

ABSTRACT

ARNEY, KRISTIN MELIA. Global Sensor Management: Allocation of Military Surveillance Assets. (Under the direction of Professor Thom J. Hodgson).

The United States uses an integrated missile defense system to detect, track, classify and destroy any threats from the air. This system can be represented by a network of nodes and arcs where the nodes are depictive of the sensors and their required functions and the arcs detail the ability of a sensor to complete a function and pass along the appropriate information to a machine in a subsequent stage in the operation.

Given a set of sensor resources, we need to develop a management tool that assigns resources to tasks and functions to maximize our ability to detect and eliminate threats and our sensor coverage of various regions of earth and space, and that tool needs to be flexible enough to deal with continuous changes to the environment in which it is operating.

To that end, we investigate and develop both a total enumeration method generating all possibilities and determining the optimal using our own developed methodology implemented through a Visual Basic code. This brute force method in order to find optimal assignments is not the preferred method due to the potential time constraints associated with scaling up the network and developing a real time decision. However, in this research, we focus on an operational planning model which assists us in understanding the complexities and sensitivities of the problem.

Throughout the project, STRATCOM's 4-task sample network is used as a test case. Probability values are randomly generated to be representative of true classified data. In solving the sample network for the static case for missile defense type event, we use a model calculating probabilities based on the Law of Total Probability. Using total enumeration, we are to run all possible allocations for the assignable assets and determine all associated endstage probabilities. From here, we can assign the assets to the appropriate task according to any constraints provided. Constraints could be both lower

bounds for tasks 2, 3, and 4 as well as specifications for the endstage probabilities for machines in stage 10 of task 1.

Results suggest that the sample network is both easy to solve and very insensitive. We assume that models to solve this instance can be applied to higher order problems with additional assignable assets, different network connections, as well as multiple events.

Global Sensor Management: Allocation of Military Surveillance Assets

by
Kristin Melia Arney

A thesis submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Master of Science

Operations Research

Raleigh, North Carolina

May 2008

APPROVED BY:

Thom J. Hodgson
Committee Chair

Russell E. King

E. Downey Brill

Tim Trainor

BIOGRAPHY

Kristin Melia Fetherston-Arney was born in Fayetteville, North Carolina in 1976. She attended Lafayette College in Easton, Pennsylvania and graduated Magna Cum Laude in May 1998 with a Bachelor of Science in Mathematics. She was commissioned as a Second Lieutenant in the United States Army in 1998 through the Reserve Officer Training Corps (ROTC) Program at Lehigh University as a Distinguished Military Graduate. She was branched Military Intelligence Corps and began her military education at Fort Huachuca, Arizona.

Her military assignments include being stationed at Fort Lewis, Washington serving as the Intelligence Officer for a Combat Engineering battalion and the Executive Officer and Collection Manager for an operational Military Intelligence battalion, as well as an extended overseas tour in Germany serving as a Plans Officer for a theater logistics unit in Kaiserslautern, Germany, a Intelligence Battle Captain during deployment to Mardin, Turkey, and a Signals Officer, the Threat Analysis Branch/Information Dominance Center Chief, Company Commander, and Acting Battalion Executive Officer within Europe's remaining Military Intelligence Group.

Recently promoted to Major in the U.S. Army, upon graduation, Kristin will be assigned to the United States Military Academy at West Point, NY as a mathematics instructor.

She is married to Major Andrew L. Fetherston, who is currently instructing future leaders in the Wolfpack ROTC Program. They have a son, George Owen, who is 3 years old and a daughter, Sophie, who was born during the fall semester of 2007.

ACKNOWLEDGMENTS

Inner Circle – And you know who you are. I could not have completed this project without us all working together. I have enjoyed the intellectual exchanges as well as the personal attacks... May we all one day meet in the cone of silence.

My wonderful family - These two years have been a wonderful time for us all to grow together. We have been able to share so much. I am forever appreciative of the never ending love from my husband, caring from my mom, and support from my dad (and I can't forget the funding and time off from the Army).

TABLE OF CONTENTS

LIST OF TABLES.....	vi
LIST OF FIGURES.....	vii
1. Introduction.....	1
1.1 Background.....	1
1.2 Problem Description.....	2
1.3 Objective.....	5
1.4 Assumptions and Limitations.....	5
1.5 Overview.....	6
2. Literature Review.....	7
2.1 Sensors.....	7
2.1.1 Sensor Networks.....	7
2.1.2 Task Scheduling.....	9
2.1.3 Sensor Functions.....	9
2.2 Network Reliability.....	10
2.3 Resource Allocation Methods.....	11
3. Methodology.....	13
3.1 Initial Model Formulations.....	13
3.1.1 Problem Description.....	13
3.1.2 Objective.....	13
3.1.3 Assumptions and Proposed Methodology.....	15
3.1.4 First Approach: Phase I Model.....	15
3.1.5 Refined Approach: Phase II Model.....	17
3.1.6 Other Model Considerations.....	19
3.1.7 Optimization.....	21
3.1.8 Conclusions.....	22
3.2 Expanded Sample Model.....	22
3.2.1 Structure.....	22
3.2.1 Implications.....	28
3.3. Model Formulation.....	29
3.3.1 Descriptions.....	29
3.3.2 Assumptions.....	30
3.3.3 Optimization of the Network.....	33
3.3.4 Calculation of Network Probabilities.....	34
3.3.4.1 Description.....	34
3.3.4.2 Notation.....	35
3.3.4.3 Example Calculation.....	37
3.3.4.4 Comparison to STRATCOM's Probability Calculations.....	38
4. Implementation and Results.....	42
4.1 Model Description Using Visual Basic.....	42
4.2 Model Refinement.....	42
4.2.1 Data Requirements.....	42
4.2.2 Accuracy.....	44

4.3 Model Results.....	47
4.3.1 Based on Initial Methodology.....	47
4.3.2 Expanded Probability Ranges.....	49
4.3.3 Overall Observations.....	51
4.3.4 Comparison.....	53
5. Conclusions.....	56
6. Future Research.....	57
6.1 Overview.....	57
6.2 Multiple Events.....	58
6.3 Timing.....	59
6.4 Accuracy.....	61
6.5 Real Time Decision Aid Heuristic.....	61
References.....	63
Appendices.....	65
Appendix A: Network Connection File.....	66
Appendix B: Results – Probability Range 0.50 to 0.80.....	70
Appendix C: Results – Probability Range 0.80 to 0.95.....	72
Appendix D: Results – Probability Range for Stage 2 0.50 to 0.80 and For all Subsequent Stages 0.95 to 1.0.....	74
Appendix E: One run – Probability Range 0.95 to 1.....	76

LIST OF TABLES

Table 3.1: A 4-machine, 3-intruder, 2-stage Constraint Matrix (P1).....	16
Table 3.2: A 4-machine, 3-intruder, 2-stage Constraint Matrix (P2).....	18
Table 3.3: Law of Total Probability for Two Machines.....	20
Table 4.1: Results for Probability Range 0.95 to 1.0.....	47
Table 4.2: Summary Results for Expanded Probability Ranges.....	50
Table 4.3: Summary Results for Comparing Lower Probabilities for Stage 2.....	51
Table 4.4: Summary Results for Comparison of two Calculations.....	53

LIST OF FIGURES

Figure 1.1: Initial Allocation of Resources.....	4
Figure 1.2: Reassignment of Resources.....	4
Figure 2.1: Serial Network.....	10
Figure 2.2: Parallel Network.....	10
Figure 2.3: Mixed Serial-Parallel Network.....	11
Figure 3.1: Job Shop Model Network.....	14
Figure 3.2: Task 1 Network Diagram.....	24
Figure 3.3: Task 2 Network Diagram.....	25
Figure 3.4: Task 3 Network Diagram.....	26
Figure 3.5: Task 4 Network Diagram.....	27
Figure 3.6: Example Network to Explain Probability Calculations.....	37
Figure 4.1: Endstage Probabilities for Machines 23/24.....	54
Figure 4.2: Endstage Probability Differences for Machines 23/24.....	55

1. Introduction

1.1 Background

An integrated missile defense system is comprised of a set of entities such as radars, aircraft, and satellites that are used to provide defense against enemy missiles and aircraft. In general, the United States uses this system to detect, track, classify and destroy any threats from the air.

A number of factors drive the requirements for a robust capability. The proliferation of ballistic missile technology, its associated threats, and weapons proliferation have accelerated in recent years. Considerable danger resides in the development of chemical, biological, and nuclear weapons that can be paired with ballistic missiles for use against the United States, military troops, and allies. State and non-state actors may use weapons of mass destruction carried on ballistic missiles.

The possession of weapons of mass destruction and ballistic missiles by potential adversaries is an urgent security issue for the US and its allies – they are real threats to international peace and security. Therefore, the United States must have the ability to defeat ballistic missile attacks should they occur. The United States must take full advantage of increasingly capable sensors and weapons to counter enemy capabilities and gain information on the capabilities of these and other enemy systems. Most importantly, all air and missile threats must be neutralized or destroyed as far away as practical from U.S. interests.

Additionally, this system must also consider the other objects which operate in space and orbit the earth. The system must track satellites and other objects in space in order to detect changes in their trajectories and to ensure these objects do not fall from space and impact the earth or collide with another object in space which could cause disruption to many other activities.

This system can be represented by a network of nodes and arcs where the nodes are depictive of the sensors and their required functions and the arcs detail the ability of a sensor to complete a function and pass along the appropriate information to a machine in a subsequent stage in the operation.

This problem is of great interest to the members of the United States Strategic Command (STRATCOM), as they attempt to improve upon the scheduling of their sensor networks. STRATCOM's mission is to "provide the nation with global deterrence capabilities and synchronized Department of Defense (DoD) effects to combat adversary weapons of mass destruction worldwide. Enable decisive global kinetic and non-kinetic combat effects through the application and advocacy of integrated intelligence, surveillance and reconnaissance (ISR); space and global strike operations; information operations; integrated missile defense and robust command and control." STRATCOM deters military attacks on the United States and its allies and if this policy of deterrence fails, employs forces to achieve the national objectives. Of those national objectives, missile defense, space surveillance, intelligence collection and missile warning are of greatest concern.

1.2 Problem Description

The research in this document is the result of the merging of these four mission areas. The evolution of sensors, which can support multiple missions creates coordination and prioritization challenges. These sensors belong to a global system supporting multiple missions including missile defense, space surveillance, intelligence collection, and missile warning. In order to best accomplish these tasks, the U.S. military has a network of sensors dedicated to the many functions required by these primary tasks. Among these sensors are some that are space-based and others that are earth- (land, sea or air) based; some are in fixed positions while others have varying degrees of mobility; most importantly, each has a unique set of functions which it is capable of addressing with differing probabilities of success, depending both upon the sensor's own limitations and the state of nature in which it is working.

The goal is to assign our network of sensor resources to tasks and subordinate functions so as to maintain the largest area of coverage with respect to space surveillance, intelligence collection and missile warning, while at the same time maximizing our probability of detecting (and ultimately destroying) any threats that might emerge from around the globe. We wish to build a decision tool that optimizes sensor support across all mission areas and calculates measures of success for missile defense and missile warning (detecting, assessing, and killing what needs to be destroyed) and space surveillance and intelligence collection (collecting what needs to be collected). In optimizing the system, we need to support the identified mission priorities and requirements for all events during any given timeframe and dynamically balance the tasks should one or more of the machines fail or another event occurs.

Figures 1.1 and 1.2 provide a simple pictorial representation of that environment, as well as the changes to the environment that will come about with the implementation of our modeling efforts. Initially, each sensor is assigned to one or more mission areas, as well as a region towards which to direct its attention. As events occur, resources may be moved or reallocated to other tasks in an effort to successfully manage the event. Or, depending upon the capabilities of the sensors, resources may be assigned to different events within a task.

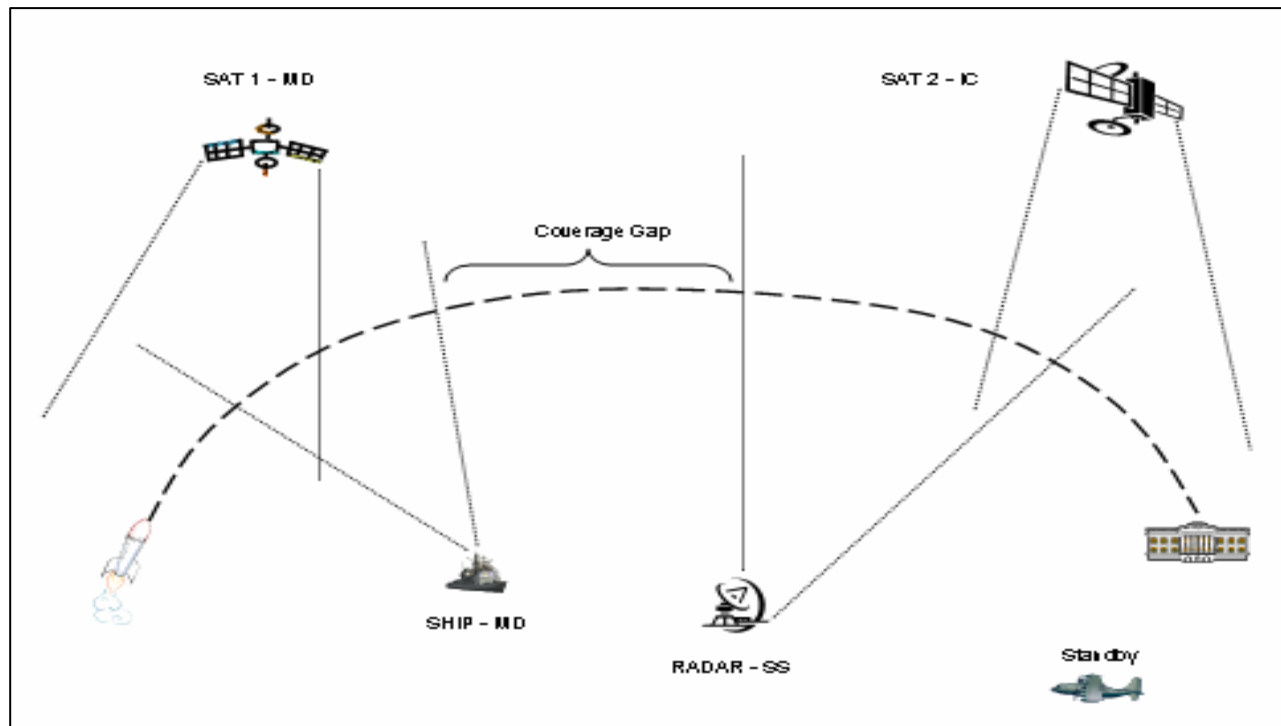


Figure 1.1: Initial Allocation of Resources

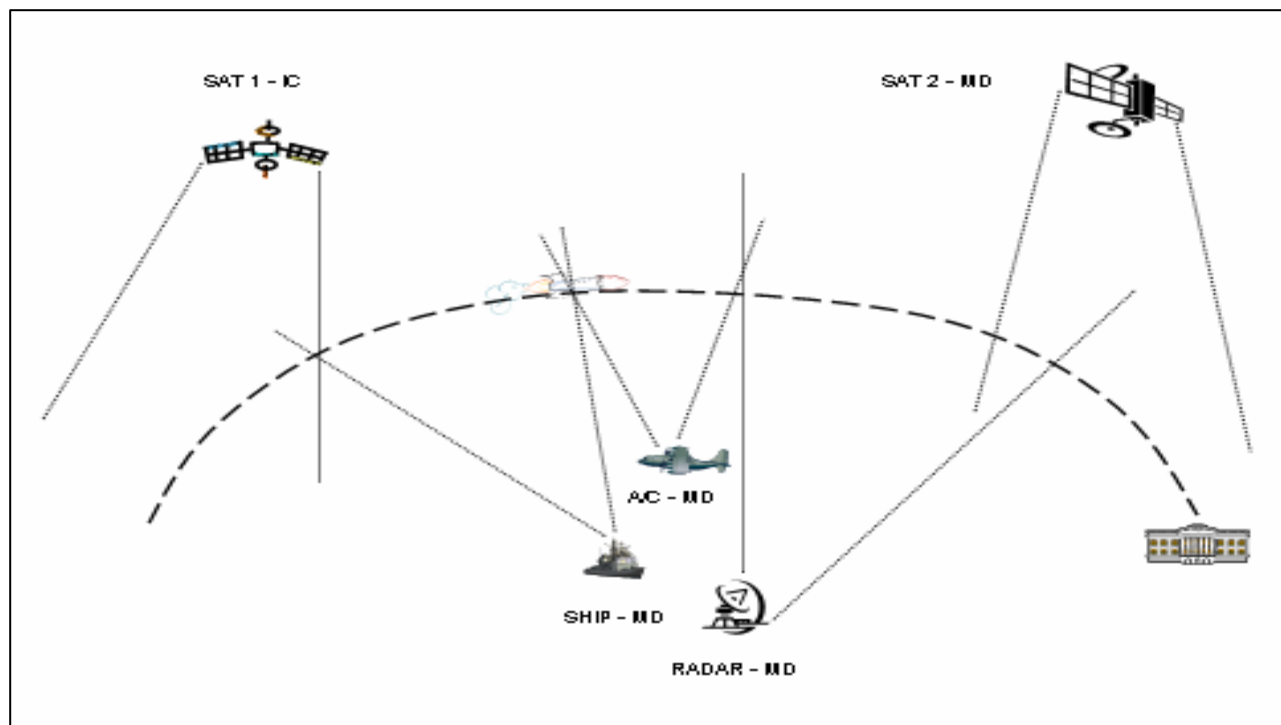


Figure 1.2: Reassignment of Resources

1.3 Objective

This problem is worthy of much research due to the inadequacies in the current process. As a high priority mission becomes available, the current process involves cancelling all other missions and tasking all assets against the high priority mission, even though the marginal contribution of some of those assets is very small. There is currently no methodology available to select an optimal combination of assets to accomplish all missions with reasonable probability of success.

Thus the challenge is two-fold. First, is the development of a operational planning model to determine the appropriate initial allocation of sensor resources to functions within (or across) mission areas, and how do we reassign sensor resources to different functions as time and circumstances dictate; and second, how do we do this in a real-time or near real-time manner that still maximizes our ability to achieve our stated objectives. The research contained herein pertains solely to the first issue.

Given a set of sensor resources, we need to develop a management tool that assigns resources to tasks and functions to maximize both our ability to detect and eliminate threats and our sensor coverage of various regions of earth and space, and that tool needs to be flexible enough to deal with continuous changes to the environment in which it is operating.

1.4 Assumptions and Limitations

The current assumption is that the required probability data for all events within the system has already been gathered and confirmed by STRATCOM. Additionally, these probabilities of success for all sensors assigned to a mission and function are known in sufficient time to be utilized as input. The methodology for calculating these probabilities in real time is dependant on sensor location, capability, and configuration in addition to the target location, type and trajectory. Since most of this data is classified, we will use methods to randomly generate appropriate values for our research. This is appropriate

since the probability values are not of prime interest, instead the goal is to determine the methodology for finding ideal sensor assignments.

One requirement for this type of problem is that the model and associated solver yields a solution in a reasonable amount of time. Finding an optimal solution may not always be possible given a time limitation. Therefore, heuristic methods approximating a solution close to optimal may be preferable as the true network of sensors could be much larger than discussed here.

1.5 Overview

The remainder of this document is as follows: Chapter 2 outlines the current literature in areas relevant to this research; Chapter 3 describes the problem, details the necessary assumptions, and develops a series of models of increasing complexity by which we plan to address and solve the problem; and Chapter 4 applies our proposed methodology to solve the given sample network and provides comparative results; Chapter 5 completes this research with conclusions; and finally, Chapter 6 discusses the way ahead for the remainder of this analysis and future research.

2. Literature Review

2.1 Sensors

Before embarking upon our model development, it is vital to fully understand the problem. A primary piece of that understanding lies in the network resources with which we will be working—namely, the sensors. We will be considering several types of sensors on several different platforms. Some of our sensors are multi-function, while others are dedicated; sensors are housed in satellites, ground radar stations, ships and aircraft; satellite and ground stations are treated as stationary, although they can be redirected to monitor different specific locations within a general region, while ships and aircraft are mobile and can be quickly redeployed, but generally have much smaller coverage areas. Each sensor has its own internal set of tasks that it is capable of accomplishing, but we need not concern ourselves with these as they are in place to allow the sensor to accomplish the larger roles that we are addressing.

2.1.1 Sensor Networks

Network models have been studied ad nauseam over the years, and the works dedicated to finding the best methods for obtaining a wide range of objectives within those networks are plentiful, to say the least. More recently, network models have been developed to address the growing number and sizes of sensor networks. These networks can have a wide range of applications (e.g. military, entertainment), and can be fairly localized or global. Regardless of the scope or the use, there are several characteristics which are common among most sensor networks—primary among these is communication, which drives all other factors. Typically, sensors are deployed within some region for the purpose of gathering data to relay back to a task manager. This data may be sent directly from each sensor to the manager, may first be passed through other sensors, or may be collected at a common point from multiple sensors before the consolidated information is relayed to the manager. In any case, the key is that data collected by each sensor is

ultimately returned to a management node for analysis [Akyildiz, et al]. There is, however, a significant challenge in this communication between sensors and the task manager that collates the data and formulates an appropriate response—namely the compatibility of data received from different sensors/sensor types. Typically more data is better, and viewing an object from more than one side/angle generally leads to a more accurate representation of the actual size and shape of the object. But with sensors of differing types and locations, the resolution and distortion resulting from each view can certainly tax the task manager's capability to fully and accurately process the information that it is receiving. Methods have been developed to account for these varying inputs, allowing for reasonably accurate representations of objects, even when viewed by different sensors types and locations [Bolen & Chandrasekar].

An important question that must be answered in this type of network is how to deploy the sensors to achieve maximum coverage of an area so as to collect the most or most relevant data. This deployment of sensors is extremely dependent upon the application and the environment, and many procedures have been proposed to address the issue. These procedures tend to draw comparisons between random deployment of sensors and targeted deployment based on the described algorithms and generally assume no a priori knowledge of the environment, thereby validating a simple objective of area coverage. The variety of approaches to this problem is wide, including a fuzzy logic approach that considers the nearness of sensors to one another [Shu et al] and a virtual force algorithm that strives for a somewhat uniform distribution of sensors by applying “disruptive force” to sensors that are too close and “attractive force” to those that are further apart [Zou & Chakrabarty]. Common characteristics among these models, and others, include detection and communication ranges, which are of utmost importance in defining the coverage area of any sensor network [Wong et al].

Finally, these networks may consist of some hybrid of static and mobile sensors. The most significant advantage here is that the immobile sensors are aligned to maximize coverage

area while the mobile sensors are left in a standby mode. Upon detection of an event, some small number of the static sensors may be tasked to monitor the event initially, but at the same time mobile sensors can be moved into place to take over any tracking functions as soon as possible. This would allow the static resources to quickly return to their sensing activities, reducing or even eliminating potential coverage gaps that could occur if mobile sensors were to “chase” a target [Butler & Rus].

2.1.2 Task Scheduling

We must then schedule the tasks that each sensor performs. Recall that in the initial state, the sensors are assigned to surveillance tasks within their sectors of interest. Here it is important to note that these static surveillance requirements are extremely demanding of resources. Typically all resources are applied to surveillance tasks, until they are drawn away by other higher priority requirements, so as to maintain the maximum coverage of the region [Miranda et al (2)]. But once events occur and the sensors’ attention becomes divided, it is vital that the tasks be scheduled in such a manner so as to reduce or even eliminate any idle time. Task scheduling is generally reserved for the various functions a single sensor may perform, and typically focuses on the tracking of multiple objects within the sensor’s area of responsibility, with the goal being to accurately track as many objects as possible. The constraint on this capability is time. That is, there is some amount of time required to send, receive, and process signals related to each target, with these durations dependent upon the distance between the sensor and the target, and target information must be updated with a particular frequency, else the probability of “losing” the target becomes too large [Miranda et al (1)].

2.1.3 Sensor Functions

While the literature tends to disagree somewhat on the terminology used and the precise set of functions that different sensors are able to perform, military applications can generally be characterized with two primary functions, surveillance and tracking [Miranda et al (3), Orman et al (1)]. Again, surveillance is defined as the general monitoring of a region of

earth or space with the purpose of detecting an event (such as the emergence of a target) and tracking is defined as focused monitoring of an object's (target's) location and other characteristics (direction, velocity, threat, etc.) until such a time when it is no longer of interest. Within these broad areas, though, additional functions may also be defined to more fully characterize the capabilities of a sensor.

2.2 Network Reliability

In general, network reliability is defined by the type of system: serial or parallel. In a serial system, the network is said to fail if any single node within the network fails. Conversely, in a parallel system the network is said to succeed if any node in the network succeeds. In addition, a network may be described as mixed, in which case it incorporates sections of each. Simple examples of each type of network are depicted in Figures 2.1, 2.2 and 2.3.

