Finding Patterns of Human Behaviors in Network and Mobility Data*, pp. 1–12, Athens.

Anderson, W. B. (1914), *Physics for Technical Students: Mechanics and Heat*, McGraw Hill, New York, USA.

Andersson, G., et al. (2005), Causes of the 2003 Major Grid Blackouts in North America and Europe, and Recommended Means to Improve System Dynamic Performance, *IEEE Transactions on Power Systems*, *20*(4), 1922–1928, doi: 10.1109/TPWRS.2005.857942.

Andrews, D. (2003), A Creative Approach to Ship Architecture, *International Journal of Maritime Engineering*, *145*(3), 229–252.

Andrews, D. (2011), Art and science in the design of physically large and complex systems, *Proceedings of the Royal Society A: Mathematical, Physical, and Engineering Sciences*, *468*(2139), 891–912.

Andrews, D., R. Pawling, J. W. Gillespie, and D. J. Singer (2012), State of the Art Report: Design for Layout, Glasgow, Scotland.

Andrews, D. J., and R. G. Pawling (2003), SURFCON-A 21st Century Ship Design Tool, in *The Proceedings of the 8th International Marine Design Conference*, pp. 150–166, Athens, Greece.

Andrews, D. J., and R. G. Pawling (2008), A Case Study in Preliminary Ship Design, *International Journal of Maritime Engineering*, *150*(Part A3).

Arena, M. V., I. Blickstein, O. Younossi, and C. A. Grammich (2006), Why Has the Cost of Navy Ships Risen? A Macroscopic Examination of the Trends in U.S. Naval Ship Costs Over the Past Several Decades, *Tech. rep.*, RAND Corporation.

Asmara, A., and U. Nienhuis (2006), Automatic piping system in ship, in *The 5th International Conference on Computer and IT Applications in the Maritime Industries*, 1999, pp. 269–280, Delft.

Barigozzi, M., G. Fagiolo, and G. Mangioni (2011), Community Structure in the Multi-network of International Trade, *Complex Networks*, pp. 1–27.

Barthélemy, M. (2011), Spatial networks, *Physics Reports*, *499*, 1–86.

Bennett, J. G., and T. Lamb (1996), Concurrent Engineering: Application and Implementation for U.S. Shipbuidling, *Journal of Ship Production*, *12*(2), 107–125.

Blohm and Voss (2003), The Meko Technology, *Naval Forces*, *Special Issue*, 42–45.

Blondel, V. D., J.-L. Guillaume, R. Lambiotte, and E. Lefebvre (2008), Fast unfolding of communities in large networks, *Journal of Statistical Mechanics: Theory and Experiment*, *8*(10), doi:10.1088/1742-5468/2008/10/P10008.

Blumenfeld, J. (2012), Enlisted mess hall at breakfast, Photograph, https://www.flickr.com/photos/epar9jeb/7556434214/.

Bonchev, D. (2009), A Simple Integrated Approach to Network Complexity and Node Centrality, in *Analysis of Complex Networks: From Biology to Linguistics*, edited by M. Dehmer and F. Emmert-Streib, chap. 3, pp. 47–53, Wiley, Weinheim.

Bonchev, D., and N. Trinajstic (1977), Information theory, distance matrix, and molecular branching, *The Journal of Chemical Physics*, *67*(10), 4517, doi:10.1063/1.434593.

Boyer, J., and W. Myrvold (1999), Stop minding your P's and Q's: A simplified O (n) planar embedding algorithm, in *10th Annual ACM-SIAM Symposium on Discrete Algorithms*, vol. 1027, pp. 140–146.

Boyer, J., and W. Myrvold (2004), On the cutting edge: Simplified O (n) planarity by edge addition, *Journal of Graph Algorithms and Applications*, *8*(3), 241–273.

Bragança, L., S. M. Vieira, and J. B. Andrade (2014), Early stage design decisions: the way to achieve sustainable buildings at lower costs., *The Scientific World Journal*, *2014*, 365,364, doi:10.1155/2014/365364.

Buckley, M. E., and D. J. Singer (2013), Designing for Flexibility with Set Based Design: Observations from Ship to Shore Connector, in *ASNE Day 2013 Proceedings*, Arlington, VA.