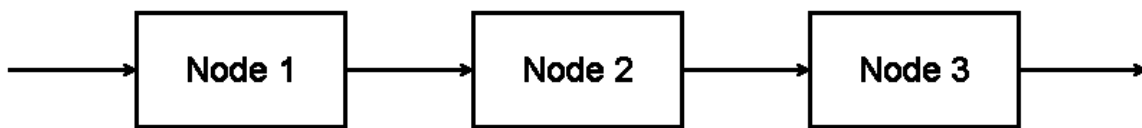


Figure 2.1: Serial Network

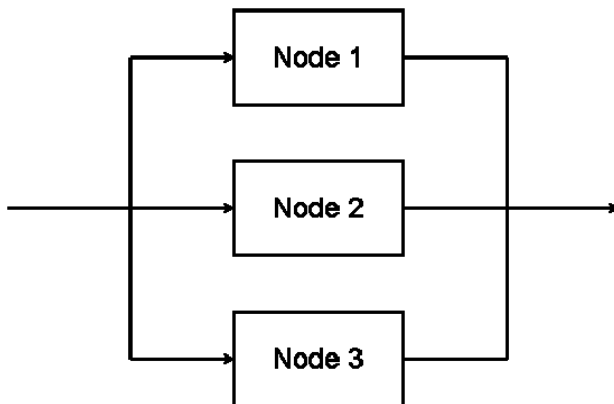


Figure 2.2: Parallel Network

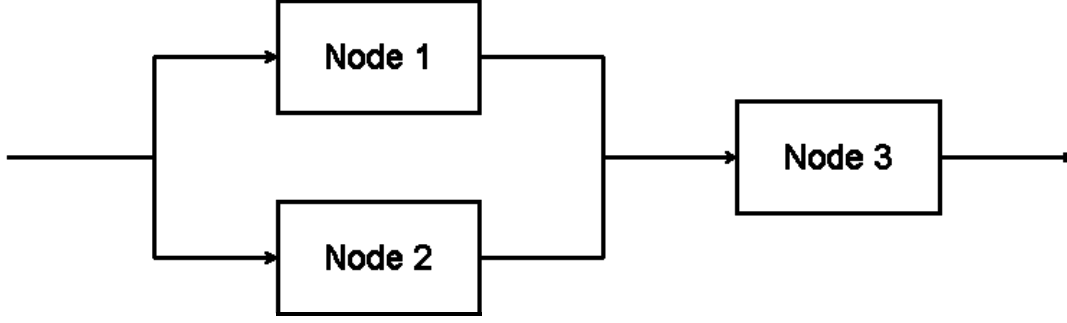


Figure 2.3: Mixed Serial-Parallel Network

For sensor networks, it is important that the failure of a single node not affect the overall success of the network with respect to the successful completion of its assigned tasks. Thus, some level of redundancy should be built in to any sensor allocation scheme to minimize the risk of this network failure [Akyildiz et al]. This risk is captured in the network's reliability measure, which again is written as a function of the individual node probabilities for success. Using the example from Figure 2.3 above, the network reliability is given as:

$$(1 - (1 - P(N_1)) * (1 - P(N_2))) * P(N_3),$$

where $P(N_i)$ represents the probability of success at node i . We can easily expand this calculation to include any number of nodes.

2.3 Resource Allocation Methods

Resource allocation is an optimization problem that seeks to assign a limited set of resources to a specific set of activities in such a manner so as to achieve the most desirable result. The desirability of an allocation scheme is defined by the objective function, but in general terms the problem can be formulated as:

$$\begin{array}{ll}
\text{Maximize} & \sum_{i=1}^n \sum_{j=1}^m c_{ij} x_{ij} \\
& \sum_{i=1}^n x_{ij} \geq 1, \quad \forall j = 1 \dots m \\
& \sum_{j=1}^m x_{ij} \leq 1, \quad \forall i = 1 \dots n \\
\text{subject to} & x_{ij} \in \{0,1\}
\end{array}$$

In this formulation we assign resources i to activities j under the constraints that each activity must be assigned at least one resource and each resource may be assigned to no more than one activity. The assignment of a resource to a specific activity yields some gain (c_{ij}) for the overall system, and it is the total gain that we seek to maximize [Wolsey]. This gain may be virtually any type of reasonable function by which the assignment of a resource to a task may be measured. Often times it is associated with revenue (or cost), but can defined in other terms as well (e.g. system reliability). In many cases a single objective is insufficient, and so we write the function in terms of some utility measure that captures all of our objectives in a single term [Gerkey & Mataric].

The application of resource allocation formulations to various problems has been studied extensively with varying degrees of complexity. For example, machine scheduling has been widely addressed as a resource allocation problem. Beyond the obvious assignment of machines to tasks, variations include the development of a hierarchical task network to aid in constraint development [Harris et al] to the implementation of many different dispatching rules and algorithms designed to create more efficient schedules [Kusiak & Ahn]. These methods have similarly been turned towards the use of assigning teams, be they human or robot, to different tasks, with the objective of optimizing some criterion. In these cases, teams may be assigned to only one task at a time, but multiple teams may be assigned to a single task in an effort to improve the overall probability of completing the assigned task [Liu et al].

3. Methodology

3.1 Initial Model Formulations

3.1.1 Problem Description

We set up the original proposed problem as a job shop (Figure 3.1). In the shop, there are two different types of tasks or missions to perform. One is to kill intruders, and the other is to collect data. It is possible for there to be more than one intruder and, in general, there will be a number of data collecting tasks.

In the intruder case, there are seven stages necessary to complete the process: detect, cue, track, classify, engage, target update, and kill. For data collection, there are six (nothing to kill). Each stage of both processes is probabilistic in that success is not guaranteed.

However, the probability of success at any stage can be enhanced by assigning additional machines to act in parallel to perform the function in the given stage.

3.1.2 Objective

The problem is to assign machines to tasks so as to maximize the overall probability of complete success – all intruders killed and all data collection tasks completed. However, there are a number of alternative objectives; for example, we want to kill intruders with at least some minimum probability while maximizing the data collection probability of success or we want to ensure that the probability of data collection meets some lower bound while maximizing the probability of success of killing all intruders.

We developed a series of models with increasing complexity in order to describe the problem and to begin to understand the true form of the model.

Job Input	Stage 1	Stage 2	Stage 3	Stage 4	Stage 5	Stage 6	Stage 7	Stage 8	Job Output
Threat	Plan	Detect	Cue	Track	Classify	Engage	Target Update	Kill	Results

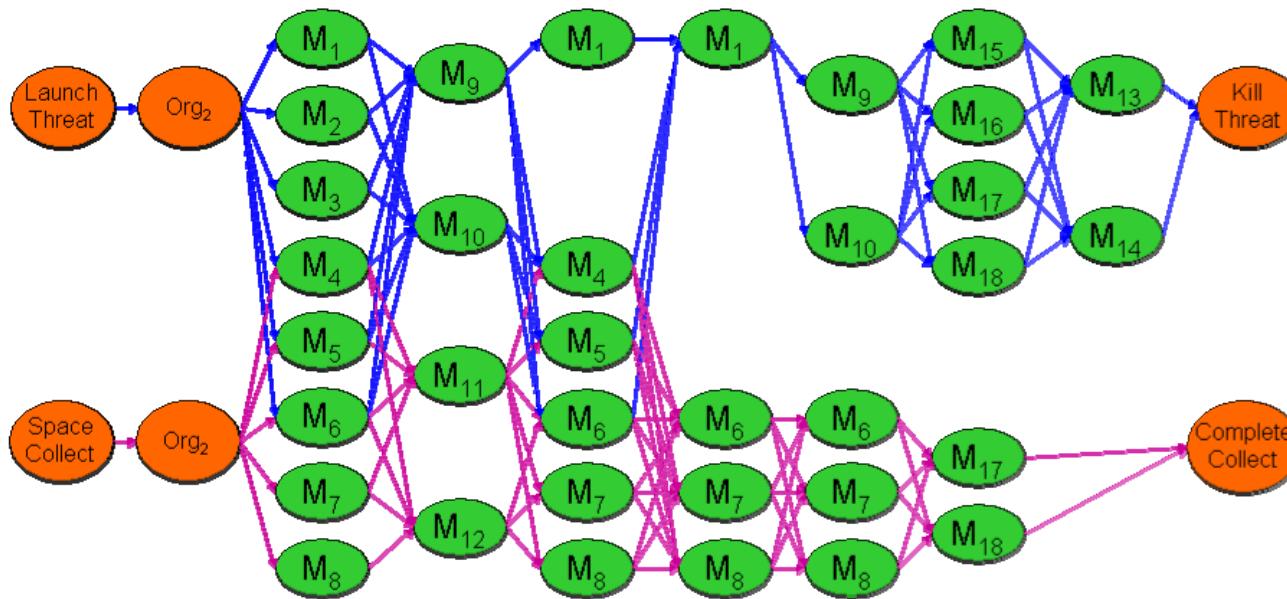


Figure 3.1: Job Shop Model Network

3.1.3 Assumptions and Proposed Methodology

As an initial approximation, the probability of a kill can be seen to be the product of the individual stage probabilities of success over all eight stages, and the probability of success for data collection can be seen to be the product of the individual stage probabilities of success over seven stages.

We know that from calculus when multiplying numbers, this can be done by taking the natural logarithm of each number, adding the logarithms, and then taking the anti-logarithm of the sum to get the product of the original numbers. Thus, taking the natural log of each of the probabilities, then maximizing the sum of the natural logs of the stage probabilities will maximize the probability of a kill. The same is true for the probability of success of the data collection efforts. If the objective is to maximize the overall probability of total success (kill the intruder, gather all data), then maximizing the sum of all of the natural logarithms for all tasks and stages will accomplish the objective.

3.1.4 First Approach: Phase I Model

In order to form an initial model for the allocation of machines, we formulated the problem as an integer program and specifically, a simple assignment problem. Define $x_{ijs} = 1$, if machine i is assigned to intruder (or data collection effort) j at stage s , and zero otherwise. Let lp_{ijs} be the natural log of the probability of success for machine i assigned to intruder (or data collection effort) j at stage s . Assume that only one machine is assigned to a task at any stage, and any machine can be assigned to any task. Then the optimization model can be expressed as:

$$\begin{aligned}
 & \text{Maximize} && \sum_{i,j,s} lp_{ijs} \cdot x_{ijs} \\
 & \text{s.t.} && \sum_i x_{ijs} = 1 \quad \forall j, s && P1 \\
 & && \sum_j x_{ijs} \leq 1 \quad \forall i, s \\
 & && x_{ijs} = 0,1
 \end{aligned}$$

Note that for $P1$ the constraint matrix appears to be unimodular. Thus, if the problem is relaxed so that $x_{ijs} \geq 0$ and solved using linear programming, all decision variables in any feasible solution will be either 0 or 1.

For $i = 1, 4$, $j = 1, 3$, and $s = 1, 2$, the constraint matrix is as follows:

Table 3.1: A 4-machine, 3-intruder, 2-stage Constraint Matrix ($P1$)

i	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	
j	1	1	1	1	2	2	2	2	3	3	3	3	1	1	1	1	2	2	2	2	3	3	3	3	
s	1	1	1	1	1	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2	2	2	2	2	Rhs
Obj	Lp	lp	lp	lp	lp	lp	Lp	lp	lp	lp	Lp	lp	lp	lp	lp	Lp	lp	lp	lp	lp	lp	lp	lp	lp	
	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					=1
																					1	1	1	1	=1
													1				1				1				≤ 1
														1				1				1			≤ 1
															1				1				1		≤ 1
																1					1			1	≤ 1

The formulation has advantages in that there is a certain amount of flexibility in the objective function. For instance, rather than maximizing the overall probability of successfully satisfying all tasks, it is possible to maximize the probability of dealing with the intruder while putting a lower bound constraint on the data gathering requirements. The point is that there are a number of ways that the commander's preferences can be interjected straightforwardly into the model.

Note that problem $P1$ is separable in that each period can be solved independently. However, initially, we assumed that a maximum of only one machine is assigned to any task in any period, where, in fact, multiple asset assignment can increase the probability of success for a task at any stage. That coupled with the fact that there are typically considerably more machines than tasks motivates expanding the model.

3.1.5 Refined Approach: Phase II Model

In order to accommodate multiple machine assignment the decision variable is defined as follows. Define $x_{ikjs} = 1$, if machines i and k are assigned to intruder (data collection task) j , and zero otherwise. Let $A(z, j, s)$ be the set of decision variables that have z in either the 1st or 2nd subscript of x_{ikjs} . Note that the original decision variable x_{ijs} , is still a part of the formulation. Also, it is easy to see how the concept can be expanded to include higher orders of combinations of machine assignment. Model $P2$ is defined as follows:

$$\begin{aligned}
& \text{Maximize} && \sum_{i,j,s} lp_{ijs} \cdot x_{ijs} + \sum_{i,k,j,s} lp_{ikjs} \cdot x_{ikjs} \\
& \text{s.t.} && \sum_i x_{ijs} + \sum_{i,k} x_{ikjs} = 1 \quad \forall j, s && P2 \\
& && \sum_j x_{ijs} + \sum_{j, A(i,j,s)} x_{ikjs} \leq 1 \quad \forall i, s \\
& && x_{ijs} = 0,1 \\
& && x_{ikjs} = 0,1
\end{aligned}$$

Note that for $P2$, it is not clear if the constraint matrix is unimodular or not. Thus, if the problem is relaxed so that $x_{ijs} \geq 0$ and solved using linear programming, it's up for grabs whether we get 0/1 solutions or not. However, since all of the entries in the constraint matrix are 0 or 1, there is always the strong possibility that we will get an all integer solution from a particular problem serendipitously. For simplicity, assume that only intruder 1, in stage 1 can have multiple assignments. For the $i = 1,4$, $j = 1,3$, and $s = 1,2$, the constraint matrix is as follows:

Table 3.2: A 4-machine, 3-intruder, 2-stage Constraint Matrix (P2)

i,k	1	2	3	4	12	13	14	23	24	34	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4		
j	1	1	1	1	1	1	1	1	1	1	2	2	2	2	3	3	3	3	1	1	1	1	2	2	2	2	3	3	3	3		
s	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	rhs	
Obj	lp	lp	Lp	lp	lp	lp	lp	Lp	Lp	Lp	Lp	lp	lp	lp	Lp	lp	lp	lp	lp	lp	Lp	lp	lp	lp	lp	lp	lp	lp	Lp	lp		
	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				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

The objective row in the matrix is missing the subscripts on the lp 's. The subscripts are in the 3 rows above each lp . Again, while the unimodularity of the matrix is questionable, the form of the constraints may enable the solution of the problem using only linear programming (with a small number of cuts). It turns out that our recent work with cuts may also help in the solution of the problem.

3.1.6 Other Model Considerations

Before attempting to formulate the complete model and begin optimization, there were additional considerations that we wanted to incorporate. We needed to address issues such as determining the probability of success, time to complete stages, and dealing with dependencies between stages.

In order to determine the probability of success, we assume that two machines are assigned to the same task at a particular stage. Let the probability of success for the two machines to be p_1 and p_2 respectively. Then, the probability of success is just the probability of both 1 and 2 detecting, or just 1 detecting, or just 2 detecting.

$$P_{success} = p_1 p_2 + p_1(1 - p_2) + p_2(1 - p_1) = p_1 + p_2 - p_1 p_2$$

This is applicable for both one and two machines and can be extended for additional machines as long as all combinations for success are included.

In order to understand, how this can be applied over multiple stages, we will consider a small example with machines A and B in the first stage connected to machine E in the second stage. Calculating the probability of success at E, the ways that success at E can occur are summed. This is consistent with the Law of Total Probability.

Recall that the probability of success for 2 independent machines [M1, M2] is the sum of the ways that success can occur. So by the Law of Total Probability:

Table 3.3: Law of Total Probability for Two Machines

M1 succeeds/M2 fails	+	M2 succeeds/M1 fails	+	M1 and M2 succeed
$\Pr(\text{success}) = p(M1)(1-p(M2))$	+	$p(M2)(1-p(M1))$	+	$p(M1)p(M2)$
$\Pr(\text{success}) = p(M1) + p(M2) - p(M1)p(M2)$				

The probability of success for the first stage for these two independent machines is:

$$P(\text{Success in Stage 1}) = P(A)*(1-P(B)) + P(B)*(1-P(A)) + P(A)*P(B)$$

Simplifying,

$$P(\text{Success in Stage 1}) = P(A) + P(B) - P(A)*P(B)$$

In order to calculate the probability of success of node E in the second stage, the appropriate conditional probabilities must be used associated with each term of the above equation. Therefore,

$$P(E) = P(A)*(1-P(B))*P(E|A) + P(B)*(1-P(A))*P(E|B) + P(A)*P(B)*P(E|A \cap B)$$

$$P(E) = P(A)*P(E|A) - P(A)*P(B)*P(E|A) + P(B)*P(E|B) - P(A)*P(B)*P(E|B) + P(A)*P(B)*P(E|A \cap B)$$

The problem, however, is that a machine's probability of success is dependent on the machine's assignment in the previous phase. It stands to reason that a machine that has just detected an intruder should be able to track that intruder better (providing it has the ability to track) than another machine that was doing something else in the previous phase.

If a machine is successful at a given stage, and is capable of performing at the next stage, then its capability to perform is enhanced at the next stage. Assuming that the data is

available to compute the initial probabilities of success, and the cascading stage probabilities of success, we needed to determine how these elements can be put into the optimization model P2. In that context, problem *P2* becomes a quadratic 0-1 integer program (i.e. the probability of success depends both on the machine assigned to the current phase and on what that machine was assigned to in the previous phase). This seems to be the true nature of the problem. However, we know that 0-1 integer quadratic programs are inherently difficult to solve. So that even if we can solve small problems as a 0-1 integer quadratic program, this approach more than likely will not scale to larger, more realistic problems therefore, we need to investigate additional methods to solve and optimize.

3.1.7 Optimization

Based on the problem with our understanding of the issues at this point in the process, it could be expected that an optimal allocation of resources would have certain recognizable characteristics. In particular, it is clear that once a machine is assigned to a given task at a given stage, it is highly desirable to continue to assign the machine to the same task in succeeding stages (if the machine has the capability to perform in succeeding stages). The observation that drives this conclusion is that the objective of maximizing the total probability of success is computed by multiplying the individual probabilities of success at each stage and each task together. Keeping a machine on the same task from stage to stage tends to increase the individual elements of that product. While it is certainly possible that conditions exist that might negate this conclusion, finding the limits of the conditions will be an element of our work.

Also, all available machines should be assigned at each stage because not to assign an available machine will reduce the probability of success and thus be suboptimal.

Due to the circumstances of the scenario of a given problem, many decisions may be determined prior to optimization. Let's use the problem depicted by Figure 3.1. In stage 2

(detect), machines 1, 2 and 3 have to be assigned to the intruder, machines 7 and 8 have to be assigned to data collection, and machines 4, 5 and 6 can be assigned as needed for stages 2 and 4 (detect and track). Machine 1 has to continue its assignment through stages 4 and 5 (track and classify), and machines 7 and 8 have to continue assignment through stages 4, 5 and 6 (track, classify and engage). Machine 6 has to be assigned to data collection for stages 5 and 6 (classify and engage). Finally, machines 17 and 18 can be assigned as needed in stage 7 (target update), except that at least one of them must be assigned to data collection. Since the probability of success is increased by repeat assignments where possible, assignments once made, will be continued as long as possible. The point is that the optimization of the job shop example can be accomplished by assigning machines 4, 5 and 6, and machines 17 and 18, a total of only 24 possible solutions.

3.1.8 Conclusions

This approach applied to our job shop example could lead to an optimal solution for a small example. Additionally, the probability calculations will definitely be applicable to higher order networks. At this point, we determined that our priority should be to develop the mathematical programming model partially described above using commercial optimization codes in order to test the small network job shop example problem.

3.2 Expanded Sample Model

3.2.1 Structure

In order to develop a mathematical programming model, we sought clarification on the model structure from STRATCOM. In response, we received clarification of the problem and a new network structure which caused us to rethink the initial model formulation.

Sensor allocation needs to occur across four mission areas – missile defense, space surveillance, intelligence collection, and missile warning. For each of these tasks, there are

machines which operate in ten different functions or stages. These machines all have different capabilities. The detailed network diagrams for all four missions are found in Figure 3.2 (Task 1 Network Diagram), Figure 3.3 (Task 2 Network Diagram), Figure 3.4 (Task 3 Network Diagram), and Figure 3.5 (Task 4 Network Diagram).

As seen in the network structure provided in the figures, there are machines working in multiple functions within a task (not a resource sharing conflict) and there are machines working in multiple tasks (a potential resource sharing conflict). Our interest lies in the machines working in multiple tasks in order to tackle the assignability problem.

To identify the machines that would be considered a resource sharing conflict, we look at the machine nodes themselves. Inside each oval representing a machine, we see four tasks – T1, T2, T3, and T4 – each with either a green, magenta or gray background color. The colors mean the following:

Green indicates that the machine is currently configured to support this task in this stage.

Magenta indicates that the machine is not currently configured to support this task in this stage but could be.

Gray indicates that the machine cannot be configured to support this task in this stage.

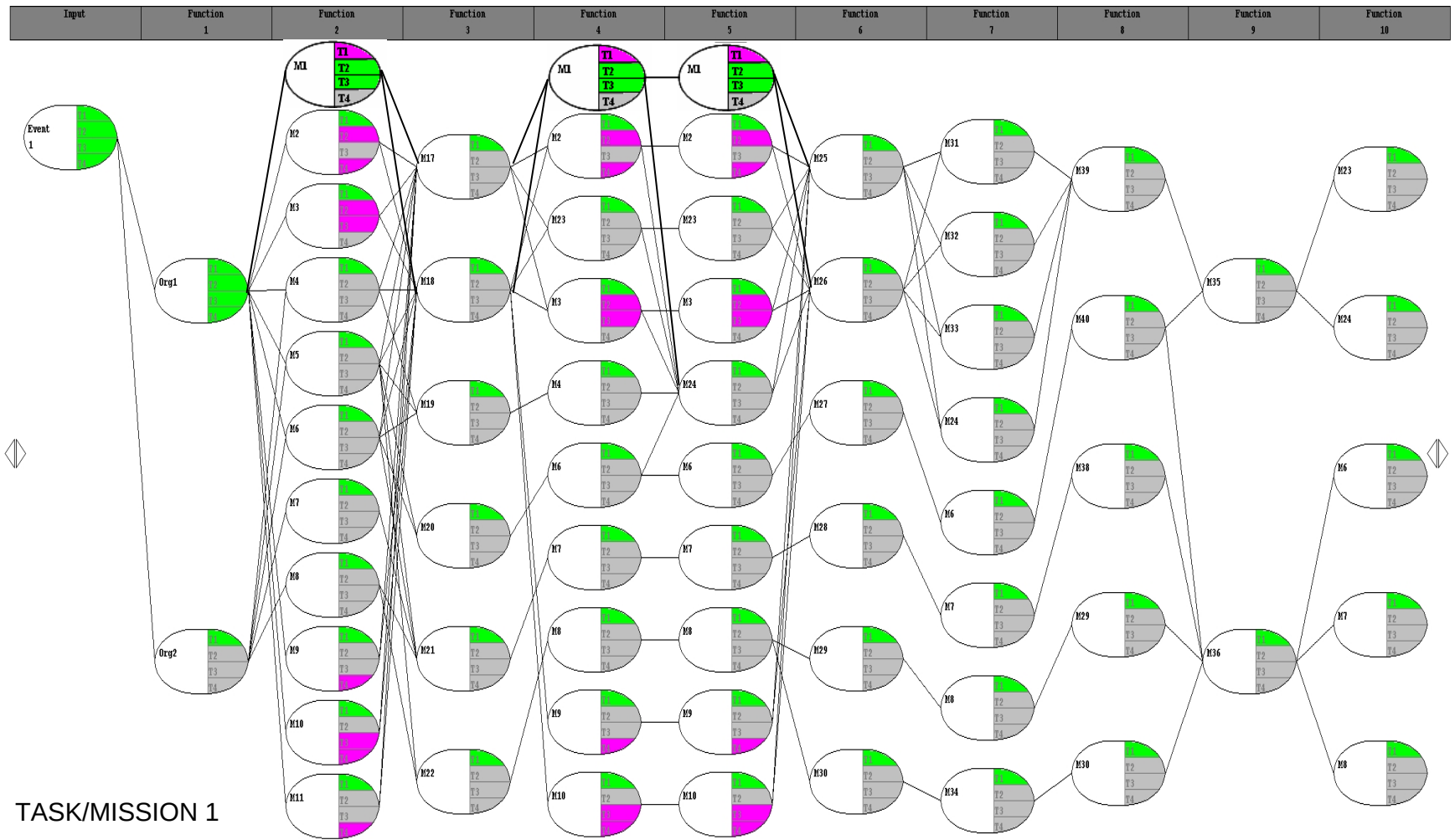


Figure 3.2: Task 1 Network Diagram

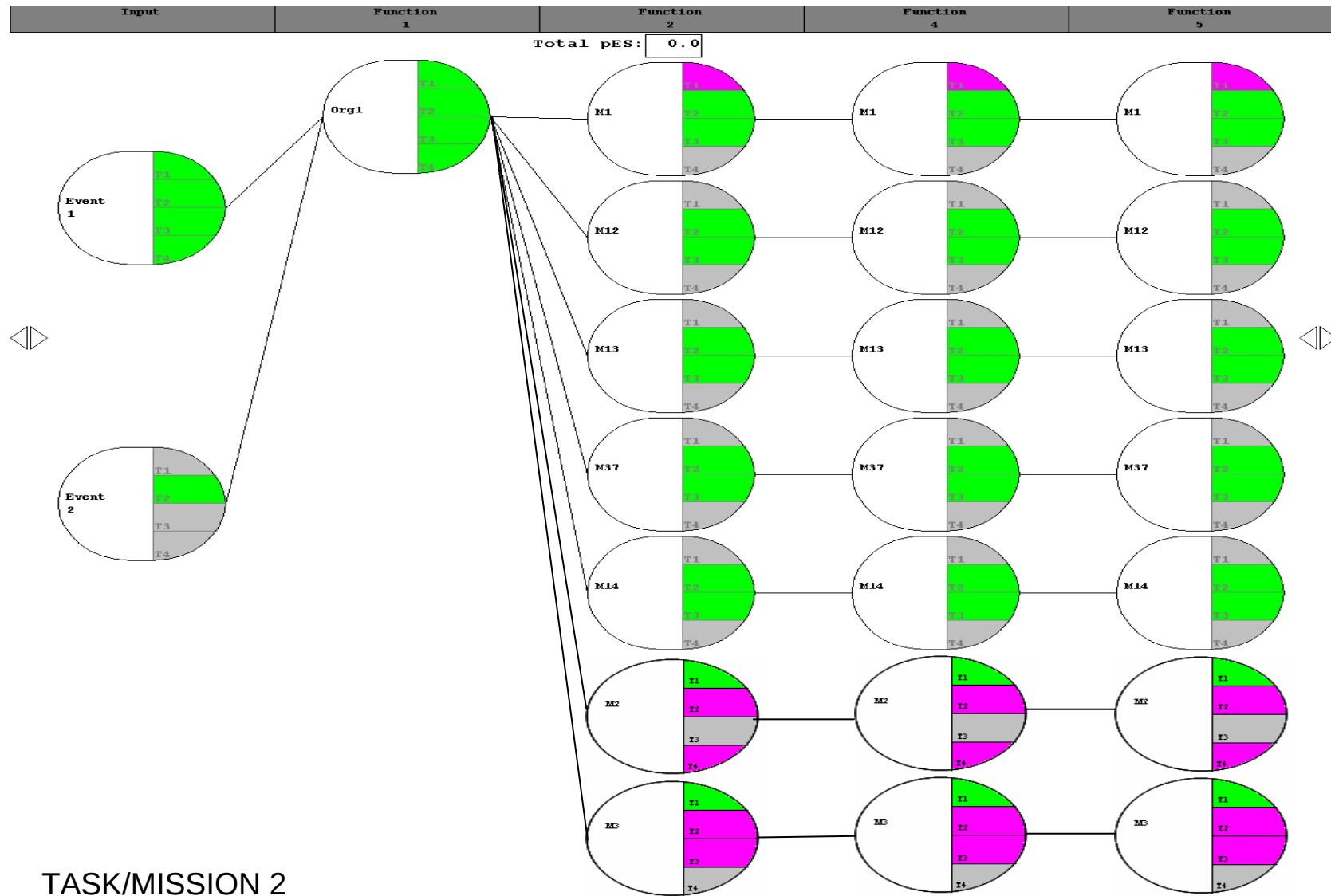
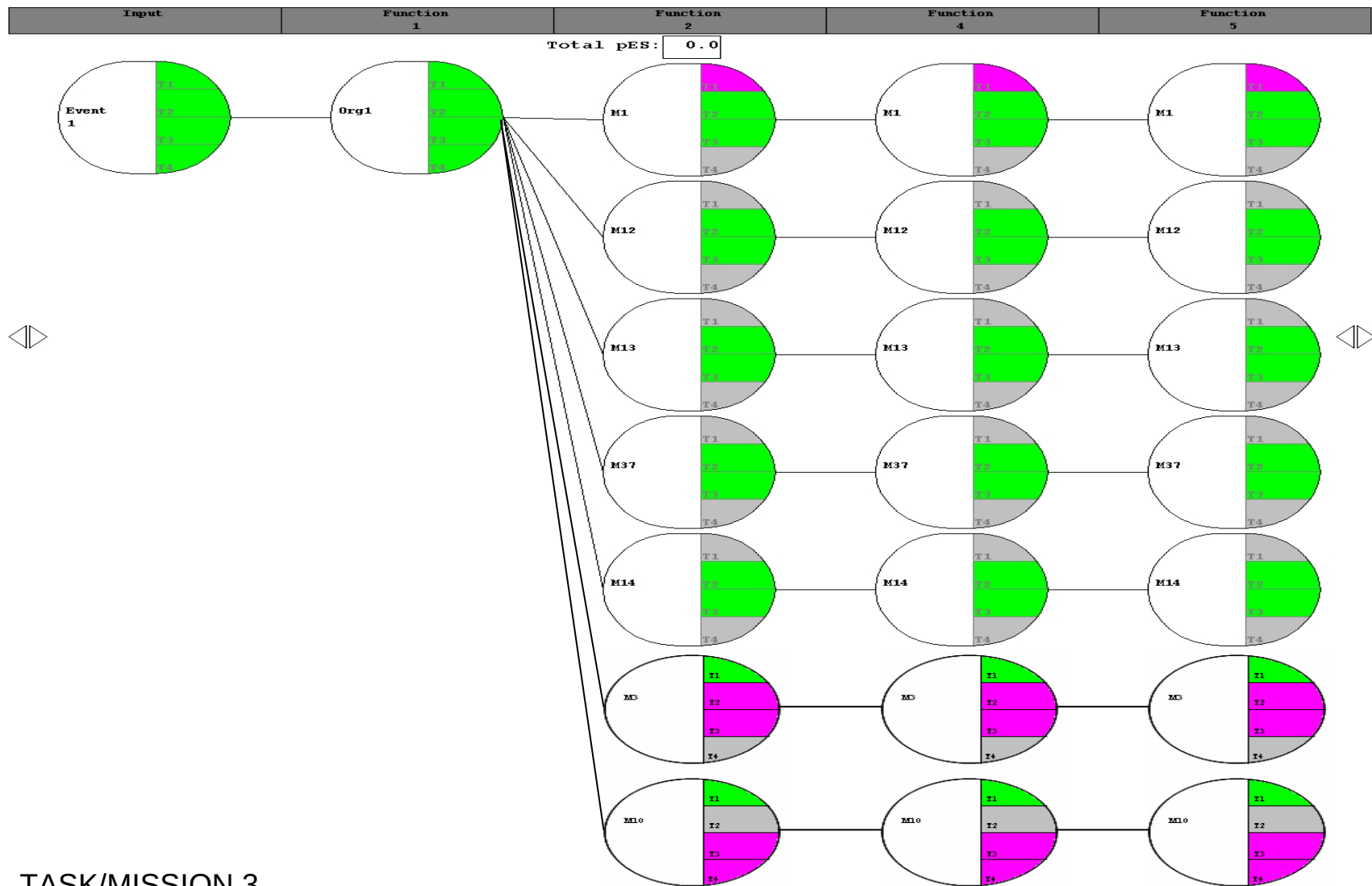
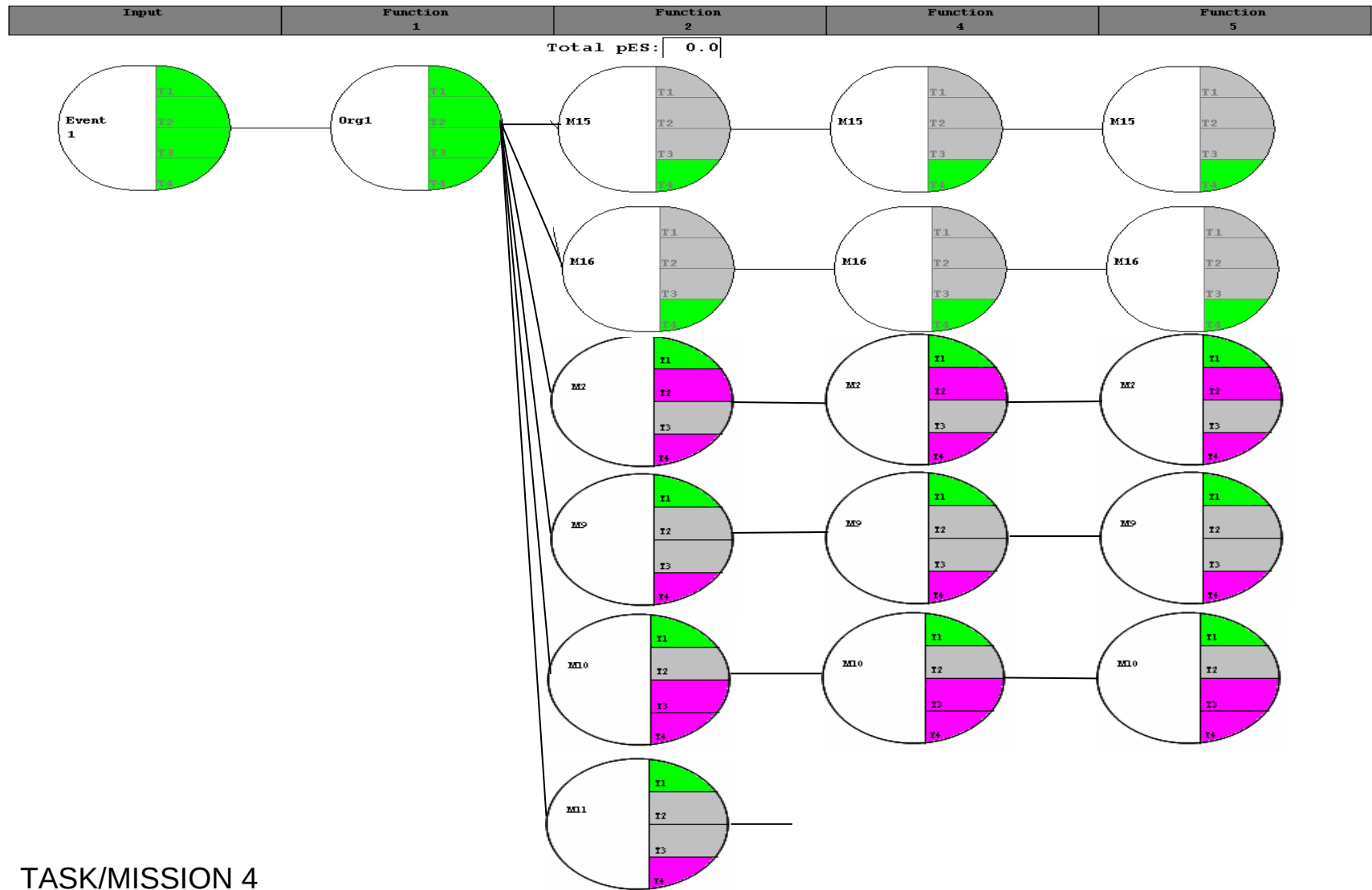


Figure 3.3: Task 2 Network Diagram



TASK/MISSION 3

Figure 3.4: Task 3 Network Diagram



TASK/MISSION 4

Figure 3.5: Task 4 Network Diagram

A machine with a resource sharing conflict then is one that has both green and magenta colors in the node. These machines, in our example network diagram, are M1, M2, M3, M9, M10, and M11. If one of these machines in task 1 was to be reconfigured, it could support one or more of the magenta tasks and therefore would no longer support the currently indicated green task. The resulting change to the configurations represented in the network diagram would be that machine disappearing from the network corresponding to the green task and appearing in the network or networks corresponding to the magenta tasks. With this in mind, all four tasks are affected by the resource sharing.

For example, by reconfiguring M2 and M3 alone, we could take machines away from T1 and reallocate them to T2/T4 and T2/T3 respectively. M2 would then be supporting both T2 and T4 simultaneously and M3 supporting both T2 and T3.

These network diagrams are viewed as snapshots of the machines currently supporting a particular task. Each network corresponds to a different task. The structure that STRATCOM provided is representative of the problem at hand. The number of machines performing each function and the number of conflicting machines among the tasks should not be assumed to be reality but it serves as a starting point for understanding the network and developing a mathematical framework at which to look at the problem of machine assignment or sensor allocation.

3.2.2 Implications

Based on the initial allocation model described in Chapter 3.1, it appeared that this job shop problem would be easily modeled as a mathematical program. More specifically, as we are interested in the assignment of sensors to tasks/mission combinations, the problem seemed to lend itself to an integer programming formulation.

Unfortunately, integer programs can become unwieldy to solve to optimality, as the number of decision variables and constraints grow, and while there are algorithms that

will ultimately find the optimal solution, these can often take significant time to conclude.

With the complexities of the inter-stage connections and the problem of the assignability among the tasks, we determined the complexity of formulating this problem as an integer program was not worth the payoff of the structure and solvability with commercial systems that this type of formulation would provide.

To that end, we intend to investigate and develop both a brute-force method generating all possibilities and determining the optimal using our own developed computer code. This method of total enumeration in order to find optimal assignments is not the preferred method due to the potential time constraints associated with a larger network and a real time decision aid. However, our focus is on the operational planning model and total enumeration will assist us in understanding the complexities and sensitivities of the problem.

For our discussion, we will use the problem instance provided by STRATCOM as our “true” problem and intend to solve this instance of the static problem. We assume that models to solve this instance can be applied to higher order problems with additional assignable assets, different network connections, as well as multiple events.

3.3 Model Formulation

3.3.1 Descriptions

The first task is the most important, as a failure in that area yields the most imminent threat to the United States. Here we must be able to detect the occurrence of some unpredicted event (e.g. a ballistic missile launch), track its path, classify its level of threat and, when necessary, engage and eliminate it. So, it is clear to see that there are in fact several activities that must occur for us to successfully accomplish the mission, and their

duration is certainly dependent upon the event itself, as is the measurement of success. The latter three missions are similar at the outset, but may diverge as an event occurs.

Initially, they consist simply of surveillance of a particular region. As a region is observed, events may occur that require further investigation and action. Regarding space surveillance, an event may be the passage of a space object into a sensor's area of coverage; in intelligence collection, an event could be the movement of a high-ranking foreign official. In the former case, the sequence of functions may be similar to those defined for missile defense—tracking of the object, its classification as a threat (perhaps the object is in a decaying orbit, or may be on a track to possibly impact a communications satellite), and any ensuing activities necessary to deal with it as a threat. In the latter case, the sequence of functions may simply include passing the information to another system for a human operator to analyze, with no further action necessary on the part of the sensor network.

3.3.2 Assumptions

In the static case, considering only one missile defense event, when first looking at the network, it seems as though assets can be assigned to each of the four tasks in stage 2, and then possibly be reassigned to different tasks in stages 4 and 5. This is not the case. In fact, due to the general structure of the network, assignment of assets will only occur in stage 2. Additionally, assignment of assets to task 1 uniquely determines what tasks the rest of the assets will be assigned to.

From inspection of the assets it is clear that assignable assets work either on task 1, or some combination of tasks 2, 3, and 4. Thus, if an asset is not assigned to task 1, it is already uniquely determined which of the other tasks it will work on instead.

The networks associated with tasks 2, 3, and 4 have a simple 'linear' structure in stages 2 through 5: any asset working on these tasks communicates only with itself from one stage

to the next (starting in stage 2). This structure implies that, if an asset is assigned to work on task 2, 3, or 4 at a stage other than stage 2, it will have a probability of success equal to zero since the asset will not receive any information from the previous stage. The only

way any asset working on task 2, 3, or 4 has a probability of success not equal to zero is for the asset to be assigned to one of those tasks at stage 2. Thus any assets that will be assigned to task 2, 3, or 4 must be assigned at stage 2.

The network associated with task 1 does not have a simple 'linear' structure, which seems to imply that reassignment of assets to task 1 can occur in stages 4 and 5, with the assets having a probability of success in each stage not equal to zero. This is the case, but in reality this reassignment will never happen. If an asset were to be reassigned after stage 2, it would have to be taken from task 2, 3, or 4. Reassigning the asset from any of tasks 2, 3, or 4 would not improve the probability of success on tasks 2, 3, or 4. In fact the probability of success on the task the asset was taken from would be equal to the probability of success had the asset not been assigned to the task in the first place.

Assigning an asset to task 1 in either stage 4 or 5 would increase the probability of success, but since it is not logical to reassign an asset from tasks 2, 3, or 4 after stage 2, this reassignment never occurs. Thus even though the task 1 network does not have a simple 'linear' structure, assignment of assets will still only take place in stage 2.

Combining all of the above observations leads to the conclusion that assignment of assets need only occur at stage 2 of each network, and that assignment of assets to task 1 uniquely determines the assignment of assets to the remaining tasks.

For an example, let us look at the assignment of M1 first with an initial assignment to task 1 and then with the assignment to both tasks 2 and 3.

In stage 2, if we assigned M1 to task 1 and considered future assignment to tasks 2 and 3 in stage 4 or 5, it would not be beneficial for tasks 2 or 3. M1 would be assigned to tasks

2 and 3 and be unable to continue processing since it would not have the benefit of receiving the appropriate data from a machine in a previous stage. Since M1 receives data only from itself in a previous stage for task 2 and 3 and M1 was not assigned to task 2 and 3 in that previous stage, M1 would have no data to process in stage 4 or 5 and could not then successfully complete task 2 or 3. Therefore, this reassignment is not productive to the entire network and would not be made. In this case, the initial allocation of machines would remain.

In stage 2, if we assigned M1 to tasks 2 and 3 initially and considered reassignment to task 1 in a subsequent stage, in order to show that the initial allocation should remain, we need to show both that the resulting network is worse for tasks 2 and 3 and that the network of task 1 is no better with the inclusion of this additional machine, M1. The resulting networks for tasks 2 and 3 produce a lower probability of success since we have removed one machine from the process. Recall, in order for the organization to be successful in task types 2 and 3 (4 as well), we assume that, if at least one asset working on a particular task in the last stage of the network, stage 2, succeeds, then that task is considered a success. Therefore, the more machines working in stage 5 of task 2 or 3, the greater the probability of overall success for the associated task.

If we assigned M1 to task 2 and 3 initially, and then removed it in subsequent stages to work on task 1, it is as if we never had that assignment initially since it has no bearing on the outcome of task 2 and 3 in stage 5. Therefore, we might as well have assigned it to task 1 initially.

If we then did not change the assignment of M1 from tasks 2 and 3 to task 1 within the entire process, and simply made the initial assignment of M1 to task 1, it follows from the first discussion that we would not want to make a reassignment and instead should follow through with the initial allocation of machines. Since M1 is representative of all other machines operating in the entire network with options as to which tasks to conduct, we

only need to look at M1 in order to make the same conclusions for all machines.

Additionally, we assume that machines are independent at each stage. Each asset works independently of every other asset at every stage. Thus the assets working in parallel at a given stage increase the probability of success at that particular stage.

We need to define success of a stage. Success at a particular stage is defined as the asset completing its stage objective and passing on information to another asset in the next stage. In addition, each asset has an initial probability of success for every task type at every stage of the network (if the asset cannot work at a particular stage and task type, the probability of success is defined as 0). This initial probability of success is conditional on the machine having received information from exactly one asset that is connected to it in the previous stage.

3.3.3 Optimization of the Network

The optimization of the network occurs in the allocation of assets in the ‘Function 2’ stage of the network. If there are N assets in the ‘Function 2’ stage that need to be allocated to a particular task type (or particular combination of task types), then there are

$$R = [\sum (N \text{ choose } i) \text{ for } i = 1, \dots, N]$$

various networks (i.e. R combinations of asset allocation) that must be considered when searching for the optimal/best asset allocation.

Based on the network assumptions and structure, we would like to maximize the values of the machines associated with task 1 in stage 10 while ensuring that the total end of stage 5 probability of success for tasks 2, 3, and 4 all meet some given threshold. In order to calculate the end of stage 5 probabilities associated with tasks 2, 3, and 4, we determined that the probability of success is

$$P(\text{end of stage 5 success}) = 1 - \Pr(\text{all assets in stage 5 failed})$$

This probability is straightforward to calculate due to the independence of the assets

throughout the stages in tasks 2, 3, and 4. This results in the calculation of one end of stage 5 probability value associated with each task. These three values are the probabilities that we must ensure meet the established threshold for the given task while we attempt to maximize the values of the end of task probabilities associated with task 1 with changes of assignments of the available machines.

3.3.4 Calculation of Network Probabilities

3.3.4.1 Description

We must be able to determine the probability with which a particular resource is able to successfully complete a function or series of functions. We assume that the probability of a sensor accomplishing a function is independent of other stages. However, the probability of overall successful completion of a task with respect to a specific event is determined by the probability of successful accomplishment of all the required functions.

In generating the probabilities needed to fully characterize the initial allocation model, we assume that the probability of successfully completing one function depends not only upon a sensor's efficiency with respect to the function in question, but also upon which assets successfully completed the preceding stage, thereby providing us with the necessary conditional probabilities. The probability of a resource successfully completing its function is calculated using conditional probabilities and the law of total probability.

The model requires the calculation of network probabilities and then the resources may be assigned to specific tasks and functions. Under most conditions, a sensor that is assigned to a specific task and event for one function will be assigned to other functions

for which it is capable within the same task and event. We will assume at this time that all assignable resources can be allocated to either missile defense or to any and all other tasks. Our network may also include “non-assignable” resource, those which are only capable of managing events for one task type. These resources maintain an important role as they contribute to the overall probability of success of the task to which they are assigned.

3.3.4.2 Notation

We let G_i be the set of all resources in stage $s-1$ that are connected to machine M_i in stage s , $A \subseteq G_i$ is the subset of resources that were successful in stage $s-1$, $A' \subseteq G_i$ is the subset of resources that were unsuccessful in stage $s-1$, $A \cup A' = G_i$ and $A \cap A' = \emptyset$. $P_{s-1}(A)$ denotes the overall probability that the resources in A were successful and the resources in A' were not successful in stage $s-1$. Since we assume that each sensor assigned to a stage is independent of all others within that stage, we can calculate $P_{s-1}(A)$ as the product of the probabilities of success and failure as appropriate for each sensor:

$$P_{s-1}(A) = \prod_{a \in A} P_{s-1}(a) * \prod_{a' \in A'} (1 - P_{s-1}(a'))$$

Similarly, we let $P_s(M_i | A)$ denote the conditional probability that machine M_i succeeds in stage s given each asset in A succeeded in stage $s-1$ and each asset in A' failed in stage $s-1$.

Stage 2

In stage 2, we assume that we know the probability that each machine (M_i) succeeds for its assigned task. In other words, the values for $P_2(M_i)$ are given for all M_i that are assigned to stage 2.

Subsequent Stages

In subsequent stages we add a level of complexity to our probability calculations. If we assume that stage 3 is “standard”, then the success of a resource in stage 3 depends upon which machines it received information from in stage 2. Now $A \subseteq G_i$ is the subset of nodes connected to M_i that were successful in the previous stage. Here we assume that we know the conditional probability of M_i succeeding given the set A , $P_3(M_i | A)$. In

other words, we are given the probability that a machine succeeds in stage 3 based on the resources in stage 2 from which it received information. So, using the law of total probability, we can calculate the overall probability of machine M_i successfully completing the stage 3 function as

$$P_3(M_i) = \sum_{A \subseteq G_i} P_3(M_i | A) * P_2(A)$$

Tasks 2, 3, and 4

For tasks $t = 2, 3, 4$ we assume that for each asset at stage 2 we are provided with $P_{t,2}(M_i)$, the probability of success of asset M_i at stage 2 of task t . Additionally we assume that we are provided with $P_{t,4}(M_i | \text{success in stage 2})$, the conditional probability that asset M_i working on task t is successful in stage 4 given that it was successful in stage 2, and $P_{t,3}(M_i | \text{success in stage 4})$, the conditional probability that asset M_i working on task t is successful in stage 5 given that it was successful in stage 4. $P_{t,4}(M_i)$, the probability of success of asset M_i at stage 4 of task t can be calculated as

$$P_{t,4}(M_i) = P_{t,4}(M_i | \text{success in stage 2}) \cdot P_{t,2}(M_i)$$

and similarly we can calculate $P_{t,3}(M_i)$ as

$$P_{t,3}(M_i) = P_{t,3}(M_i | \text{success in stage 4}) \cdot P_{t,4}(M_i)$$

3.3.4.3 Example Calculation

In order to further explain our probability calculations, we will detail a small example. In this example, we consider machines A and B in the first stage connected to machine E in the second stage.