Buldyrev, S. V., R. Parshani, G. Paul, H. E. Stanley, and S. Havlin (2010), Catastrophic cascade of failures in interdependent networks., *Nature*, *464*(7291), 1025–8, doi:10.1038/nature08932.

Calkins, D., R. Schachter, and L. Oliveira (2001), An automated computational method for planing hull form definition in concept design, *Ocean Engineering*, *28*(3), 297–327, doi:10.1016/S0029-8018(99)00069-4.

Caprace, J. D., and P. Rigo (2010a), Ship complexity assessment at the concept design stage, *Journal of Marine Science and Technology*, *16*(1), 68–75, doi: 10.1007/s00773-010-0107-9.

Caprace, J. D., and P. Rigo (2010b), A Complexity Metric for Practical Ship Design, in *Proceedings of the 11th International Symposium on Practical Design of Ships and Other Floating Structures (PRADS 2010)*, pp. 775–782, Rio de Janeiro, Brazil.

Cardoso, J. (2005), Control-flow Complexity Measurement of Processes and Weyukers Properties, *Proceedings of the World Academy of Science*, *8*(October), 213–218.

Casarosa, L. (2011), The integration of human factors, operability and personnel movement simulation into the preliminary design of ships utilising the Design Building Block approach, Ph.D. thesis, University College of London.

Code 223 NSWC-Carderock (2011), Section B - Design Activity Groups.

Code 223 NSWC-Carderock (2012), The Navy Ship Design Process.

Code 2230 NSWC-Carderock (n.d.), ASSET 6.3 Training Part II : The LEAPS Toolkit.

Collette, M. (2011), Advanced Marine Design, Course Notes.

Craggs, J., D. Bloor, B. Tanner, and H. Bullen (2004), Naval compensated gross tonnage coefficients and shipyard learning, *Journal of Ship Production*, *20*(2), 107–113.

Daqing, L., K. Kosmidis, A. Bunde, and S. Havlin (2011), Dimension of spatially embedded networks, *Nature Physics*, *7*(6), 481–484, doi:10.1038/nphys1932.

Dehmer, M. (2011), *Structural Analysis of Complex Networks*, Springer, London.

Dehmer, M., and F. Emmert-Streib (2009), *Analysis of Complex Networks: From Biology to Linguistics*, Wiley, Weinheim.

Dehmer, M., A. Mowshowitz, and F. Emmert-Streib (2013), *Advances in Network Complexity*, Wiley, Weinheim.

DeNucci, T. (2012), Capturing Design: Improving conceptual ship design through the capture of design rationale, Ph.D. thesis, Delft University of Technology.

Dicker, R. (1986), Survivability and the Modular Frigate, *International Defense Review*, *7*, 961–963.

Dijkstra, E. (1959), A note on two problems in connexion with graphs, *Numerische mathematik*, (1), 269–271.

Doerry, N. H. (2006a), Systems engineering and zonal ship design, *ASNE Engineering the Total Ship Symposium*, pp. 1–10.

Doerry, N. H. (2006b), Zonal ship design, *Naval Engineers Journal*, *118*(1), 39–53.

Dong, A. (1997), The management of design information : A decision-analytic approach, Ph.D. thesis, University of California at Berkeley.

Dong, A., and A. M. Agogino (1998), Managing design information in enterprise-wide CAD using 'smart drawings', *Computer-Aided Design*, *30*(6), 425–435, doi:10.1016/S0010-4485(97)00095-X.

Drezner, J. A., M. V. Arena, M. McKernan, R. Murphy, and J. Riposo (2011), Are Ships Different? Policies and Procedures for the Acquisition of Ship Programs, *Tech. rep.*, RAND Corporation.

DRS Technologies Inc (2013), Orca3d.

Erdös, P., and A. Rényi (1960), On the evolution of random graphs, *Publ. Math. Inst. Hungar. Acad. Sci.*

Estrada, E. (2006), Network robustness to targeted attacks. The interplay of expansibility and degree distribution, *The European Physical Journal B*, *52*(4), 563–574, doi:10.1140/epjb/e2006-00330-7.

Evers, S. (1997), USN takes "Smart Ship" onboard, *Janes' Defense Weekly*, *3*.

Federal News Service (2002), DoD News Briefing - Secretary Rumsfeld and Gen. Myers.