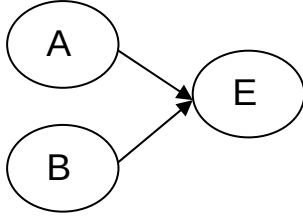


Figure 3.6: Example Network, 3-node 2-stage

Calculating the probability of success at E, the ways that success at E can occur are summed. This is consistent with the Law of Total Probability (reference Table 3.1).

The probability of success for the first stage for these two independent machines is:

$$P_{\text{ncsu}}(\text{Success in Stage 1}) = P(A) * (1 - P(B)) + P(B) * (1 - P(A)) + P(A) * P(B)$$

Simplifying,

$$P_{\text{ncsu}}(\text{Success in Stage 1}) = P(A) + P(B) - P(A) * P(B)$$

In order to calculate the probability of success of node E in the second stage, the appropriate conditional probabilities must be used associated with each term of the above equation. Therefore,

$$P_{\text{ncsu}}(E) = P(A) * (1 - P(B)) * P(E|A) + P(B) * (1 - P(A)) * P(E|B) + P(A) * P(B) * P(E|A \cap B)$$

or,

$$\begin{aligned}
P_{\text{ncsu}}(E) = & P(A)*P(E|A) - P(A)*P(B)*P(E|A) \\
& + P(B)*P(E|B) - P(A)*P(B)*P(E|B) \\
& + P(A)*P(B)*P(E|A \cap B)
\end{aligned} \tag{1}$$

If our example was larger, these calculations would be repeated in each stage for all possible combinations of machines from the previous stage.

While the method for generating network probabilities is reasonably straightforward, the calculations themselves can become quite time-consuming when we begin to consider the many sensors and allocation schemes that must be addressed. The example above was very simple—the probability of successful completion of stage 1 was easy to calculate, and the probability associated with stage 2 was dependent upon which of the two assets assigned to the task were successful in stage 1. However, we would need to generate a different set of probabilities if only resource 1 or resource 2 (but not both) were assigned to the task. So we see that with two sensors available for a particular task, we have three alternate allocations (we disregard the possibility of not assigning any sensors, as this would yield a 0 probability of successful task completion). If there were a third resource available for assignment to a task, then our number of allocation alternatives grows to 7 ($\{1\}, \{2\}, \{3\}, \{1,2\}, \{1,3\}, \{2,3\}, \{1,2,3\}$). Continuing in this manner, it is clear that the number of allocation schemes for a single task is exponential in terms of the number of resources available for that task, in the form of $(2^n - 1)$. And this ignores the existence of other tasks, as well as the possibility of assigning sets of resources to different, or multiple, tasks.

3.3.4.4 Comparison to STRATCOM's Probability Calculations

STRATCOM provided us with their current probability calculations with the intent that we could use them in our model. STRATCOM's calculations for a node's probability of success are based on the failure of its input nodes. For each input node, the probability of success is calculated and multiplied times the current node's probability of success given

that the input was successful. This is done for each input node. STRATCOM then subtracts each of these calculated values to give the probability of failure for the node relative to the input node. These values are multiplied together to provide the current node's overall probability of failure. Subtracting this from 1 yields its probability of success. These calculations have been simplified to exclude the impact of time. STRATCOM's equations are somewhat more complex in order to take into account the timeframe requirements of each node in order for them to be successful.

In the example, STRATCOM's calculations determine that the probability of success for the first stage is:

$$P_{\text{stratcom}}(\text{Success in stage 1}) = 1 - [(1 - P(A)) * (1 - P(B))]$$

Simplifying,

$$P_{\text{stratcom}}(\text{Success in Stage 1}) = P(A) + P(B) - P(A)*P(B)$$

This is consistent with the Law of Total Probability and is equivalent to our calculations.

To determine the probability of success at node E in the example, STRATCOM's method calculates a node's probability of success based on the failure of its input nodes.

Therefore, STRATCOM's formulation states that:

$$P_{\text{stratcom}}(E) = 1 - [1 - P(A)*P(E|A)] * [1 - P(B)*P(E|B)]$$

Expanding the equation,

$$P_{\text{stratcom}}(E) = 1 - 1 + P(A)*P(E|A) + P(B)*P(E|B) - P(A)*P(B)*P(E|A)*P(E|B)$$

or,

$$\begin{aligned}
P_{\text{stratcom}}(E) &= P(A)*P(E|A) \\
&+ P(B)*P(E|B) \\
&- P(A)*P(B)*P(E|A)*P(E|B)
\end{aligned} \tag{2}$$

Comparing Equations (1) and (2) there are two discrepancies to explain:

1) For an event in which one machine succeeds, the probability of success given by eq(2) must take into account the probability that the other machine failed. Comparing the first terms of equations (1) and (2), the eq(1) calculation includes additional terms which takes into account the probability of success of A with the failure of B. This difference accounts for the discrepancies between the first two terms of eq(1) and those of eq(2).

2) In eq(2), when looking at the failure of machines A and B, the possibility of both A and B failing must also be incorporated. There is a probability associated with this outcome and all possibilities need to be enumerated to satisfy the Law of Total Probability. This accounts for the discrepancy in the third term. Success for E can come from: (1) A succeeding and B failing; (2) A failing and B succeeding; or (3) both A and B succeeding.

In order to determine the differences in the calculations, subtract out all like terms between eq(1) and eq(2) examine just the terms that are different in the two formulations and factor out the like terms. The residual terms are given below.

$$\begin{aligned}
\text{Res}(1) &= - P(A)*P(B)*P(E|A) - P(A)*P(B)*P(E|B) + P(A)*P(B)*P(E|A \cap B) \\
&= - P(A)*P(B)[P(E|A) + P(E|B) - P(E|A \cap B)]
\end{aligned}$$

$$\begin{aligned}
\text{Res}(2) &= - P(A)*P(B)*P(E|A)*P(E|B) \\
&= - P(A)*P(B)[P(E|A)*P(E|B)]
\end{aligned}$$

Factoring out $P(A)*P(B)$, the remainder is:

$$\text{Res}'(1) = P(E|A) + P(E|B) - P(E|A \cap B)$$

$$\text{Res}'(2) = P(E|A)*P(E|B)$$

To see what the true differences are, we can calculate the difference with a few example probabilities.

1) If the associated probability numbers are on the order of .5,

$$\text{Res}'(1) = .5 + .5 - .5 = .5$$

$$\text{Res}'(2) = .5 * .5 = .25$$

The difference is on the order of .25

2) If the associated probability numbers are on the order of .9,

$$\text{Res}'(1) = .9 + .9 - .9 = .9$$

$$\text{Res}'(2) = .9 * .9 = .81$$

Difference is on the order of .09

3) If the associated probability numbers are on the order of .95,

$$\text{Res}'(1) = .95 + .95 - .95 = .95$$

$$\text{Res}'(2) = .95 * .95 = .9025$$

The difference is on the order of .0475

We assume that the range of true probabilities is on the order of .95 or greater, and therefore, the differences are relatively small. This is not necessarily a bad thing as the STRATCOM approach is conservative. However, given the investment in assets involved by STRATCOM and the entire DoD, even small differences can be significant and discussion of this subject is worthwhile to continue.

4. Implementation and Results

4.1 Model Description Using Visual Basic

We incorporated the previously discussed methodology into a model coded using Visual Basic. The input to the model is an excel file with the network connections for each stage for each task. The input file that represents the sample network provided by STRATCOM is provided in the Appendix A.

The code reads in the network connections file and takes the array of assignable assets in stage in order to generate all possible combinations of allocations. The data array that represents the general system and a specific allocation array of an allocation to task 1 are then combined in order to create a new data array that represents the network under a specific allocation. This is done for all allocation arrays. The code then uses the modified arrays and randomly generates all the necessary probability values required for that network. These values are stored and printed to an output file.

The code then takes the modified data arrays and associated probability values and calculates the probability of success using the method described in the previous chapter. The probability of success for each machine within a stage is printed out in a separate stage file for task 1. The final output file contains the endstage probabilities of success for all machines in stage 10 for task 1 and all end of task probabilities of success for tasks 2, 3, and 4.

4.2 Model Refinement

4.2.1 Data Requirements

In order to test our model, we needed probability values associated with machines operating in different stages on different tasks as well as conditional probability values for a machine succeeding in one stage given a specific combination of machines succeeded in a previous stage.

Since most of the probability data that STRATCOM has access to is classified we needed to create both a way to randomly generate realistic probability values as well as accurate intervals of probabilities over which we could test our model. STRATCOM stated that they did not care what actual values that we used in order to test our model. However, they wanted to ensure that the probability values were relatively high which is an accurate assumption for these machines.

Initially, we discussed randomly generating machine and conditional probabilities within the range of 0.95 and 1.0. Again, this is considered representative of the machines with which we are concerned. We planned on running the model five times with probability values randomly generated. First run would use values from .95 to 1; second would use values .96 to 1; third would use values .97 to 1; fourth would use values .98 to 1; and fifth would use values from .99 to 1. This would provide STRATCOM a way to gauge the performance of the model with respect to actual parameter values. However, due to the insensitivities found in our results, we changed the proposed method in order to gain more insight into the model. This will be discussed in the next section.

Since we were randomly generating probability values, it was critical to add a feature in our model which would allow us to run multiple replications of the model within the probability range. The model would then calculate an average of these runs to determine an average endstage probability for a machine based on an initial allocation.

Additionally, we had to determine a method for generating the realistic conditional probability values. In order to ensure that a situation did not occur that allowed for fewer input machines to produce a higher probability of success than more input machines, we adjusted the lower range on a conditional probability that includes multiple assets providing information from the previous stage to be at least as great as the highest randomly generated individual probability for each of the assets.

Consider the simple example presented in section 3.3.4.3. In that case there were two resources available in the first stage (denoted here as A and B), and one in the second stage (denoted here as E). If the conditional probabilities of E when only one of the resources in the first stage is successful are generated as $P(E|A) = 0.965$ and $P(E|B) = 0.972$, then the interval for the conditional probability of M when both A and B are successful must be in the interval $[0.972, 1]$.

Again, the reason is that the addition of a successful resource in a preceding stage can never make the probability of success in the following stage worse than if that resource was not successful. This rationale may be carried through to situations where even more resources are feeding information from one stage to the next—always the subsequent conditional probability must be at least as great as the highest conditional probability when one fewer resource was available in the preceding stage.

Additionally, we added a feature in order to enter in different ranges of probabilities for the machines in stage 2. We determined that these values may be less than the range of probabilities associated with all other stages in the network due to the function of that stage. This feature provides us with additional flexibility in the model.

4.2.2 Accuracy

Our probability calculations correspond directly to the Law of Probability. However, we have one additional aspect that we have added into the calculations tied to stages 2, 3 and 4 which incorporates accuracy. There is a probability of success associated with each node and there also needs to be a measure of accuracy associated with the data passed. We feel that this is a critical aspect to the problem and must be addressed because it affects the overall outcome.

Machines working in a stage work independently of one another. Machines in one stage may pass information to a single machine in a subsequent stage. In this subsequent stage, the machine takes the data from all machines that successfully pass the information to it.

The machine would then integrate all the data received from the previous stage, not just pick the best data to process and send along to another stage. The integration of the data from multiple machines in the cueing stage is where the accuracy aspect needs to be incorporated. (An example of accuracy can be found using Global Positioning Systems (GPS). As a GPS acquires more satellites, the reading displayed on your location becomes more accurate as well.) We do this because a machines' ability to successfully track a target depends not only upon the successful passing of the target location information from the previous stage, but also on the accuracy of the data passed.

For example, let us assume that we had a prior probability of success in stage 2, and were able to calculate the appropriate probabilities of success in stage 3 (a standard stage). For a resource in stage 4 (an enhanced stage) to complete its function, it must not only receive the information from the machine(s) that was (were) successful in stage 3, but must also consider the accuracy of the information. For example, whether or not a sensor can locate an object depends primarily upon the region in which it is searching for the object. If the machines in the previous stage produce a very accurate path projection, then it should be relatively simple for a machine in stage 4 to find the object (although there is still a probability associated with this capability). If, on the other hand, the stage 3 projection is not very accurate, the machine in stage 4 may not be able to locate the object at all, resulting in a failure of that machine for that function.

Therefore, to enhance of our probability calculations, we have added a probability mass function to account for accuracy of the information produced during the preceding stage, which is in turn a function of the assets that succeeded in the stage steps previous. Therefore in stage 4, our probability calculation is no longer based on just the single preceding stage but from the success of machines in stage 2 and 3 as well as the accuracy probability mass function from stage 3.

For each asset i in stage 3 of task 1, we assume that we have $f_i(A)$ a probability mass

function of the accuracy of the information as a function of the assets that are successful in stage 2, i.e. $\mathbf{A}$.

Define $\mathbf{H}_j$ as the set of all assets in stage 3 of task 1 that connect to asset M_j in stage 4.

Let $\mathbf{B} \in \mathcal{P}(\mathbf{H}_j)$ denote the assets that are successful at stage 3.

To calculate $P_{1,\mathbf{A}}(M_j)$, the probability of success of asset M_j in stage 4 of task 1, we must first calculate $P_{1,\mathbf{A}}(M_j | \mathbf{B}) \ \forall \ \mathbf{B} \in \mathcal{P}(\mathbf{H}_j)$, since

$$P_{1,\mathbf{A}}(M_j) = \sum_{\mathbf{B} \in \mathcal{P}(\mathbf{H}_j)} P_{1,\mathbf{A}}(M_j | \mathbf{B}) \cdot P(\mathbf{B})$$

If $\mathbf{B}$ has only one element, that is $\mathbf{B} = \{i\}$ then

$$P_{1,\mathbf{A}}(M_j | \mathbf{B}) = \sum_{\mathbf{A} \in \mathcal{P}(\mathbf{G}_i)} P_{1,\mathbf{A}}(M_j | \mathbf{A}) \cdot f_i(\mathbf{A})$$

where we assume that the $P_{1,\mathbf{A}}(M_j | \mathbf{A})$ can be calculated from the data. If $\mathbf{B}$ has two elements i , and $i + 1$, $\mathbf{B} = \{i, i + 1\}$, define $\mathbf{G}' = \mathcal{P}(\mathbf{G}_i) \cup \mathcal{P}(\mathbf{G}_{i+1})$. Then,

$$P_{1,\mathbf{A}}(M_j | \mathbf{B}) = \sum_{\mathbf{a} \in \mathcal{P}(\mathbf{G}')} P_{1,\mathbf{A}}(M_j | \mathbf{a}) \cdot [f_i(\mathbf{a})(1 - f_{i+1}(\mathbf{a})) + (1 - f_i(\mathbf{a}))f_{i+1}(\mathbf{a}) + f_i(\mathbf{a})f_{i+1}(\mathbf{a})]$$

Note that $f_i(\mathbf{a}) = 0$ if $\mathbf{a}$ was not originally a subset of $\mathbf{G}_i$, and similarly $f_{i+1}(\mathbf{a}) = 0$ if $\mathbf{a}$ was not originally a subset of $\mathbf{G}_{i+1}$. These probability calculations can be generalized for $\mathbf{B}$ having more than two elements.

This condition in our code must be set to “True” in order to generate these type of

conditional probabilities. If not, the calculations will be as described in 3.3.4 based on the Law of Total Probability with conditioning occurring back only one stage. Turning this condition off allows us to compare the endstage probability values that result with our calculations vice the results with STRATCOM's method of calculation.

4.3 Model Results

4.3.1 Based on Initial Methodology

Our initial methodology consisted of generating machine success and conditional probabilities within a series of ranges from 0.95 to 1.0. These high probabilities would be representative of the true probabilities of the machines. Using our code, we ran 30 replications in the range of probability values from 0.95 to 1.0 and calculated endstage probability values. The results are included in this chapter and are representative of all runs with probability ranges above .90 either with or without both the inclusion of the probability mass function for accuracy and the smart generation of probabilities function.

Table 4.1: Results for Probability Range .95 to 1.0

Network Allocation	M23	M24	M6	M7	M8	TASK2	TASK3	TASK4
Assigned to task 1	0.952952	0.954944	0.967798	0.963321	0.96451	1	1	1
.1.	0.952952	0.954944	0.967798	0.963321	0.96451	1	1	1
.2.	0.952952	0.954944	0.967798	0.963321	0.96451	1	1	0.999998
.3.	0.952952	0.954944	0.967798	0.963321	0.96451	1	1	1
.9.	0.952953	0.954944	0.967798	0.963321	0.96451	1	1	0.999998
.10.	0.952952	0.954944	0.967798	0.963321	0.96451	1	1	0.999998
.11.	0.952952	0.954944	0.967798	0.963321	0.96451	1	1	0.999998
.1.2.	0.952953	0.954944	0.967798	0.963321	0.96451	0.999998	1	0.999998
.1.3.	0.952953	0.954944	0.967798	0.963321	0.96451	0.999998	0.999998	1
.1.9.	0.952953	0.954944	0.967798	0.963321	0.96451	1	1	0.999998
.1.10.	0.952953	0.954944	0.967798	0.963321	0.96451	1	0.999998	0.999998
.1.11.	0.952952	0.954944	0.967798	0.963321	0.96451	1	1	0.999998
.2.3.	0.952953	0.954944	0.967798	0.963321	0.96451	0.999998	1	0.999998
.2.9.	0.952953	0.954944	0.967798	0.963321	0.96451	1	1	0.99997
.2.10.	0.952953	0.954944	0.967798	0.963321	0.96451	1	1	0.99997
.2.11.	0.952952	0.954944	0.967798	0.963321	0.96451	1	1	0.999972
.3.9.	0.952953	0.954944	0.967798	0.963321	0.96451	1	1	0.999998
.3.10.	0.952953	0.954944	0.967798	0.963321	0.96451	1	0.999998	0.999998

Table 4.1 (continued).

.3.11.	0.952952	0.954944	0.967798	0.963321	0.96451	1	1	0.999998
.9.10.	0.952953	0.954944	0.967798	0.963321	0.96451	1	1	0.999972
.9.11.	0.952953	0.954944	0.967798	0.963321	0.96451	1	1	0.999975
.10.11.	0.952952	0.954944	0.967798	0.963321	0.96451	1	1	0.999974
.1.2.3.	0.952953	0.954944	0.967798	0.963321	0.96451	0.99997	0.999998	0.999998
.1.2.9.	0.952953	0.954944	0.967798	0.963321	0.96451	0.999998	1	0.99997
.1.2.10.	0.952953	0.954944	0.967798	0.963321	0.96451	0.999998	0.999998	0.99997
.1.2.11.	0.952953	0.954944	0.967798	0.963321	0.96451	0.999998	1	0.999972
.1.3.9.	0.952953	0.954944	0.967798	0.963321	0.96451	0.999998	0.999998	0.999998
.1.3.10.	0.952953	0.954944	0.967798	0.963321	0.96451	0.999998	0.99997	0.999998
.1.3.11.	0.952953	0.954944	0.967798	0.963321	0.96451	0.999998	0.999998	0.999998
.1.9.10.	0.952953	0.954944	0.967798	0.963321	0.96451	1	0.999998	0.999972
.1.9.11.	0.952953	0.954944	0.967798	0.963321	0.96451	1	1	0.999975
.1.10.11.	0.952953	0.954944	0.967798	0.963321	0.96451	1	0.999998	0.999974
.2.3.9.	0.952953	0.954944	0.967798	0.963321	0.96451	0.999998	1	0.99997
.2.3.10.	0.952953	0.954944	0.967798	0.963321	0.96451	0.999998	0.999998	0.99997
.2.3.11.	0.952953	0.954944	0.967798	0.963321	0.96451	0.999998	1	0.999972
.2.9.10.	0.952953	0.954944	0.967798	0.963321	0.96451	1	1	0.999602
.2.9.11.	0.952953	0.954944	0.967798	0.963321	0.96451	1	1	0.999622
.2.10.11.	0.952953	0.954944	0.967798	0.963321	0.96451	1	1	0.999621
.3.9.10.	0.952953	0.954944	0.967798	0.963321	0.96451	1	0.999998	0.999972
.3.9.11.	0.952953	0.954944	0.967798	0.963321	0.96451	1	1	0.999975
.3.10.11.	0.952953	0.954944	0.967798	0.963321	0.96451	1	0.999998	0.999974
.9.10.11.	0.952953	0.954944	0.967798	0.963321	0.96451	1	1	0.99965
.1.2.3.9.	0.952953	0.954944	0.967798	0.963321	0.96451	0.99997	0.999998	0.99997
.1.2.3.10.	0.952953	0.954944	0.967798	0.963321	0.96451	0.99997	0.99997	0.99997
.1.2.3.11.	0.952953	0.954944	0.967798	0.963321	0.96451	0.99997	0.999998	0.999972
.1.2.9.10.	0.952953	0.954944	0.967798	0.963321	0.96451	0.999998	0.999998	0.999602
.1.2.9.11.	0.952953	0.954944	0.967798	0.963321	0.96451	0.999998	1	0.999622
.1.2.10.11.	0.952953	0.954944	0.967798	0.963321	0.96451	0.999998	0.999998	0.999621
.1.3.9.10.	0.952953	0.954944	0.967798	0.963321	0.96451	0.999998	0.99997	0.999972
.1.3.9.11.	0.952953	0.954944	0.967798	0.963321	0.96451	0.999998	0.999998	0.999975
.1.3.10.11.	0.952953	0.954944	0.967798	0.963321	0.96451	0.999998	0.99997	0.999974
.1.9.10.11.	0.952953	0.954944	0.967798	0.963321	0.96451	1	0.999998	0.99965
.2.3.9.10.	0.952953	0.954944	0.967798	0.963321	0.96451	0.999998	0.999998	0.999602
.2.3.9.11.	0.952953	0.954944	0.967798	0.963321	0.96451	0.999998	1	0.999622
.2.3.10.11.	0.952953	0.954944	0.967798	0.963321	0.96451	0.999998	0.999998	0.999621
.2.9.10.11.	0.952953	0.954944	0.967798	0.963321	0.96451	1	1	0.994917
.3.9.10.11.	0.952953	0.954944	0.967798	0.963321	0.96451	1	0.999998	0.99965
.1.2.3.9.10.	0.952953	0.954945	0.967798	0.963321	0.96451	0.99997	0.99997	0.999602
.1.2.3.9.11.	0.952953	0.954944	0.967798	0.963321	0.96451	0.99997	0.999998	0.999622
.1.2.3.10.11.	0.952953	0.954944	0.967798	0.963321	0.96451	0.99997	0.99997	0.999621
.1.2.9.10.11.	0.952953	0.954944	0.967798	0.963321	0.96451	0.999998	0.999998	0.994917
.1.3.9.10.11.	0.952953	0.954944	0.967798	0.963321	0.96451	0.999998	0.99997	0.99965
.2.3.9.10.11.	0.952953	0.954944	0.967798	0.963321	0.96451	0.999998	0.999998	0.994917
.1.2.3.9.10.11.	0.952953	0.954945	0.967798	0.963321	0.96451	0.99997	0.99997	0.994917

In order to interpret this table, allocations are listed in the first column of the table. The respective endstage probabilities for each machine in stage 10 in task 1 (M23, M24, M6, M7, and M8) are in the next five columns. The last three columns detail the end of task probability of success for tasks 2, 3, and 4.

As depicted in the table, the results were very insensitive with respect to the initial allocation of machines in stage 2. For this range of probability values, the difference in endstage probabilities for machine 23 between allocating all possible assets and allocating none of the assignable assets to task 1 was $0.952953 - 0.952952 = 0.000001$ which equates to one thousandth of a percent. The same was true for machine 24, $0.954945 - 0.954944 = 0.000001$. Additionally, there were no changes in the endstage probabilities associated with machines 6, 7, or 8. However, there were minor changes in the probabilities associated with tasks 2, 3, and 4. For task 2 and 3, there was a difference of 0.00003 and task 4 has the greatest difference of 0.005083 which is only 0.5 percent.

We sought to determine the reasons behind these results; therefore, we printed out the associated probabilities of success for each stage for each allocation. Due to the high probabilities in a narrow range and the network structure with such redundancy and the type of calculations only conditioning back one stage, we saw the insensitivities in the data within the first two stages. These prints out can be found in Appendix E for greater analysis.