First Marine International (2005), Findings for the Global Shipbuilding Industrial Base Benchmarking Study Part 1: Major Shipyards, *Tech. rep.*

Fishman, G. S. (2001), *Discrete-Event Simulation: Modeling, Programming, and Analysis*, Springer, New York.

Fleck, M. M. (2013), Planar Graphs, in *Building Blocks for Theorectical Computer Science*, 1.3 ed., pp. 232–244, University of Illinois.

Fortunato, S. (2010), Community detection in graphs, *Physics Reports*, *486*(3-5), 75–174, doi:10.1016/j.physrep.2009.11.002.

Frank, M. V., and D. Helmick (2007), 21st Century HVAC System for Future Naval Surface Combatants - Concept Development Report, *Tech. Rep. September*, Naval Surface Warfare Center Carderock Division, West Bethesda, MD.

Freeman, L. (1977), A set of measures of centrality based on betweenness, *Sociometry*, *40*(1), 35–41.

Gale, P. A. (2003), The Ship Design Process, in *Ship Design and Construction*, edited by T. Lamb, chap. 5, pp. 5.1–5.40, Society of Naval Architects and Marine Engineers, Jersey City, NJ.

Gallin, C. (1973), Which way computer aided ship design?, in *ICCAS*, Tokyo.

Gao, J., S. V. Buldyrev, H. E. Stanley, and S. Havlin (2011), Networks formed from interdependent networks, *Nature Physics*, *8*(1), 40–48, doi:10.1038/nphys2180.

Gaspar, H. M., A. M. Ross, D. H. Rhodes, and S. O. Erikstad (2012), Handling Complexity Aspects in Conceptual Ship Design, in *Proceedings of the 11th International Marine Design Conference*, June, pp. 1–14.

Gillespie, J. W. (2012), A Network Science Approach to Understanding and Generating Ship Arrangements in Early-Stage Design, Ph.D. thesis, University of Michigan.

Girvan, M., and M. E. J. Newman (2002), Community structure in social and biological networks, *Proceedings of the National Academy of Sciences of the United States of America*, *99*(12).

Gleich, D. (2008), Planar graphs in MatlabBGL, Website, https://www.cs.purdue.edu/homes/dgleich/packages/matlab_bgl/planar_graphs/planar_graphs.html.

Global Security (n.d.), CG-47 Ticonderoga, http://www.globalsecurity.org/military/systems/ship/cg-47.htm.

Gong, Y., and Z. Zhang (2009), Global robustness and identifiability of random, scale-free, and small-world networks., *Annals of the New York Academy of Sciences*, *1158*, 82–92, doi:10.1111/j.1749-6632.2008.03752.x.

Government Accounting Office (2002), Capturing Design and Manufacturing Knowledge Early Improces Acquisition Outcomes, *Tech. rep.*

Government Accounting Office (2007), Realistic Business Cases Needed to Execute Navy Shipbuilding Programs, *Tech. rep.*

Gray, A. (2011), Enhancement of Set-Based Design Practices Via Introduction of Uncertainty Through Use of Interval Type-2 Modeling and General Type-2 Fuzzy Logic Agent Based Methods, Phd dissertation, University of Michigan.

Gray, A., B. Cuneo, N. Vlahopoulos, and D. J. Singer (2013), The Rapid Ship Design Environment Multi-Disciplinary Optimization of a U . S . Navy Frigate, in *Proceedings of ASNE Day 2013*, pp. 1–13, Crystal City, VA.

Hassan, M. M., and G. L. Hogg (1987), A review of graph theory application to the facilities layout problem, *Omega*, *15*(4), 291–300, doi:10.1016/0305-0483(87)90017-X.

Helbing, D., and A. Johansson (2009), *Pedestrian, crowd and evacuation dynamics*, 6476–6495 pp.

Henry, S., D. Kafura, and K. Harris (1981), On the relationships among three software metrics, in *ACM Workshop/Symposium on Measurement and Evaluation of Software Quality*, pp. 81–88, College Park, MD.

Hope, J. P. (1981), The process of naval ship general arrangement design and analysis, *Naval Engineers Journal1*, *93*(4), 29–38.

Hu, Y., B. Ksherim, R. Cohen, and S. Havlin (2011), Percolation in interdependent and interconnected networks: Abrupt change from second- to first-order transitions, *Physical Review E*, *84*(6), 066,116, doi:10.1103/PhysRevE.84.066116.

Hummon, N., and P. Dereian (1989), Connectivity in a citation network: The development of DNA theory, *Social Networks*, *11*, 39–63.

International Maritime Organization (2007), Guidelines for Evacuation Analysis for New and Existing Passenger Ships. MSC Circular 1238.

Iranmanesh, A., and Y. Alizadeh (2010), An algorithm for computing the Randic and Zagreb indices of a graph, *Optoelectronics and Advanced Materials*, *4*(1), 50–52.

Jacobi, F. (2003), Modular ship design, *Naval Forces*, *24*(1), 36–44.

Jutla, I. S., L. G. S. Jeub, and P. J. Mucha (2011), A generalized Louvain method for community detection implemented in MATLAB, Computer Program, http://netwiki.amath.unc.edu/GenLouvain.

Keane, R. G., and B. F. Tibbitts (1996), A Revolution in Warship Design: Navy-Industry Integrated Product Teams, *Journal of Ship Production*, *12*(4), 254–268.

Keane Jr., R. G. (2011), Reducing Total Ownership Cost: Designing Inside-Out of the Hull, in *Proceedings of ASNE Day 20112*, Arlington, VA.

Kemink, L., T. A. McKenney, M. C. Parker, and A. Partida (2009), Cougar-108 Design Report, *Tech. rep.*, University of Michigan.

Kier, L., and L. Hall (2002), The meaning of molecular connectivity: A bimolecular accessibility model, *Croatica chemica acta*, *75*(2), 371–382.

Knight, J. (2014), A Prospect Theory-Based Real Option Analogy for Evaluating Flexible Systems and Architectures in Naval Ship Design, Ph.d. thesis, University of Michigan.

Kortler, S., M. Kreimeyer, and U. Lindemann (2009), A Planarity-based Complexity Metric, *Proceedings of th 17th International Conference on Engineering Design*, p. 381.

Lambiotte, R., and M. Rosvall (2012), Ranking and clustering of nodes in networks with smart teleportation, *Physical Review E*, *85*(5), 056,107–056,115.

Lambiotte, R., J. Delvenne, and M. Barahona (2008), Laplacian dynamics and multiscale modular structure in networks, *arXiv preprint arXiv:0812.1770*, pp. 1–29.

Larsson, L., and R. E. Eliasson (2007), *Principles of Yacht Design*, 3rd ed., International Marine/McGraw Hill.

Latorre, R., and J. M. Vasconcellos (2002), Introduction of Software Packages in Naval Architecture, Marine and Ocean Engineering Courses, *International Journal of Engineering Education*, *18*(1), 98–105.

Leicht, E., and R. D'Souza (2009), Percolation on interacting networks, *arXiv preprint arXiv:0907.0894*, *2*, 2–6.

Lewis, K., J. Kaufman, M. Gonzalez, A. Wimmer, and N. Christakis (2008), Tastes, ties, and time: A new social network dataset using Facebook.com, *Social Networks*, *30*(4), 330–342, doi:10.1016/j.socnet.2008.07.002.

Liu, H., M. Lu, and F. Tian (2005), On the Randić index, *Journal of Mathematical Chemistry*, *38*(3), 345–354, doi:10.1007/s10910-005-5824-7.

Marvis, D. N., and D. DeLaurentis (2000), Methodology for Examing the Simultaneous Impact of Requirements, Vechicle Characteristics, and Technologies on Militray Aircraft Design, in *Proceedings of the 22nd Congress of the International Council on the Aeronautical Sciences*, pp. 145.1–145.10, Harrogate, England.

Masucci, A., D. Smith, A. Crooks, and M. Batty (2009), Random planar graphs and the London street network, *The European Physical Journal*, pp. 1–14.

McKenney, T. A. (2013), An Early-Stage Set-Based Design Reduction Decision Support Framework Utilizing Design Space Mapping and a Graph Theoretic Markov Decision Process Formulation, Ph.d. thesis, University of Michigan.

McKenney, T. A., L. Kemink, and D. J. Singer (2011), Adapting to Changes in Design Requirements using Set-Based Design, *Naval Engineers Journal*, (3), 67–77.