4.3.2 Expanded Probability Ranges

Due to the insensitivities found in our initial results, we determined that we would run greater ranges which would include lower probabilities as well. This would allow us to make additional conclusions and gain more insight into the model. We conducted three additional runs and the summary of the results are included in the following table. The first run was conducted using the probability range from 0.50 to 0.80; the second from

0.80 to 0.95; and the third using a range of 0.50 to 0.80 for the stage 2 probabilities and a range of 0.95 to 1 for all subsequent stages. We listed only the values associated with all assignables or no assignables for task 1 since this was where we observed the greatest difference.

Table 4.2: Summary Results for Expanded Probability Ranges

Probability Values	Allocation: Task 1	M23	M24	M6	M7	M8	TASK2	TASK3	TASK4
.50 - .80	no assignables	0.289696	0.275545	0.106017	0.106974	0.107279	0.894144	0.886744	0.855247
	all assignables	0.305506	0.29102	0.106017	0.106974	0.107279	0.726135	0.721479	0.478094
.80 - .95	no assignables	0.733902	0.738589	0.709047	0.710764	0.701567	0.999547	0.999602	0.99879
	all assignables	0.734122	0.738813	0.709047	0.710764	0.701567	0.987186	0.988255	0.894395
Stage 2: .50 - .80	no assignables	0.909243	0.908133	0.904994	0.898995	0.911903	0.999999	0.999999	0.999992
Subsequent: .95 - 1	all assignables	0.909257	0.908148	0.904994	0.898995	0.911903	0.999591	0.999602	0.98129

The results were still quite insensitive; however, given these expanded ranges and lower probability, we did see a noticeable change in some of the endstage or end of task probabilities. For example, for machines 23 and 24 in the probability range of 0.50 to 0.80 there was an almost two percent change in endstage probabilities. An even greater difference was seen in the end of task probabilities for tasks 2, 3, and 4 for all ranges. For task 4, using probability values between 0.50 and 0.80 produced an 18 percent difference in end of task probabilities of success when comparing the allocations of all possible assignables for task 4 to no assignables for task 4. In this table, the entry for no assignables under the network allocation task 1 column equates to assigning no assignables to task 1, which means all machines were allocated elsewhere i.e. all assignables for task 4. This is because machines are either allocated to task 1 or to all other tasks where they can work.

Even when changing the initial probabilities for stage 2 to a lower range while maintaining the representative range of the true probability values for these machines for all subsequent ranges, we determined that the model was insensitive. Changing these initial probabilities did lower the overall endstage and end of task success probabilities,

as noted in the table below, but the conclusions were still consistent with those made above.

Table 4.3: Summary Results Comparing Lower Probabilities for Stage 2

Probability Values	Allocation: Task 1	M23	M24	M6	M7	M8	TASK2	TASK3	TASK4
All: .95 - 1	no assignables	0.952952	0.954944	0.967798	0.963321	0.96451	1	1	1
	all assignables	0.952953	0.954945	0.967798	0.963321	0.96451	0.99997	0.99997	0.994917
Stage 2: .50 - .80	no assignables	0.909243	0.908133	0.904994	0.898995	0.911903	0.999999	0.999999	0.999992
Subsequent: .95 - 1	all assignables	0.909257	0.908148	0.904994	0.898995	0.911903	0.999591	0.999602	0.98129

4.3.3 Overall Observations

There were a number of contributing factors to the insensitivity of the results. We observed that the insensitivities extend throughout the spectrum of probability values but were especially evident in the 0.90 or higher probability range which included the values that are truly representative of the real probabilities associated with these types of machines. This was only one factor in the resulting insensitive endstage and end of task probability values.

Another contributing factor was the network structure, both the inherent redundancy in stages as well as the connections and limitations throughout the network. For sensor networks, it is important that the failure of a single node not affect the overall success of the network with respect to the successful completion of its assigned tasks. The level of redundancy built into the task 1 network is such that it definitely minimizes the risks of network failure, regardless of the initial allocation of machines.

The network associated with task 1 is a mixed serial-parallel, and therefore there are stages in which many nodes feed into just a couple of nodes. Stage 9 is an example. Due to this feature and our probability calculations which condition only on the contributions of the one previous stage, when we calculate the endstage probability of success for each of the machines in stage 10, each node's probability is only multiplied by the conditional probability associated with one node in the previous stage succeeding and that one node's

probability of success in stage 9. Therefore, for all allocations, the computations for machine 23 at stage 10 differ only by the contributing factor of the probability of success of node 35. Since, for allocations, this success probability is a relatively high number based on the redundancy built into the network in the previous stages, the contribution of this difference across all allocations is limited. For example, let us assume that the probability associated with machine 23 successfully completing its function at stage 10 is 0.985 and the conditional probability of machine 23 succeeding given it successfully received information from machine 35 is 0.97. If one initial allocation results in a probability of success for node 35 to be 0.95 and another allocation results in a probability of 0.955, we see that probability of success of node 23 based on our probability calculations and the law of total probability is:

First allocation: $0.985 * 0.97 * 0.95 = 0.907678$

Second allocation: $0.985 * 0.97 * 0.955 = 0.912455$

We see that the difference is only .5 percent which is, again, very insensitive to the initial allocation of machines in stage 2.

Additionally, due to the structure and limitations of the network as specified, machines 6, 7, and 8 have no difference in resulting endstage probabilities. The network can be followed backwards from these three machines to find that the only contributing nodes in stage 2 are non assignable nodes. Therefore, if we are working in the static case with the only decisions on the allocation of assets being made in stage 2, there is no effect from any assignable node.

For tasks 2, 3, and 4, we see that all are serial networks, and therefore do not have the redundancy inherent in the structure. In a serial system, the network is said to fail if any single node within the system fails. However, for these tasks, each machine, regardless of its linear structure, is a redundancy for all of the other machines in the network

because each machine operating in consecutive stages is its own serial system. In the end, stage 5, only one of the machines needs to be successful in order for the entire task to be successful. This feature contributes to the difference in end of task probability values between difference allocations.

4.3.4 Comparison

Continuing from our comparison between our calculations and those of STRATCOM in the previous chapter which details the difference in probability of success for a small 3-node 2-stage example, we have applied the same probability calculations to the entire sample network.

We calculated the end stage probabilities for the sample network using both the law of total probability and STRATCOM's formula. We repeated these calculations for a range of probability values (.5, .6, .7, .75, .8, .9, .95, .99, and .995). The summary results are included in Table 4.4, and generally mirror the results of the small 3-node 2-stage example.

Table 4.4: Summary Results for Comparison of two Calculations

Probability	STRATCOM		NCSU		Difference		STRATCOM/NCSU		
	Endstage Probabilities		Endstage Probabilities				overall probability of success		
	M23/M24	M6/M7/M8	M23/M24	M6/M7/M8	M23/M24	M6/M7/M8	TASK2	TASK3	TASK4
0.5	0.1984103	0.0154158	0.100813	0.0122213	0.097597	0.0031944	0.413818	0.413818	0.234375
0.6	0.341975	0.0696534	0.207126	0.0552056	0.134848	0.0144479	0.622198	0.622198	0.385344
0.7	0.5001898	0.2257771	0.353201	0.1818429	0.146989	0.0439342	0.813679	0.813679	0.568351
0.75	0.5856827	0.3601204	0.442565	0.29487	0.143118	0.0652504	0.888291	0.888291	0.665771
0.8	0.6750695	0.5269701	0.543097	0.4419849	0.131972	0.0849852	0.943287	0.943287	0.761856
0.9	0.8560473	0.8509968	0.771567	0.7696089	0.08448	0.0813879	0.994606	0.994606	0.926559
0.95	0.9370831	0.9449402	0.890461	0.8979622	0.046622	0.0469779	0.999586	0.999586	0.979658
0.99	0.989417	0.9899882	0.979526	0.9800885	0.009891	0.0098997	0.999999	0.999999	0.999118
0.995	0.9948521	0.9949992	0.989878	0.9900242	0.004974	0.004975	1	1	0.999777

We observed from these comparisons that the greatest difference in endstage probabilities for any given probability value (i.e. .5, .6, ..., .995) occurs when all assignable assets are

allocated to task 1. Additionally, as with the difference calculations from the 3-node 2-stage example, the differences over the entire network for task 1 (i.e. nodes 23/24) have the same form – we get a maximum difference between the two methods at approximately .7 (see Figures 4.1 and 4.2). Finally, for tasks 2, 3, and 4, the two calculations produce the same result. Since these tasks are made up of parallel serial processes our probability calculations are the same.

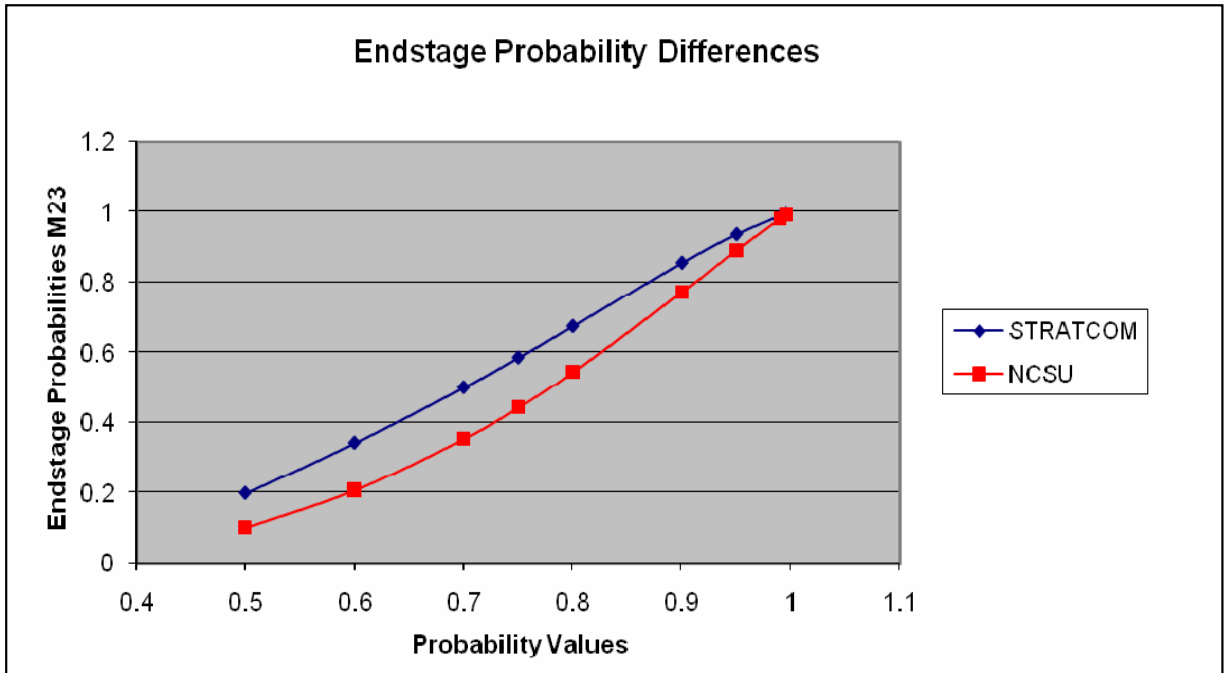


Figure 4.1: Endstage Probabilities for Machines 23/24

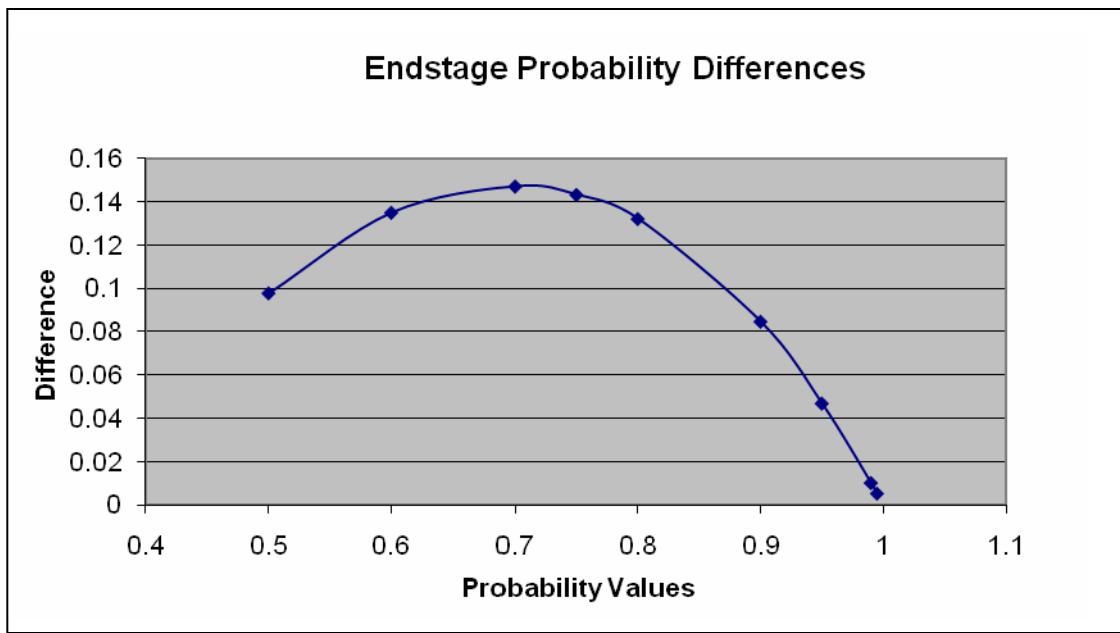


Figure 4.2: Endstage Probability Differences for Machines 23/24

5. Conclusions

We have solved the current sample network provided by STRATCOM for the static case for one event using a model based on the law of total probability. It was both easy to solve and very insensitive.

Using total enumeration, we were able to run all possible allocations for the assignable assets and determine all associated endstage probabilities. From here, we would be able to assign the assets to the appropriate task according to any constraints provided. Constraints could be both lower bounds for tasks 2, 3, and 4 as well as specifications for the endstage probabilities for machines in stage 10 of task 1. The time to run all allocations was on the order of 1.5 seconds. Therefore, for this instance of the problem, even using a brute force method, we were able to determine an optimal solution in a minimal amount of time.

Analyzing the results for the sample network, the insensitivity of the endstage probabilities based on the initial allocations is obvious. Due to this result, a change in allocation of a machine in task 1 has little change in the resulting endstage probabilities. The positive conclusion from this is that due to the redundancy and high probabilities associated with the machines in the network, changes in allocation due to multiple events or machines being down for maintenance will not drastically affect the critical endstage probabilities associated with the missile defense task. However, because of this, a change in allocation to dedicate more assets to a critical missile defense task will not give STRATCOM much added benefit in order to ensure mission success in destroying the incoming threat.

6. Future Research

6.1 Overview

As previously stated, one of the objectives of the problem as a whole is to develop a model with which a decision-maker can quickly and effectively allocate sensor assets to best meet the four tasks. We have a good planning model which is a static model – the next step is to move onto a dynamic execution model.

While our model may perform well for this relatively small, simple network of sensors, we understand the need to scale it to manage a more representative number of assets and events. Furthermore, we have not yet addressed the issue of time—specifically with respect to the frequency of event occurrence and the corresponding amount of time which sensors are unavailable for other functions because of their current allocation.

Additionally, we had discussions about how we envisioned the everyday operations for tasks 2, 3, and 4 for an operational model. There are three cases which must be included: (1) Standard operations would be in support of tasks 2, 3, and 4 until there was a missile launch which would then focus the majority of the effort and therefore machines to task 1. In this case, we need a real time decision aid to support the reallocation of machines; (2) there are times when predictive allocations can be made if STRATCOM knows a task 1 event will be occurring or when a machine will need to be nonoperational for maintenance, for example. This could be scheduled in the operational model and the calculations for reallocation done in advance; and (3) there are times when machines will be allocated to task 1 even if there is currently not an event of task type 1 occurring – that machine may be just waiting for a task 1 event to occur. All instances must be incorporated to a model to handle standard operational issues as well as a real time decision aid for commanders.

6.2 Multiple Events

The initial allocation model only handles one event. It can quickly respond to the inputs of the user and can find an assignment solution. However, we need to have the ability to manage additional events as they occur. For example, if we have created an initial assignment that includes resource allocation to manage the intelligence collection mission within a certain region, but a secondary event occurs in that area, we may need to reallocate some resources from a different task in order to manage the new event, while still maintaining coverage over the mission from which those resources were reallocated. In its most basic sense, this is only marginally different from the initial allocation model as the objective does not change. The difference lies in the probabilities. Whereas we initially developed an allocation scheme in an effort to plan for particular types of events in various regions, we now face a situation where our initial plan may yield a suboptimal or even an infeasible solution.

For example, let us assume that our initial allocation model provided us with a 90% chance of successfully managing intelligence collection in a particular region. This allocation was based on the assumption that only one event would occur within that region at any given time. But, while tracking one event, we detect another in the same region. Based on the resources that we have assigned to that mission, our probability of successfully managing both events may drop to 80%, which may be below our threshold for the mission. By reallocating resources from another mission (say, with an initial level of 95%), we may be able to bring the probability in the first region back up to 90% while the probability in the second region drops only to 90%, and so we have a better solution for the new problem.

Additionally, we know that there are machines that can handle multiple events, tasks and/or functions at a time. However, there are constraints on the number of functions that the different resources can manage. Let's suppose that we have three resources assigned to MD when a launch occurs. Based on this assignment, we can generate a

corresponding probability of success. But now let us assume that a second launch occurs, and that one of our resources assigned to that task can only manage a single event. In this case we essentially create a mirror of the network to manage the second event, but now must decide to which of the two to assign the resource in question. The resulting assignment could yield a success probability that is too low, and so we would then need to reassign an asset from a different task, which would in turn reduce the probability associated with that task.

So, it is clear that the occurrence of multiple events could have a very significant impact on our allocation of assets. Therefore, any model must clarify the appropriate method for dealing with these situations.

6.3 Timing

There are two aspects of time that we must consider when developing our model, solution time and time increments. First, because of the dynamic nature of the problem itself, we must be able to generate a solution quickly, as our resource sets change and as multiple events occur. And second, we must account for the fact that with every time increment, whatever that increment may be, the world view is changing as new events occur, functions are completed, etc.

As stated previously, we must be able to generate a solution quickly. The reasons behind this are twofold. First, with respect to the initial allocation model, we recognize the fact that resource availability can change at any time. That is, some resources may break down or require routine maintenance, others may come into service, or an orbiting sensor may pass out of range of a particular region and into range of another. Second, the dynamic nature of our problem is such that we may need to reallocate our resources because our initial allocation is being taxed and so there is a coverage gap as a result.

Even as we are in the midst of reassigning resources to address this gap, a new event may

be detected requiring even more changes to our sensor allocation scheme, possibly even among the resources that would have been reassigned to address the gap left by the first event. Thus, it becomes even more imperative that we are able to develop a solution method that works quickly, so that we may respond in kind when these situations occur.

As the size of our model grows to become representative of the size of the actual network, the time required to generate solutions will similarly increase. So, we must develop an approach that can find a solution in a “reasonable” amount of time. We essentially have three options: (1) run the reallocation at fixed intervals; (2) run the reallocation in response to events as they occur; or (3) some hybrid of the two.

We discussed the possibility of discretizing the time element into intervals small enough to still accurately complete our functions, but large enough to allow us the ability to find a good solution to the problem of reallocating resources at the end of the time interval. The idea of discretizing time into reasonably small elements and running a resource reallocation at each has some advantages and disadvantages. Most notably, a fixed time interval allows us to develop our model with that interval in mind, essentially adding a constraint to our model. The biggest disadvantage, though, stems from the timing of events. By fixing the time interval, it is possible that several events could occur and overload the network (or a portion of the network) before other resources are reallocated to support the management of these events.

The second approach is the idea of event-driven reallocation. In this case, rather than running our reallocation at specific times, we run it in response to new events. The primary advantage here is that we need not run the reallocation when the network is functioning well and nothing has occurred to impact that level of functioning. The biggest disadvantage emerges from the possibility of multiple events occurring in rapid succession. In this case, it is possible that we are faced with a situation that reallocation may not be complete before a new event occurs and another reallocation is warranted.

A third approach is a hybrid of the first two, allowing for reallocation at somewhat wider fixed intervals than described previously, but with the capability to run the reallocation in response to specific types of events. In this manner only higher priority events may trigger an automatic reallocation, while lower priority events do not. Instead, these lower priority events are initially managed according to the previous allocation, and the reallocation at the end of the current time interval would then consider these new events. The important issues here then become the prioritization of events, as well as an appropriate length for this wider time interval. It must still be small enough so that we are not completely discounting the interval, but large enough that we allow the reallocation to be primarily event-driven.

6.4 Accuracy

In section 3.3.4, Calculation of Network Probabilities, we detail our initial ideas of incorporating accuracy into the model. This is a critical feature which we have taken out of our implementation and results in order to compare our straight calculations with STRATCOM's. In future work pertaining to this allocation problem, we need to determine the correct methodology for incorporating accuracy into any future probability calculations. The issue is to relate the stated probability mass function to an actual physical measure of accuracy around the proposed location of the object at any given time dependant on the number of machines devoted to that object in the current stage.

6.5 Real Time Decision Aid Heuristic

The most important step in continuing this research will revolve around the scaling of the problem to a more representative system—namely a system with several hundred sensors and multiple events within each task. Doing this would make it unreasonable to enumerate all of the possible network assignments, much less calculate the myriad probabilities that would correspond with each potential set of assignments. So it becomes

imperative to develop a method for finding the best network allocation that we are able within the existing time constraints. The intent should be to develop a heuristic approach that will allow us to quickly generate a feasible solution, and then apply techniques to search for improvements to that solution.

This last step will satisfy the final objective of the problem as a whole - to develop a model with which a decision-maker can quickly and effectively allocate sensor assets to best meet the four tasks.

References

1. Akyildiz, I.F., W. Su, Y. Sankarasubramaniam and E. Cayirci (2002). "A survey on sensor networks," *IEEE Communications Magazine*, 40(8), 102-114.
2. Bolen, S.M. and V. Chandrasekar (2003). "Methodology for aligning and comparing spaceborne radar and ground-based radar observations," *Journal of Atmospheric and Oceanic Technology*, 20(5), 647-659.
3. Butler, Z. and D. Rus (2003). "Event-based motion control for mobile-sensor networks," *IEEE Pervasive Computing*, 2(4), 34-42.
4. Gerkey, B.P. and M.J. Mataric (2004). "A formal analysis and taxonomy of task allocation in multi-robot systems," *The International Journal of Robotics Research*, 23(9), 939-954.
5. Ballistic Missile Defense System [BMDS] Booklet, *Global Ballistic Missile Defense: A Layered Integrated Defense*, (Missile Defense Agency, Washington, DC, 1 April 2006).
6. Harris, B., F. Lewis and D.J. Cook (1998). "Machine planning for manufacturing: dynamic resource allocation and on-line supervisory control," *Journal of Intelligent Manufacturing*, 9(5), 413-430.
7. Joint Publication [JP] 3-01, *Countering Air and Missile Threats* (Headquarters, Department of Defense, Washington, DC, 5 February 2007).
8. Joint Publication [JP] 3-14, *Joint Doctrine for Space Operations* (Headquarters, Department of Defense, Washington, DC, 9 August 2002).
9. Joint Publication [JP] 3-60, *Joint Targeting* (Director of Operations, United States Air Force, Washington, DC, 13 April 2007).
10. Kusiak, A. and J. Ahn (1992). "Intelligent scheduling of automated machining systems," *Computer-Integrated Manufacturing Systems*, 5(1), 3-14.
11. Liu, Y., M.A. Simaan and J.B. Cruz Jr. (2003). "An application of dynamic Nash task assignment strategies to multi-team military air operations," *Automatica*, 39(8), 1469-1478.
12. Miranda, S.L.C., C.J. Baker, K. Woodbridge and H.D. Griffiths (2004). "Phased array radar resource management: a comparison of scheduling algorithms," *Proceedings of the IEEE Radar Conference*, 79-84.

13. Miranda, S.L.C., C.J. Baker, K. Woodbridge and H.D. Griffiths (2007). "Fuzzy logic approach for prioritization of radar tasks and sectors of surveillance in multifunction radar," *IET Radar, Sonar and Navigation*, 1(2), 131-141.
14. Miranda, S.L.C., C.J. Baker, K. Woodbridge and H.D. Griffiths (2007). "Comparison of scheduling algorithms for multifunction radar," *IET Radar, Sonar and Navigation*, 1(6), 414-424.
15. Orman, A.J., C.N. Potts, A.K. Shahani and A.R. Moore (1996). "Scheduling for a multifunction phased array radar system," *European Journal of Operational Research*, 90(1), 13-25.
16. Orman, A.J., A.K. Shahani and A.R. Moore (1998). "Modelling for the control of a complex radar system," *Computers and Operations Research*, 25(3), 239-249.
17. Shu, H., Q. Liang and J. Gao (2007). "Distributed sensor networks deployment using fuzzy logic systems," *International Journal of Wireless Information Networks*, 14(3), 163-173.
18. Wolsey, L.A. (1998). *Integer Programming*, New York, NY: John Wiley & Sons, Inc.
19. Wong, T., T. Tsuchiya and T. Kikuno (2004). "A self-organizing technique for sensor placement in wireless micro-sensor networks," *Proceedings of the 18th International Conference on Advanced Information Networking and Application*, 1, 78-83.
20. Zou, Y. and K. Chakrabarty (2003). "Sensor deployment and target localization based on virtual forces," *22nd Annual Joint Conference of the IEEE Computer and Communications Societies*, 2, 1293-1303.

Appendices

Appendix A: Network Connections File

TASK1										
Stage2										
1										
2										
3										
4										
5										
6										
7										
8										
9										
10										
11										
endStage										
Stage3										
17	1	2	3	4	5	6	9	10	11	
18	1	2	3	4	5	6	9	10	11	
19	4	5	6							
20	5	6								
21	5	6	7	8						
22	5	8								
endStage										
Stage4										
1	17	18								
2	17	18								
23	17	18								
3	17	18								
4	19									
6	20									
7	21									
8	22									
9	18									
10	18									
endStage										
Stage5										
1	1									
2	2									
23	23									
3	3									
24	1	2	23	3	4	6				
6	6									
7	7									
8	8									
9	9									

10	10						
endStage							
Stage6							
25	1	2	23	3	24	9	10
26	1	2	23	3	24	9	10
27	6						
28	7						
29	8						
30	8						
endStage							
Stage7							
31	25	26					
32	25	26					
33	25	26					
24	25	26					
6	27						
7	28						
8	29						
34	30						
endStage							
Stage8							
39	31	32	33	24			
40	6						
38	7						
29	8						
30	34						
endStage							
Stage9							
35	39	40					
36	40	38	29	30			
endStage							
Stage10							
23	35						
24	35						
6	36						
7	36						
8	36						
endStage							
TASK2							
Stage2							
1							
12							
13							
37							

14	
2	
3	
endStage	
Stage4	
1	1
12	12
13	13
37	37
14	14
2	2
3	3
endStage	
Stage5	
1	1
12	12
13	13
37	37
14	14
2	2
3	3
endStage	
TASK3	
Stage2	
1	
12	
13	
37	
14	
3	
10	
endStage	
Stage4	
1	1
12	12
13	13
37	37
14	14
3	3
10	10
endStage	
Stage5	
1	1
12	12

13	13
37	37
14	14
3	3
10	10
endStage	
TASK4	
Stage2	
15	
16	
2	
9	
10	
11	
endStage	
Stage4	
15	15
16	16
2	2
9	9
10	10
11	11
endStage	
Stage5	
15	15
16	16
2	2
9	9
10	10
11	11
endStage	

Appendix B: Results – Probability Range 0.50 to 0.80

NETWORK ALLOCATION	23	24	6	7	8	TASK2	TASK3	TASK4
Assigned to task 1	0.289696	0.275545	0.106017	0.106974	0.107279	0.894144	0.886744	0.855247
.1.	0.299412	0.285051	0.106017	0.106974	0.107279	0.855594	0.843348	0.855247
.2.	0.299177	0.284917	0.106017	0.106974	0.107279	0.854053	0.886744	0.801458
.3.	0.299305	0.28501	0.106017	0.106974	0.107279	0.854308	0.849461	0.855247
.9.	0.296079	0.281749	0.106017	0.106974	0.107279	0.894144	0.886744	0.8035
.10.	0.29608	0.28182	0.106017	0.106974	0.107279	0.894144	0.848664	0.802241
.11.	0.29047	0.276338	0.106017	0.106974	0.107279	0.894144	0.886744	0.794523
.1.2.	0.302718	0.288332	0.106017	0.106974	0.107279	0.801092	0.843348	0.801458
.1.3.	0.302705	0.288288	0.106017	0.106974	0.107279	0.801072	0.791841	0.855247
.1.9.	0.301761	0.287332	0.106017	0.106974	0.107279	0.855594	0.843348	0.8035
.1.10.	0.301673	0.287275	0.106017	0.106974	0.107279	0.855594	0.790442	0.802241
.1.11.	0.299692	0.285348	0.106017	0.106974	0.107279	0.855594	0.843348	0.794523
.2.3.	0.302502	0.288148	0.106017	0.106974	0.107279	0.799221	0.849461	0.801458
.2.9.	0.301634	0.287262	0.106017	0.106974	0.107279	0.854053	0.886744	0.730387
.2.10.	0.301522	0.287189	0.106017	0.106974	0.107279	0.854053	0.848664	0.728851
.2.11.	0.299414	0.285163	0.106017	0.106974	0.107279	0.854053	0.886744	0.718377
.3.9.	0.301767	0.287364	0.106017	0.106974	0.107279	0.854308	0.849461	0.8035
.3.10.	0.301604	0.287238	0.106017	0.106974	0.107279	0.854308	0.798808	0.802241
.3.11.	0.299539	0.285251	0.106017	0.106974	0.107279	0.854308	0.849461	0.794523
.9.10.	0.299774	0.285394	0.106017	0.106974	0.107279	0.894144	0.848664	0.731151
.9.11.	0.296527	0.282215	0.106017	0.106974	0.107279	0.894144	0.886744	0.721685
.10.11.	0.296452	0.282196	0.106017	0.106974	0.107279	0.894144	0.848664	0.719265
.1.2.3.	0.304235	0.289801	0.106017	0.106974	0.107279	0.726135	0.791841	0.801458
.1.2.9.	0.303925	0.289486	0.106017	0.106974	0.107279	0.801092	0.843348	0.730387
.1.2.10.	0.30381	0.289393	0.106017	0.106974	0.107279	0.801092	0.790442	0.728851
.1.2.11.	0.302822	0.288442	0.106017	0.106974	0.107279	0.801092	0.843348	0.718377
.1.3.9.	0.303946	0.289478	0.106017	0.106974	0.107279	0.801072	0.791841	0.8035
.1.3.10.	0.3038	0.289354	0.106017	0.106974	0.107279	0.801072	0.721479	0.802241
.1.3.11.	0.302817	0.288405	0.106017	0.106974	0.107279	0.801072	0.791841	0.794523
.1.9.10.	0.303238	0.288788	0.106017	0.106974	0.107279	0.855594	0.790442	0.731151
.1.9.11.	0.301941	0.287525	0.106017	0.106974	0.107279	0.855594	0.843348	0.721685
.1.10.11.	0.301813	0.28742	0.106017	0.106974	0.107279	0.855594	0.790442	0.719265
.2.3.9.	0.303824	0.289395	0.106017	0.106974	0.107279	0.799221	0.849461	0.730387
.2.3.10.	0.303661	0.289258	0.106017	0.106974	0.107279	0.799221	0.798808	0.728851
.2.3.11.	0.302589	0.288237	0.106017	0.106974	0.107279	0.799221	0.849461	0.718377
.2.9.10.	0.303158	0.288746	0.106017	0.106974	0.107279	0.854053	0.848664	0.631236
.2.9.11.	0.301796	0.287433	0.106017	0.106974	0.107279	0.854053	0.886744	0.618436
.2.10.11.	0.301653	0.287321	0.106017	0.106974	0.107279	0.854053	0.848664	0.615385
.3.9.10.	0.303238	0.288795	0.106017	0.106974	0.107279	0.854308	0.798808	0.731151
.3.9.11.	0.30192	0.287522	0.106017	0.106974	0.107279	0.854308	0.849461	0.721685
.3.10.11.	0.301727	0.287364	0.106017	0.106974	0.107279	0.854308	0.798808	0.719265
.9.10.11.	0.300002	0.285623	0.106017	0.106974	0.107279	0.894144	0.848664	0.619178
.1.2.3.9.	0.305009	0.290535	0.106017	0.106974	0.107279	0.726135	0.791841	0.730387

.1.2.3.10.	0.304868	0.290413	0.106017	0.106974	0.107279	0.726135	0.721479	0.728851
.1.2.3.11.	0.304282	0.289851	0.106017	0.106974	0.107279	0.726135	0.791841	0.718377
.1.2.9.10.	0.304697	0.29024	0.106017	0.106974	0.107279	0.801092	0.790442	0.631236
.1.2.9.11.	0.304005	0.289572	0.106017	0.106974	0.107279	0.801092	0.843348	0.618436
.1.2.10.11.	0.303871	0.289456	0.106017	0.106974	0.107279	0.801092	0.790442	0.615385
.1.3.9.10.	0.304706	0.290221	0.106017	0.106974	0.107279	0.801072	0.721479	0.731151
.1.3.9.11.	0.304027	0.289563	0.106017	0.106974	0.107279	0.801072	0.791841	0.721685
.1.3.10.11.	0.303862	0.289419	0.106017	0.106974	0.107279	0.801072	0.721479	0.719265
.1.9.10.11.	0.303337	0.28889	0.106017	0.106974	0.107279	0.855594	0.790442	0.619178
.2.3.9.10.	0.304624	0.290167	0.106017	0.106974	0.107279	0.799221	0.798808	0.631236
.2.3.9.11.	0.303894	0.289467	0.106017	0.106974	0.107279	0.799221	0.849461	0.618436
.2.3.10.11.	0.303716	0.289314	0.106017	0.106974	0.107279	0.799221	0.798808	0.615385
.2.9.10.11.	0.303253	0.288842	0.106017	0.106974	0.107279	0.854053	0.848664	0.478094
.3.9.10.11.	0.303329	0.288888	0.106017	0.106974	0.107279	0.854308	0.798808	0.619178
.1.2.3.9.10.	0.305476	0.290989	0.106017	0.106974	0.107279	0.726135	0.721479	0.631236
.1.2.3.9.11.	0.305051	0.290579	0.106017	0.106974	0.107279	0.726135	0.791841	0.618436
.1.2.3.10.11.	0.3049	0.290445	0.106017	0.106974	0.107279	0.726135	0.721479	0.615385
.1.2.9.10.11.	0.304748	0.290292	0.106017	0.106974	0.107279	0.801092	0.790442	0.478094
.1.3.9.10.11.	0.304758	0.290274	0.106017	0.106974	0.107279	0.801072	0.721479	0.619178
.2.3.9.10.11.	0.304672	0.290216	0.106017	0.106974	0.107279	0.799221	0.798808	0.478094
.1.2.3.9.10.11.	0.305506	0.29102	0.106017	0.106974	0.107279	0.726135	0.721479	0.478094

Appendix C: Results – Probability Range 0.80 to 0.95

NETWORK ALLOCATION	23	24	6	7	8	TASK2	TASK3	TASK4
Assigned to task 1	0.733902	0.738589	0.709047	0.710764	0.701567	0.999547	0.999602	0.99879
.1.	0.73404	0.73873	0.709047	0.710764	0.701567	0.998616	0.998804	0.99879
.2.	0.734018	0.738706	0.709047	0.710764	0.701567	0.998597	0.999602	0.996311
.3.	0.734035	0.738724	0.709047	0.710764	0.701567	0.998654	0.99875	0.99879
.9.	0.734034	0.738723	0.709047	0.710764	0.701567	0.999547	0.999602	0.996349
.10.	0.73402	0.738708	0.709047	0.710764	0.701567	0.999547	0.998744	0.996262
.11.	0.733903	0.73859	0.709047	0.710764	0.701567	0.999547	0.999602	0.996296
.1.2.	0.734073	0.738763	0.709047	0.710764	0.701567	0.995663	0.998804	0.996311
.1.3.	0.734078	0.738768	0.709047	0.710764	0.701567	0.995872	0.996264	0.99879
.1.9.	0.734084	0.738774	0.709047	0.710764	0.701567	0.998616	0.998804	0.996349
.1.10.	0.734074	0.738765	0.709047	0.710764	0.701567	0.998616	0.996245	0.996262
.1.11.	0.73404	0.73873	0.709047	0.710764	0.701567	0.998616	0.998804	0.996296
.2.3.	0.734071	0.73876	0.709047	0.710764	0.701567	0.995867	0.99875	0.996311
.2.9.	0.734072	0.738761	0.709047	0.710764	0.701567	0.998597	0.999602	0.98884
.2.10.	0.734062	0.738751	0.709047	0.710764	0.701567	0.998597	0.998744	0.988607
.2.11.	0.734018	0.738706	0.709047	0.710764	0.701567	0.998597	0.999602	0.988712
.3.9.	0.734081	0.738771	0.709047	0.710764	0.701567	0.998654	0.99875	0.996349
.3.10.	0.734075	0.738765	0.709047	0.710764	0.701567	0.998654	0.996048	0.996262
.3.11.	0.734035	0.738724	0.709047	0.710764	0.701567	0.998654	0.99875	0.996296
.9.10.	0.734075	0.738765	0.709047	0.710764	0.701567	0.999547	0.998744	0.988764
.9.11.	0.734035	0.738724	0.709047	0.710764	0.701567	0.999547	0.999602	0.988809
.10.11.	0.73402	0.738709	0.709047	0.710764	0.701567	0.999547	0.998744	0.988538
.1.2.3.	0.734095	0.738785	0.709047	0.710764	0.701567	0.987186	0.996264	0.996311
.1.2.9.	0.7341	0.738791	0.709047	0.710764	0.701567	0.995663	0.998804	0.98884
.1.2.10.	0.734094	0.738785	0.709047	0.710764	0.701567	0.995663	0.996245	0.988607
.1.2.11.	0.734073	0.738763	0.709047	0.710764	0.701567	0.995663	0.998804	0.988712
.1.3.9.	0.734104	0.738794	0.709047	0.710764	0.701567	0.995872	0.996264	0.996349
.1.3.10.	0.734098	0.738788	0.709047	0.710764	0.701567	0.995872	0.988255	0.996262
.1.3.11.	0.734078	0.738768	0.709047	0.710764	0.701567	0.995872	0.996264	0.996296
.1.9.10.	0.7341	0.738791	0.709047	0.710764	0.701567	0.998616	0.996245	0.988764
.1.9.11.	0.734084	0.738775	0.709047	0.710764	0.701567	0.998616	0.998804	0.988809
.1.10.11.	0.734074	0.738765	0.709047	0.710764	0.701567	0.998616	0.996245	0.988538
.2.3.9.	0.734098	0.738787	0.709047	0.710764	0.701567	0.995867	0.99875	0.98884
.2.3.10.	0.734094	0.738783	0.709047	0.710764	0.701567	0.995867	0.996048	0.988607
.2.3.11.	0.734071	0.73876	0.709047	0.710764	0.701567	0.995867	0.99875	0.988712
.2.9.10.	0.734093	0.738783	0.709047	0.710764	0.701567	0.998597	0.998744	0.965646
.2.9.11.	0.734072	0.738762	0.709047	0.710764	0.701567	0.998597	0.999602	0.96583
.2.10.11.	0.734062	0.738751	0.709047	0.710764	0.701567	0.998597	0.998744	0.96501
.3.9.10.	0.7341	0.73879	0.709047	0.710764	0.701567	0.998654	0.996048	0.988764
.3.9.11.	0.734081	0.738771	0.709047	0.710764	0.701567	0.998654	0.99875	0.988809
.3.10.11.	0.734075	0.738765	0.709047	0.710764	0.701567	0.998654	0.996048	0.988538
.9.10.11.	0.734075	0.738765	0.709047	0.710764	0.701567	0.999547	0.998744	0.965486
.1.2.3.9.	0.734114	0.738804	0.709047	0.710764	0.701567	0.987186	0.996264	0.98884

.1.2.3.10.	0.73411	0.7388	0.709047	0.710764	0.701567	0.987186	0.988255	0.988607
.1.2.3.11.	0.734095	0.738785	0.709047	0.710764	0.701567	0.987186	0.996264	0.988712
.1.2.9.10.	0.734112	0.738803	0.709047	0.710764	0.701567	0.995663	0.996245	0.965646
.1.2.9.11.	0.7341	0.738791	0.709047	0.710764	0.701567	0.995663	0.998804	0.96583
.1.2.10.11.	0.734094	0.738785	0.709047	0.710764	0.701567	0.995663	0.996245	0.96501
.1.3.9.10.	0.734114	0.738805	0.709047	0.710764	0.701567	0.995872	0.988255	0.988764
.1.3.9.11.	0.734104	0.738794	0.709047	0.710764	0.701567	0.995872	0.996264	0.988809
.1.3.10.11.	0.734098	0.738788	0.709047	0.710764	0.701567	0.995872	0.988255	0.988538
.1.9.10.11.	0.734101	0.738792	0.709047	0.710764	0.701567	0.998616	0.996245	0.965486
.2.3.9.10.	0.734111	0.738801	0.709047	0.710764	0.701567	0.995867	0.996048	0.965646
.2.3.9.11.	0.734098	0.738787	0.709047	0.710764	0.701567	0.995867	0.99875	0.96583
.2.3.10.11.	0.734094	0.738783	0.709047	0.710764	0.701567	0.995867	0.996048	0.96501
.2.9.10.11.	0.734093	0.738783	0.709047	0.710764	0.701567	0.998597	0.998744	0.894395
.3.9.10.11.	0.7341	0.73879	0.709047	0.710764	0.701567	0.998654	0.996048	0.965486
.1.2.3.9.10.	0.734122	0.738813	0.709047	0.710764	0.701567	0.987186	0.988255	0.965646
.1.2.3.9.11.	0.734114	0.738804	0.709047	0.710764	0.701567	0.987186	0.996264	0.96583
.1.2.3.10.11.	0.73411	0.7388	0.709047	0.710764	0.701567	0.987186	0.988255	0.96501
.1.2.9.10.11.	0.734112	0.738803	0.709047	0.710764	0.701567	0.995663	0.996245	0.894395
.1.3.9.10.11.	0.734114	0.738805	0.709047	0.710764	0.701567	0.995872	0.988255	0.965486
.2.3.9.10.11.	0.734111	0.738801	0.709047	0.710764	0.701567	0.995867	0.996048	0.894395
.1.2.3.9.10.11.	0.734122	0.738813	0.709047	0.710764	0.701567	0.987186	0.988255	0.894395

Appendix D: Results – Probability Range for Stage 2 0.50 to 0.80 and for all Subsequent Stages 0.95 to 1.0

NETWORK ALLOCATION	23	24	6	7	8	TASK2	TASK3	TASK4
Assigned to task 1	0.909243	0.908133	0.904994	0.898995	0.911903	0.999999	0.999999	0.999992
.1.	0.909252	0.908143	0.904994	0.898995	0.911903	0.999992	0.999991	0.999992
.2.	0.909249	0.908139	0.904994	0.898995	0.911903	0.999993	0.999999	0.999943
.3.	0.90925	0.908141	0.904994	0.898995	0.911903	0.999992	0.999993	0.999992
.9.	0.909249	0.90814	0.904994	0.898995	0.911903	0.999999	0.999999	0.999935
.10.	0.90925	0.908141	0.904994	0.898995	0.911903	0.999999	0.999991	0.999944
.11.	0.909243	0.908134	0.904994	0.898995	0.911903	0.999999	0.999999	0.999949
.1.2.	0.909254	0.908144	0.904994	0.898995	0.911903	0.999945	0.999991	0.999943
.1.3.	0.909255	0.908145	0.904994	0.898995	0.911903	0.999941	0.999946	0.999992
.1.9.	0.909254	0.908145	0.904994	0.898995	0.911903	0.999992	0.999991	0.999935
.1.10.	0.909254	0.908145	0.904994	0.898995	0.911903	0.999992	0.999936	0.999944
.1.11.	0.909252	0.908143	0.904994	0.898995	0.911903	0.999992	0.999991	0.999949
.2.3.	0.909252	0.908143	0.904994	0.898995	0.911903	0.999947	0.999993	0.999943
.2.9.	0.909252	0.908143	0.904994	0.898995	0.911903	0.999993	0.999999	0.999555
.2.10.	0.909251	0.908142	0.904994	0.898995	0.911903	0.999993	0.999991	0.999622
.2.11.	0.909249	0.908139	0.904994	0.898995	0.911903	0.999993	0.999999	0.999651
.3.9.	0.909253	0.908143	0.904994	0.898995	0.911903	0.999992	0.999993	0.999935
.3.10.	0.909254	0.908144	0.904994	0.898995	0.911903	0.999992	0.999944	0.999944
.3.11.	0.90925	0.908141	0.904994	0.898995	0.911903	0.999992	0.999993	0.999949
.9.10.	0.909253	0.908144	0.904994	0.898995	0.911903	0.999999	0.999991	0.999565
.9.11.	0.90925	0.90814	0.904994	0.898995	0.911903	0.999999	0.999999	0.9996
.10.11.	0.90925	0.908141	0.904994	0.898995	0.911903	0.999999	0.999991	0.99965
.1.2.3.	0.909256	0.908147	0.904994	0.898995	0.911903	0.999591	0.999946	0.999943
.1.2.9.	0.909255	0.908146	0.904994	0.898995	0.911903	0.999945	0.999991	0.999555
.1.2.10.	0.909255	0.908145	0.904994	0.898995	0.911903	0.999945	0.999936	0.999622
.1.2.11.	0.909254	0.908144	0.904994	0.898995	0.911903	0.999945	0.999991	0.999651
.1.3.9.	0.909256	0.908146	0.904994	0.898995	0.911903	0.999941	0.999946	0.999935
.1.3.10.	0.909256	0.908147	0.904994	0.898995	0.911903	0.999941	0.999602	0.999944
.1.3.11.	0.909255	0.908145	0.904994	0.898995	0.911903	0.999941	0.999946	0.999949
.1.9.10.	0.909255	0.908146	0.904994	0.898995	0.911903	0.999992	0.999936	0.999565
.1.9.11.	0.909254	0.908145	0.904994	0.898995	0.911903	0.999992	0.999991	0.9996
.1.10.11.	0.909254	0.908145	0.904994	0.898995	0.911903	0.999992	0.999936	0.99965
.2.3.9.	0.909254	0.908145	0.904994	0.898995	0.911903	0.999947	0.999993	0.999555
.2.3.10.	0.909255	0.908145	0.904994	0.898995	0.911903	0.999947	0.999944	0.999622
.2.3.11.	0.909252	0.908143	0.904994	0.898995	0.911903	0.999947	0.999993	0.999651
.2.9.10.	0.909254	0.908144	0.904994	0.898995	0.911903	0.999993	0.999991	0.997051
.2.9.11.	0.909252	0.908143	0.904994	0.898995	0.911903	0.999993	0.999999	0.997247
.2.10.11.	0.909251	0.908142	0.904994	0.898995	0.911903	0.999993	0.999991	0.997606
.3.9.10.	0.909255	0.908145	0.904994	0.898995	0.911903	0.999992	0.999944	0.999565
.3.9.11.	0.909253	0.908143	0.904994	0.898995	0.911903	0.999992	0.999993	0.9996
.3.10.11.	0.909254	0.908144	0.904994	0.898995	0.911903	0.999992	0.999944	0.99965

.9.10.11.	0.909253	0.908144	0.904994	0.898995	0.911903	0.999999	0.999991	0.99728
.1.2.3.9.	0.909257	0.908147	0.904994	0.898995	0.911903	0.999591	0.999946	0.999555
.1.2.3.10.	0.909257	0.908147	0.904994	0.898995	0.911903	0.999591	0.999602	0.999622
.1.2.3.11.	0.909256	0.908147	0.904994	0.898995	0.911903	0.999591	0.999946	0.999651
.1.2.9.10.	0.909256	0.908146	0.904994	0.898995	0.911903	0.999945	0.999936	0.997051
.1.2.9.11.	0.909255	0.908146	0.904994	0.898995	0.911903	0.999945	0.999991	0.997247
.1.2.10.11.	0.909255	0.908145	0.904994	0.898995	0.911903	0.999945	0.999936	0.997606
.1.3.9.10.	0.909257	0.908147	0.904994	0.898995	0.911903	0.999941	0.999602	0.999565
.1.3.9.11.	0.909256	0.908146	0.904994	0.898995	0.911903	0.999941	0.999946	0.9996
.1.3.10.11.	0.909256	0.908147	0.904994	0.898995	0.911903	0.999941	0.999602	0.99965
.1.9.10.11.	0.909255	0.908146	0.904994	0.898995	0.911903	0.999992	0.999936	0.99728
.2.3.9.10.	0.909255	0.908146	0.904994	0.898995	0.911903	0.999947	0.999944	0.997051
.2.3.9.11.	0.909254	0.908145	0.904994	0.898995	0.911903	0.999947	0.999993	0.997247
.2.3.10.11.	0.909255	0.908145	0.904994	0.898995	0.911903	0.999947	0.999944	0.997606
.2.9.10.11.	0.909254	0.908144	0.904994	0.898995	0.911903	0.999993	0.999991	0.98129
.3.9.10.11.	0.909255	0.908145	0.904994	0.898995	0.911903	0.999992	0.999944	0.99728
.1.2.3.9.10.	0.909257	0.908148	0.904994	0.898995	0.911903	0.999591	0.999602	0.997051
.1.2.3.9.11.	0.909257	0.908147	0.904994	0.898995	0.911903	0.999591	0.999946	0.997247
.1.2.3.10.11.	0.909257	0.908147	0.904994	0.898995	0.911903	0.999591	0.999602	0.997606
.1.2.9.10.11.	0.909256	0.908146	0.904994	0.898995	0.911903	0.999945	0.999936	0.98129
.1.3.9.10.11.	0.909257	0.908147	0.904994	0.898995	0.911903	0.999941	0.999602	0.99728
.2.3.9.10.11.	0.909255	0.908146	0.904994	0.898995	0.911903	0.999947	0.999944	0.98129
.1.2.3.9.10.11.	0.909257	0.908148	0.904994	0.898995	0.911903	0.999591	0.999602	0.98129