McKenzie, M. (2012), Stability-optimization algorithms for the detection of community structure in networks, Ph.D. thesis, University of Oxford.

Miller, J. H., and S. E. Page (2007), *Complex Adaptive Systems*, Princeton University Press, Princeton.

Miroyannis, A. (2006), Estimation of Ship Construction Costs, Masters thesis, Massachusetts Institute of Technology.

Mucha, P. J., T. Richardson, K. Macon, M. a. Porter, and J.-P. Onnela (2010), Community structure in time-dependent, multiscale, and multiplex networks., *Science*, *328*(5980), 876–8, doi:10.1126/science.1184819.

Muther, R. (1973), *Systematic Layout Planning*, Cahners Books, Boston, USA.

National Science Board (2012), Science and Engineering Labor Force, in *Science and Engineering Indicators 2012*, chap. 3, pp. 3.1–3.66, National Science Foundation, Arlington, VA.

Naval Sea Systems Command (2005), Electric Power Distribution Systems, *Tech. Rep. S90860KY–STM–010*, US Navy.

Neti, S. (2005), Ship design optimization using ASSET, Ph.D. thesis, Virginia Polytechnic Institute and State University.

Newman, M. E. J. (2004), Analysis of weighted networks, *Physical Review E*, *70*(5), 056,131, doi:10.1103/PhysRevE.70.056131.

Newman, M. E. J. (2010), *Networks: An Introduction*, Oxford University Press, Oxford.

Newman, M. E. J. (2012), Communities, modules and large-scale structure in networks, *Nature Physics*, *8*, 25–31.

Nick, E. K. (2008), Fuzzy Optimal Allocation and Arrangement of Spaces in Naval Surface Ship Design, Ph.D. thesis, University of Michigan.

Nowacki, H. (2009), Development in marine design metholdology: Roots, results and future trends, in *Proceedings of the 10th International Marine Design Conference*, Trondheim, Norway.

Office of Naval Research (2011), Simulation Toolset for Analysis of Mission, Personnel, and Systems, *Tech. rep.*

Page, S. E. (2009), Understanding Complexity, Video, 12 Lectures, The Great Courses.

Page, S. E. (2010), *Diversity and complexity*, Princeton University Press, Princeton, Oxford.

Parker, M. C., and D. J. Singer (2013), The Impact of Design Tools: Looking for Insights with a Network Theoretic Approach, in *Proceedings of the 12th International Conference on Computer Applications and Information Technology in the Maritime Industries*, pp. 96–109, Cortona.

Parshani, R., S. Buldyrev, and S. Havlin (2010), Interdependent Networks: Reducing the Coupling Strength Leads to a Change from a First to Second Order Percolation Transition, *Physical Review Letters*, *105*(4), 048,701, doi:10.1103/PhysRevLett.105.048701.

Parsons, G. H., J. H. Nam, and D. Singer (1998), A Scalar Metric for Assessing the Producibility of a Hull form in Early Design, in *Great Lakes/Great Rivers SNAME Section Meeting*.

Parsons, M. G., H. Chung, E. Nick, A. Daniels, S. Liu, and J. Patel (2008), Intelligent Ship Arrangements : A New Approach to General Arrangement, *Naval Engineers Journal*, *120*(3), 51–65, doi:10.1111/j.1559-3584.2008.00153.x.

Pawling, R. (2007), The Application of the Design Building Block Approach to Innovative Ship Design, Ph.D. thesis, University College London.

Pawling, R., and D. Andrews (2011), Design sketching: the next advance in computer aided preliminary ship design, in *The 10th International Conference on Computer and IT Applications in the Maritime Industries*, 1991, pp. 426–441.

Pawling, R., D. Andrews, and L. Uk (2013), An Integrated Approach to Style Definition in Early Stage Design, in *Proceedings of the 12th International Conference on Computer Applications and Information Technology in the Maritime Industries*, pp. 248–263, Cortona.

Petry, C., and J. Rumburg (1993), Zonal electrical distribution systems: an affordable architecture for the future, *Naval Engineers Journal*, *105*(3), 45–51.

Radicchi, F., C. Castellano, F. Cecconi, V. Loreto, and D. Parisi (2004), Defining and identifying communities in networks, *Proceedings of the National Academy of Sciences of the United States of America*, *101*, 2658–2663.