Appendix E: One run – Probability Range 0.95 to 1.0

Stage 3

Assets assigned to task 1	17	18	19	20	21	22
No assets assigned to task 1.	0.998855	0.979461	0.990012	0.994213	0.989104	0.978195
.1.	0.998859	0.979464	0.990012	0.994213	0.989104	0.978195
.2.	0.999194	0.979948	0.990012	0.994213	0.989104	0.978195
.3.	0.998859	0.980069	0.990012	0.994213	0.989104	0.978195
.9.	0.998884	0.996355	0.990012	0.994213	0.989104	0.978195
.10.	0.998858	0.980187	0.990012	0.994213	0.989104	0.978195
.11.	0.998859	0.979768	0.990012	0.994213	0.989104	0.978195
.1.2.	0.999194	0.979948	0.990012	0.994213	0.989104	0.978195
.1.3.	0.998859	0.980069	0.990012	0.994213	0.989104	0.978195
.1.9.	0.998884	0.996355	0.990012	0.994213	0.989104	0.978195
.1.10.	0.998859	0.980187	0.990012	0.994213	0.989104	0.978195
.1.11.	0.998859	0.979768	0.990012	0.994213	0.989104	0.978195
.2.3.	0.999194	0.980089	0.990012	0.994213	0.989104	0.978195
.2.9.	0.999195	0.996363	0.990012	0.994213	0.989104	0.978195
.2.10.	0.999194	0.980193	0.990012	0.994213	0.989104	0.978195
.2.11.	0.999194	0.979958	0.990012	0.994213	0.989104	0.978195
.3.9.	0.998884	0.996365	0.990012	0.994213	0.989104	0.978195
.3.10.	0.998859	0.980195	0.990012	0.994213	0.989104	0.978195
.3.11.	0.998859	0.980082	0.990012	0.994213	0.989104	0.978195
.9.10.	0.998884	0.996366	0.990012	0.994213	0.989104	0.978195
.9.11.	0.998884	0.99636	0.990012	0.994213	0.989104	0.978195
.10.11.	0.998859	0.980191	0.990012	0.994213	0.989104	0.978195
.1.2.3.	0.999194	0.980089	0.990012	0.994213	0.989104	0.978195
.1.2.9.	0.999195	0.996363	0.990012	0.994213	0.989104	0.978195
.1.2.10.	0.999194	0.980193	0.990012	0.994213	0.989104	0.978195
.1.2.11.	0.999194	0.979958	0.990012	0.994213	0.989104	0.978195
.1.3.9.	0.998884	0.996365	0.990012	0.994213	0.989104	0.978195
.1.3.10.	0.998859	0.980195	0.990012	0.994213	0.989104	0.978195
.1.3.11.	0.998859	0.980082	0.990012	0.994213	0.989104	0.978195
.1.9.10.	0.998884	0.996366	0.990012	0.994213	0.989104	0.978195
.1.9.11.	0.998884	0.99636	0.990012	0.994213	0.989104	0.978195
.1.10.11.	0.998859	0.980191	0.990012	0.994213	0.989104	0.978195
.2.3.9.	0.999195	0.996365	0.990012	0.994213	0.989104	0.978195
.2.3.10.	0.999194	0.980195	0.990012	0.994213	0.989104	0.978195
.2.3.11.	0.999194	0.98009	0.990012	0.994213	0.989104	0.978195
.2.9.10.	0.999195	0.996367	0.990012	0.994213	0.989104	0.978195
.2.9.11.	0.999195	0.996363	0.990012	0.994213	0.989104	0.978195
.2.10.11.	0.999194	0.980193	0.990012	0.994213	0.989104	0.978195
.3.9.10.	0.998884	0.996367	0.990012	0.994213	0.989104	0.978195
.3.9.11.	0.998884	0.996365	0.990012	0.994213	0.989104	0.978195
.3.10.11.	0.998859	0.980195	0.990012	0.994213	0.989104	0.978195
.9.10.11.	0.998884	0.996367	0.990012	0.994213	0.989104	0.978195

.1.2.3.9.	0.999195	0.996365	0.990012	0.994213	0.989104	0.978195
.1.2.3.10.	0.999194	0.980195	0.990012	0.994213	0.989104	0.978195
.1.2.3.11.	0.999194	0.98009	0.990012	0.994213	0.989104	0.978195
.1.2.9.10.	0.999195	0.996367	0.990012	0.994213	0.989104	0.978195
.1.2.9.11.	0.999195	0.996363	0.990012	0.994213	0.989104	0.978195
.1.2.10.11.	0.999194	0.980193	0.990012	0.994213	0.989104	0.978195
.1.3.9.10.	0.998884	0.996367	0.990012	0.994213	0.989104	0.978195
.1.3.9.11.	0.998884	0.996365	0.990012	0.994213	0.989104	0.978195
.1.3.10.11.	0.998859	0.980195	0.990012	0.994213	0.989104	0.978195
.1.9.10.11.	0.998884	0.996367	0.990012	0.994213	0.989104	0.978195
.2.3.9.10.	0.999195	0.996367	0.990012	0.994213	0.989104	0.978195
.2.3.9.11.	0.999195	0.996365	0.990012	0.994213	0.989104	0.978195
.2.3.10.11.	0.999194	0.980195	0.990012	0.994213	0.989104	0.978195
.2.9.10.11.	0.999195	0.996367	0.990012	0.994213	0.989104	0.978195
.3.9.10.11.	0.998884	0.996367	0.990012	0.994213	0.989104	0.978195
.1.2.3.9.10.	0.999195	0.996367	0.990012	0.994213	0.989104	0.978195
.1.2.3.9.11.	0.999195	0.996365	0.990012	0.994213	0.989104	0.978195
.1.2.3.10.11.	0.999194	0.980195	0.990012	0.994213	0.989104	0.978195
.1.2.9.10.11.	0.999195	0.996367	0.990012	0.994213	0.989104	0.978195
.1.3.9.10.11.	0.998884	0.996367	0.990012	0.994213	0.989104	0.978195
.2.3.9.10.11.	0.999195	0.996367	0.990012	0.994213	0.989104	0.978195
.1.2.3.9.10.11.	0.999195	0.996367	0.990012	0.994213	0.989104	0.978195

Stage 4

Assets assigned to task 1	1	2	23	3	4	6	7	8	9	10
No assets	0	0	0.998086	0	0.958039	0.992495	0.944072	0.974538	0	0
.1.	0.967415	0	0.998086	0	0.958039	0.992495	0.944072	0.974538	0	0
.2.	0	0.968504	0.998094	0	0.958039	0.992495	0.944072	0.974538	0	0
.3.	0	0	0.998087	0.986476	0.958039	0.992495	0.944072	0.974538	0	0
.9.	0	0	0.998105	0	0.958039	0.992495	0.944072	0.974538	0.954635	0
.10.	0	0	0.998087	0	0.958039	0.992495	0.944072	0.974538	0	0.975189
.11.	0	0	0.998086	0	0.958039	0.992495	0.944072	0.974538	0	0
.1.2.	0.967427	0.968504	0.998094	0	0.958039	0.992495	0.944072	0.974538	0	0
.1.3.	0.967416	0	0.998087	0.986476	0.958039	0.992495	0.944072	0.974538	0	0
.1.9.	0.967434	0	0.998105	0	0.958039	0.992495	0.944072	0.974538	0.954635	0
.1.10.	0.967416	0	0.998087	0	0.958039	0.992495	0.944072	0.974538	0	0.975189
.1.11.	0.967416	0	0.998086	0	0.958039	0.992495	0.944072	0.974538	0	0
.2.3.	0	0.968504	0.998095	0.986483	0.958039	0.992495	0.944072	0.974538	0	0
.2.9.	0	0.968516	0.998108	0	0.958039	0.992495	0.944072	0.974538	0.954643	0
.2.10.	0	0.968504	0.998095	0	0.958039	0.992495	0.944072	0.974538	0	0.975195
.2.11.	0	0.968504	0.998095	0	0.958039	0.992495	0.944072	0.974538	0	0
.3.9.	0	0	0.998105	0.986897	0.958039	0.992495	0.944072	0.974538	0.954645	0
.3.10.	0	0	0.998087	0.986479	0.958039	0.992495	0.944072	0.974538	0	0.975197
.3.11.	0	0	0.998087	0.986476	0.958039	0.992495	0.944072	0.974538	0	0

.9.10.	0	0	0.998105	0	0.958039	0.992495	0.944072	0.974538	0.954647	0.991286
.9.11.	0	0	0.998105	0	0.958039	0.992495	0.944072	0.974538	0.95464	0
.10.11.	0	0	0.998087	0	0.958039	0.992495	0.944072	0.974538	0	0.975193
.1.2.3.	0.967428	0.968504	0.998095	0.986483	0.958039	0.992495	0.944072	0.974538	0	0
.1.2.9.	0.96744	0.968516	0.998108	0	0.958039	0.992495	0.944072	0.974538	0.954643	0
.1.2.10.	0.967428	0.968504	0.998095	0	0.958039	0.992495	0.944072	0.974538	0	0.975195
.1.2.11.	0.967427	0.968504	0.998095	0	0.958039	0.992495	0.944072	0.974538	0	0
.1.3.9.	0.967434	0	0.998105	0.986897	0.958039	0.992495	0.944072	0.974538	0.954645	0
.1.3.10.	0.967416	0	0.998087	0.986479	0.958039	0.992495	0.944072	0.974538	0	0.975197
.1.3.11.	0.967416	0	0.998087	0.986476	0.958039	0.992495	0.944072	0.974538	0	0
.1.9.10.	0.967434	0	0.998105	0	0.958039	0.992495	0.944072	0.974538	0.954647	0.991286
.1.9.11.	0.967434	0	0.998105	0	0.958039	0.992495	0.944072	0.974538	0.95464	0
.1.10.11.	0.967416	0	0.998087	0	0.958039	0.992495	0.944072	0.974538	0	0.975193
.2.3.9.	0	0.968516	0.998108	0.986898	0.958039	0.992495	0.944072	0.974538	0.954645	0
.2.3.10.	0	0.968504	0.998095	0.986485	0.958039	0.992495	0.944072	0.974538	0	0.975197
.2.3.11.	0	0.968504	0.998095	0.986483	0.958039	0.992495	0.944072	0.974538	0	0
.2.9.10.	0	0.968516	0.998108	0	0.958039	0.992495	0.944072	0.974538	0.954647	0.991286
.2.9.11.	0	0.968516	0.998108	0	0.958039	0.992495	0.944072	0.974538	0.954643	0
.2.10.11.	0	0.968504	0.998095	0	0.958039	0.992495	0.944072	0.974538	0	0.975195
.3.9.10.	0	0	0.998105	0.986897	0.958039	0.992495	0.944072	0.974538	0.954647	0.991286
.3.9.11.	0	0	0.998105	0.986897	0.958039	0.992495	0.944072	0.974538	0.954645	0
.3.10.11.	0	0	0.998087	0.986479	0.958039	0.992495	0.944072	0.974538	0	0.975197
.9.10.11.	0	0	0.998105	0	0.958039	0.992495	0.944072	0.974538	0.954647	0.991286
.1.2.3.9.	0.96744	0.968516	0.998108	0.986898	0.958039	0.992495	0.944072	0.974538	0.954645	0
.1.2.3.10.	0.967428	0.968504	0.998095	0.986485	0.958039	0.992495	0.944072	0.974538	0	0.975197
.1.2.3.11.	0.967428	0.968504	0.998095	0.986483	0.958039	0.992495	0.944072	0.974538	0	0
.1.2.9.10.	0.96744	0.968516	0.998108	0	0.958039	0.992495	0.944072	0.974538	0.954647	0.991286
.1.2.9.11.	0.96744	0.968516	0.998108	0	0.958039	0.992495	0.944072	0.974538	0.954643	0
.1.2.10.11.	0.967428	0.968504	0.998095	0	0.958039	0.992495	0.944072	0.974538	0	0.975195
.1.3.9.10.	0.967434	0	0.998105	0.986897	0.958039	0.992495	0.944072	0.974538	0.954647	0.991286
.1.3.9.11.	0.967434	0	0.998105	0.986897	0.958039	0.992495	0.944072	0.974538	0.954645	0
.1.3.10.11.	0.967416	0	0.998087	0.986479	0.958039	0.992495	0.944072	0.974538	0	0.975197
.1.9.10.11.	0.967434	0	0.998105	0	0.958039	0.992495	0.944072	0.974538	0.954647	0.991286
.2.3.9.10.	0	0.968516	0.998108	0.986898	0.958039	0.992495	0.944072	0.974538	0.954647	0.991286
.2.3.9.11.	0	0.968516	0.998108	0.986898	0.958039	0.992495	0.944072	0.974538	0.954645	0
.2.3.10.11.	0	0.968504	0.998095	0.986485	0.958039	0.992495	0.944072	0.974538	0	0.975197
.2.9.10.11.	0	0.968516	0.998108	0	0.958039	0.992495	0.944072	0.974538	0.954647	0.991286
.3.9.10.11.	0	0	0.998105	0.986897	0.958039	0.992495	0.944072	0.974538	0.954647	0.991286
.1.2.3.9.10.	0.96744	0.968516	0.998108	0.986898	0.958039	0.992495	0.944072	0.974538	0.954647	0.991286
.1.2.3.9.11.	0.96744	0.968516	0.998108	0.986898	0.958039	0.992495	0.944072	0.974538	0.954645	0
.1.2.3.10.11.	0.967428	0.968504	0.998095	0.986485	0.958039	0.992495	0.944072	0.974538	0	0.975197
.1.2.9.10.11.	0.96744	0.968516	0.998108	0	0.958039	0.992495	0.944072	0.974538	0.954647	0.991286
.1.3.9.10.11.	0.967434	0	0.998105	0.986897	0.958039	0.992495	0.944072	0.974538	0.954647	0.991286
.2.3.9.10.11.	0	0.968516	0.998108	0.986898	0.958039	0.992495	0.944072	0.974538	0.954647	0.991286
.1.2.3.9.10.11.	0.96744	0.968516	0.998108	0.986898	0.958039	0.992495	0.944072	0.974538	0.954647	0.991286

Stage 5

Assets assigned to task 1	1	2	23	3	24	6	7	8	9	10
No assets assigned to task 1.	0	0	0.975652	0	0.982293	0.985483	0.938972	0.965947	0	0
.1.	0.950705	0	0.975652	0	0.982304	0.985483	0.938972	0.965947	0	0
.2.	0	0.953351	0.97566	0	0.993233	0.985483	0.938972	0.965947	0	0
.3.	0	0	0.975653	0.947759	0.982297	0.985483	0.938972	0.965947	0	0
.9.	0	0	0.975671	0	0.982293	0.985483	0.938972	0.965947	0.918555	0
.10.	0	0	0.975653	0	0.982293	0.985483	0.938972	0.965947	0	0.926685
.11.	0	0	0.975652	0	0.982293	0.985483	0.938972	0.965947	0	0
.1.2.	0.950717	0.953351	0.97566	0	0.993234	0.985483	0.938972	0.965947	0	0
.1.3.	0.950706	0	0.975653	0.947759	0.982304	0.985483	0.938972	0.965947	0	0
.1.9.	0.950724	0	0.975671	0	0.982304	0.985483	0.938972	0.965947	0.918555	0
.1.10.	0.950706	0	0.975653	0	0.982304	0.985483	0.938972	0.965947	0	0.926685
.1.11.	0.950706	0	0.975652	0	0.982304	0.985483	0.938972	0.965947	0	0
.2.3.	0	0.953351	0.975661	0.947765	0.993234	0.985483	0.938972	0.965947	0	0
.2.9.	0	0.953364	0.975673	0	0.993234	0.985483	0.938972	0.965947	0.918562	0
.2.10.	0	0.953351	0.975661	0	0.993233	0.985483	0.938972	0.965947	0	0.926691
.2.11.	0	0.953351	0.97566	0	0.993233	0.985483	0.938972	0.965947	0	0
.3.9.	0	0	0.975671	0.948163	0.982297	0.985483	0.938972	0.965947	0.918564	0
.3.10.	0	0	0.975653	0.947762	0.982297	0.985483	0.938972	0.965947	0	0.926692
.3.11.	0	0	0.975653	0.947759	0.982297	0.985483	0.938972	0.965947	0	0
.9.10.	0	0	0.975671	0	0.982293	0.985483	0.938972	0.965947	0.918566	0.941981
.9.11.	0	0	0.975671	0	0.982293	0.985483	0.938972	0.965947	0.91856	0
.10.11.	0	0	0.975653	0	0.982293	0.985483	0.938972	0.965947	0	0.926689
.1.2.3.	0.950717	0.953351	0.975661	0.947765	0.993234	0.985483	0.938972	0.965947	0	0
.1.2.9.	0.95073	0.953364	0.975673	0	0.993234	0.985483	0.938972	0.965947	0.918562	0
.1.2.10.	0.950717	0.953351	0.975661	0	0.993234	0.985483	0.938972	0.965947	0	0.926691
.1.2.11.	0.950717	0.953351	0.97566	0	0.993234	0.985483	0.938972	0.965947	0	0
.1.3.9.	0.950724	0	0.975671	0.948163	0.982304	0.985483	0.938972	0.965947	0.918564	0
.1.3.10.	0.950706	0	0.975653	0.947762	0.982304	0.985483	0.938972	0.965947	0	0.926692
.1.3.11.	0.950706	0	0.975653	0.947759	0.982304	0.985483	0.938972	0.965947	0	0
.1.9.10.	0.950724	0	0.975671	0	0.982304	0.985483	0.938972	0.965947	0.918566	0.941981
.1.9.11.	0.950724	0	0.975671	0	0.982304	0.985483	0.938972	0.965947	0.91856	0
.1.10.11.	0.950706	0	0.975653	0	0.982304	0.985483	0.938972	0.965947	0	0.926689
.2.3.9.	0	0.953364	0.975673	0.948164	0.993234	0.985483	0.938972	0.965947	0.918564	0
.2.3.10.	0	0.953351	0.975661	0.947768	0.993234	0.985483	0.938972	0.965947	0	0.926692
.2.3.11.	0	0.953351	0.975661	0.947765	0.993234	0.985483	0.938972	0.965947	0	0
.2.9.10.	0	0.953364	0.975673	0	0.993234	0.985483	0.938972	0.965947	0.918566	0.941981
.2.9.11.	0	0.953364	0.975673	0	0.993234	0.985483	0.938972	0.965947	0.918563	0
.2.10.11.	0	0.953351	0.975661	0	0.993233	0.985483	0.938972	0.965947	0	0.926691
.3.9.10.	0	0	0.975671	0.948163	0.982297	0.985483	0.938972	0.965947	0.918566	0.941981
.3.9.11.	0	0	0.975671	0.948163	0.982297	0.985483	0.938972	0.965947	0.918564	0
.3.10.11.	0	0	0.975653	0.947762	0.982297	0.985483	0.938972	0.965947	0	0.926692
.9.10.11.	0	0	0.975671	0	0.982293	0.985483	0.938972	0.965947	0.918566	0.941981

.1.2.3.9.	0.95073	0.953364	0.975673	0.948164	0.993234	0.985483	0.938972	0.965947	0.918564	0
.1.2.3.10.	0.950717	0.953351	0.975661	0.947768	0.993234	0.985483	0.938972	0.965947	0	0.926692
.1.2.3.11.	0.950717	0.953351	0.975661	0.947765	0.993234	0.985483	0.938972	0.965947	0	0
.1.2.9.10.	0.95073	0.953364	0.975673	0	0.993234	0.985483	0.938972	0.965947	0.918566	0.941981
.1.2.9.11.	0.95073	0.953364	0.975673	0	0.993234	0.985483	0.938972	0.965947	0.918563	0
.1.2.10.11.	0.950717	0.953351	0.975661	0	0.993234	0.985483	0.938972	0.965947	0	0.926691
.1.3.9.10.	0.950724	0	0.975671	0.948163	0.982304	0.985483	0.938972	0.965947	0.918566	0.941981
.1.3.9.11.	0.950724	0	0.975671	0.948163	0.982304	0.985483	0.938972	0.965947	0.918564	0
.1.3.10.11.	0.950706	0	0.975653	0.947762	0.982304	0.985483	0.938972	0.965947	0	0.926692
.1.9.10.11.	0.950724	0	0.975671	0	0.982304	0.985483	0.938972	0.965947	0.918566	0.941981
.2.3.9.10.	0	0.953364	0.975673	0.948164	0.993234	0.985483	0.938972	0.965947	0.918566	0.941981
.2.3.9.11.	0	0.953364	0.975673	0.948164	0.993234	0.985483	0.938972	0.965947	0.918564	0
.2.3.10.11.	0	0.953351	0.975661	0.947768	0.993234	0.985483	0.938972	0.965947	0	0.926692
.2.9.10.11.	0	0.953364	0.975673	0	0.993234	0.985483	0.938972	0.965947	0.918566	0.941981
.3.9.10.11.	0	0	0.975671	0.948163	0.982297	0.985483	0.938972	0.965947	0.918566	0.941981
.1.2.3.9.10.	0.95073	0.953364	0.975673	0.948164	0.993234	0.985483	0.938972	0.965947	0.918566	0.941981
.1.2.3.9.11.	0.95073	0.953364	0.975673	0.948164	0.993234	0.985483	0.938972	0.965947	0.918564	0
.1.2.3.10.11.	0.950717	0.953351	0.975661	0.947768	0.993234	0.985483	0.938972	0.965947	0	0.926692
.1.2.9.10.11.	0.95073	0.953364	0.975673	0	0.993234	0.985483	0.938972	0.965947	0.918566	0.941981
.1.3.9.10.11.	0.950724	0	0.975671	0.948163	0.982304	0.985483	0.938972	0.965947	0.918566	0.941981
.2.3.9.10.11.	0	0.953364	0.975673	0.948164	0.993234	0.985483	0.938972	0.965947	0.918566	0.941981
.1.2.3.9.10.11.	0.95073	0.953364	0.975673	0.948164	0.993234	0.985483	0.938972	0.965947	0.918566	0.941981

Stage 6

Assets assigned to task 1	25	26	27	28	29	30
No assets assigned to task 1.	0.978952	0.988466	0.972698	0.899968	0.963896	0.920226
.1.	0.979353	0.997067	0.972698	0.899968	0.963896	0.920226
.2.	0.984378	0.98936	0.972698	0.899968	0.963896	0.920226
.3.	0.979342	0.989227	0.972698	0.899968	0.963896	0.920226
.9.	0.979333	0.989281	0.972698	0.899968	0.963896	0.920226
.10.	0.979334	0.989221	0.972698	0.899968	0.963896	0.920226
.11.	0.978952	0.988466	0.972698	0.899968	0.963896	0.920226
.1.2.	0.984385	0.997111	0.972698	0.899968	0.963896	0.920226
.1.3.	0.979372	0.997105	0.972698	0.899968	0.963896	0.920226
.1.9.	0.979372	0.997107	0.972698	0.899968	0.963896	0.920226
.1.10.	0.979372	0.997104	0.972698	0.899968	0.963896	0.920226
.1.11.	0.979353	0.997067	0.972698	0.899968	0.963896	0.920226
.2.3.	0.984385	0.989374	0.972698	0.899968	0.963896	0.920226
.2.9.	0.984385	0.989375	0.972698	0.899968	0.963896	0.920226
.2.10.	0.984385	0.989374	0.972698	0.899968	0.963896	0.920226
.2.11.	0.984378	0.98936	0.972698	0.899968	0.963896	0.920226
.3.9.	0.979365	0.989343	0.972698	0.899968	0.963896	0.920226
.3.10.	0.979362	0.989277	0.972698	0.899968	0.963896	0.920226
.3.11.	0.979342	0.989227	0.972698	0.899968	0.963896	0.920226