Randić, M. (1975), On Characterization of Molecular Branching, *Journal of the American Chemical Society*, *97*(23), 6609–6615, doi:10.1021/ja00856a001.

Rehman, F., and X. Yan (2008), A Case Study to Support Conceptual Design Decision Making Using Context Knowledge, in *Advanced Design and Manufacture to Gain a Competitive Edge: New Manufacturing Techniques and their Role in Improving Enterprise Performance*, pp. 13–22, Springer.

Rickles, D., P. Hawe, and A. Shiell (2007), A simple guide to chaos and complexity., *Journal of epidemiology and community health*, *61*(11), 933–7, doi: 10.1136/jech.2006.054254.

Rigterink, D., and D. J. Singer (2009), An assessment of producibility of a stiffened hull plate in early design stage, in *Proceedings of 8th International Conference on Computer and IT Applications in the Maritime Industries*, 1, pp. 564–575, Budapest, Hungary.

Rigterink, D., and D. J. Singer (2014a), A network complexity metric based on planarity and community structure, *Journal of Complex Networks*, doi: 10.1093/comnet/cnu005.

Rigterink, D., and D. J. Singer (2014b), Distributed Systems Design Evaluation Using A Network-Scientific Framework, poster. ASNE Day 2014. Arlington, VA.

Rigterink, D., M. Collette, and D. J. Singer (2012), A Novel Structural Complexity Metric and its Impact on Structural Cost Estimating, in *Proceedings of the 11th International Marine Design Conference*, Glasgow.

Rigterink, D., M. Collette, and D. J. Singer (2013a), A method for comparing panel complexity to traditional material and production cost estimating techniques, *Ocean Engineering*, *70*, 61–71, doi:10.1016/j.oceaneng.2013.05.031.

Rigterink, D., R. Piks, and D. J. Singer (2013b), The use of network theory to model disparate ship design information, in *Proceedings of the 12th Symposium on Practical Design of Ships and Floating Structures*.

Rigterink, D., C. Shields, and D. J. Singer (2014), Network Community Detection to Create Reduced-Order Personnel Movement Models for Early Stage Design, presentation. ASNE Day 2014. Arlington, VA.

Rinaldi, S., J. Peerenboom, and T. Kelly (2001), Identifying, understanding, and analyzing critical infrastructure interdependencies, *Control Systems, IEEE*, *21*(6), 11–25.

Rittel, H. W. J., and M. M. Webber (1973), Dilemmas in a General Theory of Planning, *Policy Sciences*, *4*, 155–169.

Ross, J. M. (2003), Computer-Based Tools, in *Ship Design and Construction*, edited by T. Lamb, chap. 13, pp. 1–26, Society of Naval Architects and Marine Engineers, Jersey City, NJ.

Rosvall, M., and C. T. Bergstrom (2008), Maps of random walks on complex networks reveal community structure., *Proceedings of the National Academy of Sciences of the United States of America*, *105*(4), 1118–23, doi:10.1073/pnas.0706851105.

Rosvall, M., A. Trusina, P. Minnhagen, and K. Sneppen (2005), Networks and cities: An information perspective, *Physical Review Letters*, (4), 2–5.

Rosvall, M., D. Axelsson, and C. Bergstrom (2009), The map equation, *The European Physical Journal Special Topics*, *178*(1), 13–23.

Sawai, H. (2013), A Small-World Network Immune from Random Failures and Resilient to Targeted Attacks, *Procedia Computer Science*, *18*, 976–985, doi:10.1016/j.procs.2013.05.263.

Schaeffer, S. E. (2007), Graph clustering, *Computer Science Review*, *1*(1), 27–64, doi:10.1016/j.cosrev.2007.05.001.

Schank, J. F., C. G. Pernin, M. V. Arena, C. C. Price, and S. K. Woodward (2009), Controlling the Cost of C4I Upgrades on Naval Ships, *Tech. rep.*, RAND Corporation.

Scofield, T. (2006), Manning and Automation Model for Naval Ship Analysis and Optimization, Ph.D. thesis, Virginia Polytechnic Institute and State University.

Sellers, M. (2011), Social network analysis, *Tech. rep.*, Mercyhurst College, http://intl520-summer2011-mas.wikispaces.com/Home.