9.10.	0.979365	0.989344	0.972698	0.899968	0.963896	0.920226
9.11.	0.979333	0.989281	0.972698	0.899968	0.963896	0.920226
10.11.	0.979334	0.989221	0.972698	0.899968	0.963896	0.920226
1.2.3.	0.984385	0.997112	0.972698	0.899968	0.963896	0.920226
1.2.9.	0.984385	0.997112	0.972698	0.899968	0.963896	0.920226
1.2.10.	0.984385	0.997112	0.972698	0.899968	0.963896	0.920226
1.2.11.	0.984385	0.997111	0.972698	0.899968	0.963896	0.920226
1.3.9.	0.979373	0.997111	0.972698	0.899968	0.963896	0.920226
1.3.10.	0.979373	0.997107	0.972698	0.899968	0.963896	0.920226
1.3.11.	0.979372	0.997105	0.972698	0.899968	0.963896	0.920226
1.9.10.	0.979373	0.997111	0.972698	0.899968	0.963896	0.920226
1.9.11.	0.979372	0.997107	0.972698	0.899968	0.963896	0.920226
1.10.11.	0.979372	0.997104	0.972698	0.899968	0.963896	0.920226
2.3.9.	0.984385	0.989376	0.972698	0.899968	0.963896	0.920226
2.3.10.	0.984385	0.989375	0.972698	0.899968	0.963896	0.920226
2.3.11.	0.984385	0.989374	0.972698	0.899968	0.963896	0.920226
2.9.10.	0.984385	0.989376	0.972698	0.899968	0.963896	0.920226
2.9.11.	0.984385	0.989375	0.972698	0.899968	0.963896	0.920226
2.10.11.	0.984385	0.989374	0.972698	0.899968	0.963896	0.920226
3.9.10.	0.979367	0.989347	0.972698	0.899968	0.963896	0.920226
3.9.11.	0.979365	0.989343	0.972698	0.899968	0.963896	0.920226
3.10.11.	0.979362	0.989277	0.972698	0.899968	0.963896	0.920226
9.10.11.	0.979365	0.989344	0.972698	0.899968	0.963896	0.920226
1.2.3.9.	0.984385	0.997112	0.972698	0.899968	0.963896	0.920226
1.2.3.10.	0.984385	0.997112	0.972698	0.899968	0.963896	0.920226
1.2.3.11.	0.984385	0.997112	0.972698	0.899968	0.963896	0.920226
1.2.9.10.	0.984385	0.997112	0.972698	0.899968	0.963896	0.920226
1.2.9.11.	0.984385	0.997112	0.972698	0.899968	0.963896	0.920226
1.2.10.11.	0.984385	0.997112	0.972698	0.899968	0.963896	0.920226
1.3.9.10.	0.979373	0.997111	0.972698	0.899968	0.963896	0.920226
1.3.9.11.	0.979373	0.997111	0.972698	0.899968	0.963896	0.920226
1.3.10.11.	0.979373	0.997107	0.972698	0.899968	0.963896	0.920226
1.9.10.11.	0.979373	0.997111	0.972698	0.899968	0.963896	0.920226
2.3.9.10.	0.984385	0.989376	0.972698	0.899968	0.963896	0.920226
2.3.9.11.	0.984385	0.989376	0.972698	0.899968	0.963896	0.920226
2.3.10.11.	0.984385	0.989375	0.972698	0.899968	0.963896	0.920226
2.9.10.11.	0.984385	0.989376	0.972698	0.899968	0.963896	0.920226
3.9.10.11.	0.979367	0.989347	0.972698	0.899968	0.963896	0.920226
1.2.3.9.10.	0.984385	0.997112	0.972698	0.899968	0.963896	0.920226
1.2.3.9.11.	0.984385	0.997112	0.972698	0.899968	0.963896	0.920226
1.2.3.10.11.	0.984385	0.997112	0.972698	0.899968	0.963896	0.920226
1.2.9.10.11.	0.984385	0.997112	0.972698	0.899968	0.963896	0.920226
1.3.9.10.11.	0.979373	0.997111	0.972698	0.899968	0.963896	0.920226
2.3.9.10.11.	0.984385	0.989376	0.972698	0.899968	0.963896	0.920226

1.2.3.9.10.11.	0.984385	0.997112	0.972698	0.899968	0.963896	0.920226
----------------	----------	----------	----------	----------	----------	----------

Stage 7

Assets assigned to task 1	31	32	33	24	6	7	8	34
No assets assigned to task 1.	0.997648	0.994228	0.970141	0.999465	0.955588	0.883327	0.942187	0.881466
.1.	0.998033	0.994415	0.970319	0.999779	0.955588	0.883327	0.942187	0.881466
.2.	0.997744	0.99443	0.970253	0.999554	0.955588	0.883327	0.942187	0.881466
.3.	0.997686	0.994257	0.970163	0.999496	0.955588	0.883327	0.942187	0.881466
.9.	0.997689	0.994258	0.970164	0.999498	0.955588	0.883327	0.942187	0.881466
.10.	0.997686	0.994257	0.970163	0.999496	0.955588	0.883327	0.942187	0.881466
.11.	0.997648	0.994228	0.970141	0.999465	0.955588	0.883327	0.942187	0.881466
.1.2.	0.998049	0.994548	0.970369	0.999794	0.955588	0.883327	0.942187	0.881466
.1.3.	0.998035	0.994416	0.97032	0.99978	0.955588	0.883327	0.942187	0.881466
.1.9.	0.998035	0.994416	0.97032	0.99978	0.955588	0.883327	0.942187	0.881466
.1.10.	0.998035	0.994416	0.97032	0.99978	0.955588	0.883327	0.942187	0.881466
.1.11.	0.998033	0.994415	0.970319	0.999779	0.955588	0.883327	0.942187	0.881466
.2.3.	0.997745	0.994431	0.970253	0.999554	0.955588	0.883327	0.942187	0.881466
.2.9.	0.997745	0.994431	0.970253	0.999554	0.955588	0.883327	0.942187	0.881466
.2.10.	0.997745	0.994431	0.970253	0.999554	0.955588	0.883327	0.942187	0.881466
.2.11.	0.997744	0.99443	0.970253	0.999554	0.955588	0.883327	0.942187	0.881466
.3.9.	0.997692	0.99426	0.970166	0.999501	0.955588	0.883327	0.942187	0.881466
.3.10.	0.997689	0.994259	0.970165	0.999498	0.955588	0.883327	0.942187	0.881466
.3.11.	0.997686	0.994257	0.970163	0.999496	0.955588	0.883327	0.942187	0.881466
.9.10.	0.997692	0.99426	0.970166	0.999501	0.955588	0.883327	0.942187	0.881466
.9.11.	0.997689	0.994258	0.970164	0.999498	0.955588	0.883327	0.942187	0.881466
.10.11.	0.997686	0.994257	0.970163	0.999496	0.955588	0.883327	0.942187	0.881466
.1.2.3.	0.998049	0.994548	0.970369	0.999795	0.955588	0.883327	0.942187	0.881466
.1.2.9.	0.998049	0.994548	0.970369	0.999795	0.955588	0.883327	0.942187	0.881466
.1.2.10.	0.998049	0.994548	0.970369	0.999795	0.955588	0.883327	0.942187	0.881466
.1.2.11.	0.998049	0.994548	0.970369	0.999794	0.955588	0.883327	0.942187	0.881466
.1.3.9.	0.998035	0.994416	0.970321	0.99978	0.955588	0.883327	0.942187	0.881466
.1.3.10.	0.998035	0.994416	0.97032	0.99978	0.955588	0.883327	0.942187	0.881466
.1.3.11.	0.998035	0.994416	0.97032	0.99978	0.955588	0.883327	0.942187	0.881466
.1.9.10.	0.998035	0.994416	0.970321	0.99978	0.955588	0.883327	0.942187	0.881466
.1.9.11.	0.998035	0.994416	0.97032	0.99978	0.955588	0.883327	0.942187	0.881466
.1.10.11.	0.998035	0.994416	0.97032	0.99978	0.955588	0.883327	0.942187	0.881466
.2.3.9.	0.997745	0.994431	0.970253	0.999555	0.955588	0.883327	0.942187	0.881466
.2.3.10.	0.997745	0.994431	0.970253	0.999554	0.955588	0.883327	0.942187	0.881466
.2.3.11.	0.997745	0.994431	0.970253	0.999554	0.955588	0.883327	0.942187	0.881466
.2.9.10.	0.997745	0.994431	0.970253	0.999555	0.955588	0.883327	0.942187	0.881466
.2.9.11.	0.997745	0.994431	0.970253	0.999554	0.955588	0.883327	0.942187	0.881466

.2.10.11.	0.997745	0.994431	0.970253	0.999554	0.955588	0.883327	0.942187	0.881466
.3.9.10.	0.997692	0.99426	0.970166	0.999501	0.955588	0.883327	0.942187	0.881466
.3.9.11.	0.997692	0.99426	0.970166	0.999501	0.955588	0.883327	0.942187	0.881466
.3.10.11.	0.997689	0.994259	0.970165	0.999498	0.955588	0.883327	0.942187	0.881466
.9.10.11.	0.997692	0.99426	0.970166	0.999501	0.955588	0.883327	0.942187	0.881466
.1.2.3.9.	0.998049	0.994548	0.970369	0.999795	0.955588	0.883327	0.942187	0.881466
.1.2.3.10.	0.998049	0.994548	0.970369	0.999795	0.955588	0.883327	0.942187	0.881466
.1.2.3.11.	0.998049	0.994548	0.970369	0.999795	0.955588	0.883327	0.942187	0.881466
.1.2.9.10.	0.998049	0.994548	0.970369	0.999795	0.955588	0.883327	0.942187	0.881466
.1.2.9.11.	0.998049	0.994548	0.970369	0.999795	0.955588	0.883327	0.942187	0.881466
.1.2.10.11.	0.998049	0.994548	0.970369	0.999795	0.955588	0.883327	0.942187	0.881466
.1.3.9.10.	0.998035	0.994416	0.970321	0.99978	0.955588	0.883327	0.942187	0.881466
.1.3.9.11.	0.998035	0.994416	0.970321	0.99978	0.955588	0.883327	0.942187	0.881466
.1.3.10.11.	0.998035	0.994416	0.97032	0.99978	0.955588	0.883327	0.942187	0.881466
.1.9.10.11.	0.998035	0.994416	0.970321	0.99978	0.955588	0.883327	0.942187	0.881466
.2.3.9.10.	0.997745	0.994431	0.970253	0.999555	0.955588	0.883327	0.942187	0.881466
.2.3.9.11.	0.997745	0.994431	0.970253	0.999555	0.955588	0.883327	0.942187	0.881466
.2.3.10.11.	0.997745	0.994431	0.970253	0.999554	0.955588	0.883327	0.942187	0.881466
.2.9.10.11.	0.997745	0.994431	0.970253	0.999555	0.955588	0.883327	0.942187	0.881466
.3.9.10.11.	0.997692	0.99426	0.970166	0.999501	0.955588	0.883327	0.942187	0.881466
.1.2.3.9.10.	0.998049	0.994548	0.970369	0.999795	0.955588	0.883327	0.942187	0.881466
.1.2.3.9.11.	0.998049	0.994548	0.970369	0.999795	0.955588	0.883327	0.942187	0.881466
.1.2.3.10.11.	0.998049	0.994548	0.970369	0.999795	0.955588	0.883327	0.942187	0.881466
.1.2.9.10.11.	0.998049	0.994548	0.970369	0.999795	0.955588	0.883327	0.942187	0.881466
.1.3.9.10.11.	0.998035	0.994416	0.970321	0.99978	0.955588	0.883327	0.942187	0.881466
.2.3.9.10.11.	0.997745	0.994431	0.970253	0.999555	0.955588	0.883327	0.942187	0.881466
.1.2.3.9.10.11.	0.998049	0.994548	0.970369	0.999795	0.955588	0.883327	0.942187	0.881466

Stage 8

Assets assigned to task 1	39	40	38	29	30
No assets assigned to task 1.	0.963979	0.932874	0.87665	0.902057	0.840365
.1.	0.963981	0.932874	0.87665	0.902057	0.840365
.2.	0.96398	0.932874	0.87665	0.902057	0.840365
.3.	0.963979	0.932874	0.87665	0.902057	0.840365
.9.	0.963979	0.932874	0.87665	0.902057	0.840365
.10.	0.963979	0.932874	0.87665	0.902057	0.840365
.11.	0.963979	0.932874	0.87665	0.902057	0.840365
.1.2.	0.963981	0.932874	0.87665	0.902057	0.840365
.1.3.	0.963981	0.932874	0.87665	0.902057	0.840365
.1.9.	0.963981	0.932874	0.87665	0.902057	0.840365
.1.10.	0.963981	0.932874	0.87665	0.902057	0.840365
.1.11.	0.963981	0.932874	0.87665	0.902057	0.840365

.2.3.	0.96398	0.932874	0.87665	0.902057	0.840365
.2.9.	0.96398	0.932874	0.87665	0.902057	0.840365
.2.10.	0.96398	0.932874	0.87665	0.902057	0.840365
.2.11.	0.96398	0.932874	0.87665	0.902057	0.840365
.3.9.	0.963979	0.932874	0.87665	0.902057	0.840365
.3.10.	0.963979	0.932874	0.87665	0.902057	0.840365
.3.11.	0.963979	0.932874	0.87665	0.902057	0.840365
.9.10.	0.963979	0.932874	0.87665	0.902057	0.840365
.9.11.	0.963979	0.932874	0.87665	0.902057	0.840365
.10.11.	0.963979	0.932874	0.87665	0.902057	0.840365
.1.2.3.	0.963981	0.932874	0.87665	0.902057	0.840365
.1.2.9.	0.963981	0.932874	0.87665	0.902057	0.840365
.1.2.10.	0.963981	0.932874	0.87665	0.902057	0.840365
.1.2.11.	0.963981	0.932874	0.87665	0.902057	0.840365
.1.3.9.	0.963981	0.932874	0.87665	0.902057	0.840365
.1.3.10.	0.963981	0.932874	0.87665	0.902057	0.840365
.1.3.11.	0.963981	0.932874	0.87665	0.902057	0.840365
.1.9.10.	0.963981	0.932874	0.87665	0.902057	0.840365
.1.9.11.	0.963981	0.932874	0.87665	0.902057	0.840365
.1.10.11.	0.963981	0.932874	0.87665	0.902057	0.840365
.2.3.9.	0.96398	0.932874	0.87665	0.902057	0.840365
.2.3.10.	0.96398	0.932874	0.87665	0.902057	0.840365
.2.3.11.	0.96398	0.932874	0.87665	0.902057	0.840365
.2.9.10.	0.96398	0.932874	0.87665	0.902057	0.840365
.2.9.11.	0.96398	0.932874	0.87665	0.902057	0.840365
.2.10.11.	0.96398	0.932874	0.87665	0.902057	0.840365
.3.9.10.	0.963979	0.932874	0.87665	0.902057	0.840365
.3.9.11.	0.963979	0.932874	0.87665	0.902057	0.840365
.3.10.11.	0.963979	0.932874	0.87665	0.902057	0.840365
.9.10.11.	0.963979	0.932874	0.87665	0.902057	0.840365
.1.2.3.9.	0.963981	0.932874	0.87665	0.902057	0.840365
.1.2.3.10.	0.963981	0.932874	0.87665	0.902057	0.840365
.1.2.3.11.	0.963981	0.932874	0.87665	0.902057	0.840365
.1.2.9.10.	0.963981	0.932874	0.87665	0.902057	0.840365
.1.2.9.11.	0.963981	0.932874	0.87665	0.902057	0.840365
.1.2.10.11.	0.963981	0.932874	0.87665	0.902057	0.840365
.1.3.9.10.	0.963981	0.932874	0.87665	0.902057	0.840365
.1.3.9.11.	0.963981	0.932874	0.87665	0.902057	0.840365
.1.3.10.11.	0.963981	0.932874	0.87665	0.902057	0.840365
.1.9.10.11.	0.963981	0.932874	0.87665	0.902057	0.840365
.2.3.9.10.	0.96398	0.932874	0.87665	0.902057	0.840365
.2.3.9.11.	0.96398	0.932874	0.87665	0.902057	0.840365
.2.3.10.11.	0.96398	0.932874	0.87665	0.902057	0.840365
.2.9.10.11.	0.96398	0.932874	0.87665	0.902057	0.840365
.3.9.10.11.	0.963979	0.932874	0.87665	0.902057	0.840365
.1.2.3.9.10.	0.963981	0.932874	0.87665	0.902057	0.840365

	0.963981	0.932874	0.87665	0.902057	0.840365
.1.2.3.9.11.					
.1.2.3.10.11.	0.963981	0.932874	0.87665	0.902057	0.840365
.1.2.9.10.11.	0.963981	0.932874	0.87665	0.902057	0.840365
.1.3.9.10.11.	0.963981	0.932874	0.87665	0.902057	0.840365
.2.3.9.10.11.	0.96398	0.932874	0.87665	0.902057	0.840365
.1.2.3.9.10.11.	0.963981	0.932874	0.87665	0.902057	0.840365

Stage 9

Assets assigned to task 1	35	36
No assets assigned to task 1.	0.996764	0.991725
.1.	0.996764	0.991725
.2.	0.996764	0.991725
.3.	0.996764	0.991725
.9.	0.996764	0.991725
.10.	0.996764	0.991725
.11.	0.996764	0.991725
.1.2.	0.996764	0.991725
.1.3.	0.996764	0.991725
.1.9.	0.996764	0.991725
.1.10.	0.996764	0.991725
.1.11.	0.996764	0.991725
.2.3.	0.996764	0.991725
.2.9.	0.996764	0.991725
.2.10.	0.996764	0.991725
.2.11.	0.996764	0.991725
.3.9.	0.996764	0.991725
.3.10.	0.996764	0.991725
.3.11.	0.996764	0.991725
.9.10.	0.996764	0.991725
.9.11.	0.996764	0.991725
.10.11.	0.996764	0.991725
.1.2.3.	0.996764	0.991725
.1.2.9.	0.996764	0.991725
.1.2.10.	0.996764	0.991725
.1.2.11.	0.996764	0.991725
.1.3.9.	0.996764	0.991725
.1.3.10.	0.996764	0.991725
.1.3.11.	0.996764	0.991725
.1.9.10.	0.996764	0.991725
.1.9.11.	0.996764	0.991725
.1.10.11.	0.996764	0.991725
.2.3.9.	0.996764	0.991725

.2.3.10.	0.996764	0.991725
.2.3.11.	0.996764	0.991725
.2.9.10.	0.996764	0.991725
.2.9.11.	0.996764	0.991725
.2.10.11.	0.996764	0.991725
.3.9.10.	0.996764	0.991725
.3.9.11.	0.996764	0.991725
.3.10.11.	0.996764	0.991725
.9.10.11.	0.996764	0.991725
.1.2.3.9.	0.996764	0.991725
.1.2.3.10.	0.996764	0.991725
.1.2.3.11.	0.996764	0.991725
.1.2.9.10.	0.996764	0.991725
.1.2.9.11.	0.996764	0.991725
.1.2.10.11.	0.996764	0.991725
.1.3.9.10.	0.996764	0.991725
.1.3.9.11.	0.996764	0.991725
.1.3.10.11.	0.996764	0.991725
.1.9.10.11.	0.996764	0.991725
.2.3.9.10.	0.996764	0.991725
.2.3.9.11.	0.996764	0.991725
.2.3.10.11.	0.996764	0.991725
.2.9.10.11.	0.996764	0.991725
.3.9.10.11.	0.996764	0.991725
.1.2.3.9.10.	0.996764	0.991725
.1.2.3.9.11.	0.996764	0.991725
.1.2.3.10.11.	0.996764	0.991725
.1.2.9.10.11.	0.996764	0.991725
.1.3.9.10.11.	0.996764	0.991725
.2.3.9.10.11.	0.996764	0.991725
.1.2.3.9.10.11.	0.996764	0.991725

Task 1 Endstage (Stage 10) and Task 2, 3, and 4 End of Task Probability of Success

Network Allocation	23	24	6	7	8	TASK2	TASK3	TASK4
No assets on task 1.	0.994361	0.952062	0.972344	0.984569	0.969514	1	1	1
.1.	0.994361	0.952062	0.972344	0.984569	0.969514	1	1	1
.2.	0.994361	0.952062	0.972344	0.984569	0.969514	1	1	1
.3.	0.994361	0.952062	0.972344	0.984569	0.969514	1	1	1
.9.	0.994361	0.952062	0.972344	0.984569	0.969514	1	1	1
.10.	0.994361	0.952062	0.972344	0.984569	0.969514	1	1	1
.11.	0.994361	0.952062	0.972344	0.984569	0.969514	1	1	0.999998
.1.2.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999999	1	1
.1.3.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999999	0.999999	1

.1.9.	0.994361	0.952062	0.972344	0.984569	0.969514	1	1	1
.1.10.	0.994361	0.952062	0.972344	0.984569	0.969514	1	1	1
.1.11.	0.994361	0.952062	0.972344	0.984569	0.969514	1	1	0.999998
.2.3.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999999	1	1
.2.9.	0.994361	0.952062	0.972344	0.984569	0.969514	1	1	0.999995
.2.10.	0.994361	0.952062	0.972344	0.984569	0.969514	1	1	0.999993
.2.11.	0.994361	0.952062	0.972344	0.984569	0.969514	1	1	0.999964
.3.9.	0.994361	0.952062	0.972344	0.984569	0.969514	1	1	1
.3.10.	0.994361	0.952062	0.972344	0.984569	0.969514	1	0.999999	1
.3.11.	0.994361	0.952062	0.972344	0.984569	0.969514	1	1	0.999998
.9.10.	0.994361	0.952062	0.972344	0.984569	0.969514	1	1	0.999995
.9.11.	0.994361	0.952062	0.972344	0.984569	0.969514	1	1	0.999973
.10.11.	0.994361	0.952062	0.972344	0.984569	0.969514	1	1	0.99996
.1.2.3.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999985	0.999999	1
.1.2.9.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999999	1	0.999995
.1.2.10.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999999	1	0.999993
.1.2.11.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999999	1	0.999964
.1.3.9.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999999	0.999999	1
.1.3.10.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999999	0.99999	1
.1.3.11.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999999	0.999999	0.999998
.1.9.10.	0.994361	0.952062	0.972344	0.984569	0.969514	1	1	0.999995
.1.9.11.	0.994361	0.952062	0.972344	0.984569	0.969514	1	1	0.999973
.1.10.11.	0.994361	0.952062	0.972344	0.984569	0.969514	1	1	0.99996
.2.3.9.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999999	1	0.999995
.2.3.10.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999999	0.999999	0.999993
.2.3.11.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999999	1	0.999964
.2.9.10.	0.994361	0.952062	0.972344	0.984569	0.969514	1	1	0.999913
.2.9.11.	0.994361	0.952062	0.972344	0.984569	0.969514	1	1	0.999565
.2.10.11.	0.994361	0.952062	0.972344	0.984569	0.969514	1	1	0.999352
.3.9.10.	0.994361	0.952062	0.972344	0.984569	0.969514	1	0.999999	0.999995
.3.9.11.	0.994361	0.952062	0.972344	0.984569	0.969514	1	1	0.999973
.3.10.11.	0.994361	0.952062	0.972344	0.984569	0.969514	1	0.999999	0.99996
.9.10.11.	0.994361	0.952062	0.972344	0.984569	0.969514	1	1	0.999518
.1.2.3.9.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999985	0.999999	0.999995
.1.2.3.10.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999985	0.99999	0.999993
.1.2.3.11.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999985	0.999999	0.999964
.1.2.9.10.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999999	1	0.999913
.1.2.9.11.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999999	1	0.999565
.1.2.10.11.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999999	1	0.999352
.1.3.9.10.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999999	0.99999	0.999995
.1.3.9.11.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999999	0.999999	0.999973
.1.3.10.11.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999999	0.99999	0.99996
.1.9.10.11.	0.994361	0.952062	0.972344	0.984569	0.969514	1	1	0.999518
.2.3.9.10.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999999	0.999999	0.999913
.2.3.9.11.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999999	1	0.999565
.2.3.10.11.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999999	0.999999	0.999352

.2.9.10.11.	0.994361	0.952062	0.972344	0.984569	0.969514	1	1	0.992116
.3.9.10.11.	0.994361	0.952062	0.972344	0.984569	0.969514	1	0.999999	0.999518
.1.2.3.9.10.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999985	0.99999	0.999913
.1.2.3.9.11.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999985	0.999999	0.999565
.1.2.3.10.11.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999985	0.99999	0.999352
.1.2.9.10.11.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999999	1	0.992116
.1.3.9.10.11.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999999	0.99999	0.999518
.2.3.9.10.11.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999999	0.999999	0.992116
.1.2.3.9.10.11.	0.994361	0.952062	0.972344	0.984569	0.969514	0.999985	0.99999	0.992116