Shanker, O. (2013), Dimension Measure for Complex Networks, in *Advances in Network Complexity*, edited by M. Dehmer, A. Mowshowitz, and F. Emmert-Streib, 1 ed., chap. 8, pp. 191–208, Wiley, Weinheim.

ShipConstructor Software Inc. (2012), ShipConstructor.

Shouman, M. A., G. M. Nawara, A. H. Reyad, and K. El-darandaly (2001), Facility layout problem (FLP) and intelligent techniques: a survey, in *Proceedings of the 7th International Conference on Production Engineeering, Design and Control*, pp. 409–422, Alexandria, Egypt.

Singer, D. J. (2003), A Hybrid Agent Approach for Set-Based Conceptual Ship Design Through the Use of a Fuzzy Logic Agent to Facilitate Communications and Negotiation, Phd dissertation, University of Michigan.

Singer, D. J., M. Collette, M. P. Van Oyen, and N. Vlahopoulos (2012), Development of the Symbiotic Design Concept and LEAPS Based Supporting Design Tools for the STAMPS Program, *Tech. rep.*

Singh, S. P., and R. R. K. Sharma (2005), A review of different approaches to the facility layout problems, *The International Journal of Advanced Manufacturing Technology*, *30*(5-6), 425–433, doi:10.1007/s00170-005-0087-9.

Sivakumar, L., and M. Dehmer (2012), Information inequalities and Generalized Graph Entropies, *PloS one*, *7*(6), e38,159.

Slabodkin, G. (1998), Software glitches leave Navy Smart Ship dead in Water, *Government Computer News.*

Sornette, D. (2009), Dragon-kings, black swans and the prediction of crises, *SSRN Electronic Journal*, doi:10.2139/ssrn.1470006.

Taleb, N. N. (2007), *The Black Swan: the Impact of the Highly Improbably*, Random House, United States.

Teknomo, K., and P. Fernandez (2012), Simulating optimum egress time, *Safety Science*, *50*(5), 1228–1236, doi:10.1016/j.ssci.2011.12.025.

Thomassen, C. (1981), Kuratowski's theorem, *Journal of Graph Theory*, *5*(3), 225–241, doi:10.1002/jgt.3190050304.

Todeschini, R., and V. Consonni (2009), *Molecular Descriptors for Chemoinformatics Volume II: Appendices, References*, vol. I, 2 ed., 161–162 pp., Wiley, Weinheim.

United States Navy (2013), Cruisers, Website, http://www.navy.mil/navydata/fact_display.asp?cid=4200&tid=800&ct=4.

Van Dokkum, K. (2007), *Ship Knowledge*, 4th ed., DOKMAR, Enkhuizen, The Netherlands.

van Oers, B. (2011), A Packing Approach for the Early Stage Design of Service Vessels, Ph.D. thesis, Delft University of Technology.

van Oers, B., and H. Hopman (2012), Simpler and Faster: a 2.5D Packing-Based Approach for Early Stage Ship Design, in *Proceedings of the 11th International Marine Design Conference*, pp. 297–316, Glasgow, Scotland.

Ward, A., D. K. I. Sobek, J. J. Cristiano, and J. K. Liker (1995), Toyota Concurrent Engineering and Set-Based Design, in *Engineered in Japan: Japanese Technology Management Practices*, chap. 8, pp. 192–216, Oxford University Press, New York, NY.

Watson, D., and A. Gilfillan (1977), Some ship design methods, *Naval Architect*, *119*.

Watts, D. J., and S. H. Strogatz (1998), Collective dynamics of 'small-world' networks., *Nature*, *393*(6684), 440–442, doi:10.1038/30918.

Weyuker, E. (1988), Evaluating software complexity measures, *IEEE Transactions on Software Engineering*, *14*(9), 1357–1365, doi:10.1109/32.6178.

Wired News (1998), Sunk by Windows NT, *Wired News*.

Zhang, J.-d., and J.-h. Sui (2002), Distributed and redundant design of ship monitoring and control network, *Journal of Marine Science and Application*, *1*(2), 12–17, doi:10.1007/BF02935832.

Zhao, J., and K. Xu (2009), Enhancing the robustness of scale-free networks, *Journal of Physics A: Mathematical and Theoretical*, *42*(19), 195,003, doi:10.1088/1751-8113/42/19/195003.