7-8-2010

Modeling and Analysis of Cooperative Search Systems

Carlos A. Portilla
University of South Florida

Modeling and Analysis of Cooperative Search Systems

by

Carlos A. Portilla

A thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science in Industrial Engineering
Department of Industrial and Management Systems Engineering
College of Engineering
University of South Florida

Major Professor: Ali Yalcin, Ph.D.
David Jeffcoat, Ph.D.
Bo Zeng, Ph.D.

Date of Approval
July 8, 2010

Keywords: Petri Net, cueing, simulation, target detection, UAVs

Dedication

This thesis is dedicated to my loving parents Jorge and Dorys for instilling the importance of hard work and higher education; they are an example to follow. Last but not least, it is dedicated to my brother Jorge for always supporting me in every possible way.

Acknowledgements

This thesis would not be possible without the invaluable help of many people. First and foremost, I would like to thank my thesis supervisor Dr. Ali Yalcin for his guidance and support during all these years.

I am very grateful for the help and advice from my thesis committee Dr. Jeffcoat and Dr. Zeng. Finally, I want to thank my family that has been the major motivation in my life and my friends for all their support.

Table of Contents

List of Tables

List of Figures

Modeling and Analysis of Cooperative Search Systems

Carlos A Portilla

Abstract

The analysis of performance gains arising from cueing in cooperative search systems with autonomous vehicles has been studied using Continuous Time Markov Chains; where the search time distributions are assumed to follow the exponential distributions. This work proposes the use of Petri Nets to model and analyze these systems. The Petri Net model considers two types of autonomous vehicles: a search-only vehicle and n search-engage vehicles. Specific performance metrics are defined to measure system's performance. Through simulation, it is shown that the search time with stationary targets and cues that provide exact target location follows a triangular distribution. A methodology for approximating general distributions and incorporating them into the Petri Net model for performance analysis is presented.

Chapter 1

Introduction

1.1 Search Theory

Search theory is one of the oldest areas of operations research and encompasses all the models and algorithms that refer to the problem of finding a hidden target (L. D. Stone 1992). It uses the principles and methods of operations research to resolve search problems. The search scenarios that have been studied in search theory are: single searcher, cooperative search and coordinated search. In single searcher, there is basically one entity performing the search or the exploration. Cooperative and Coordinated search involve more than one entity working toward a goal in which there is a common interest or reward (Cao, Fukunaga and Kahng 2004). Coordinated search implies that there is collaboration between the entities. The scenario studied in this research involves cooperative search. The entities work together to cover the search area faster but there is no coordination among them to search and engage targets.

1.2 UAV Applications and Cooperative Search Systems

The entities that will be considered throughout this research are unmanned autonomous vehicles (UAVs). UAVs are robots which can perform tasks without continuous human guidance. UAVs are becoming increasingly prevalent; their use has increased exponentially over the last decade (Oracle Corporation 2007). UAVs have applications on land, in sea, and in the air. They are used for a broad range of applications, including police observation of civil disturbances, for work and measurement in radioactive environments, and reconnaissance support in natural disasters.

Some missions are better performed by a team of UAVs instead of only one single agent. For instance a dangerous and/or extensive mission, where it is unlikely that a single UAV survives to complete the task, a team is more suitable to perform it. In addition, research has shown that searching a particular area can be completed more quickly using multiple UAVs (Cole, et al. 2009). In all cases, cooperation among the UAVs is required for efficient and/or successful completion of the mission.

1.3 Cueing

The type of cooperation that will be studied in this research is cueing. In cooperative search applications, cueing is defined as any information that provides focus to a search; such as limiting the search area or providing a search heading (D. Jeffcoat 2004). Research has shown that cueing can significantly improve the probability of locating targets in cooperative search applications over a fixed period of time (Jeffcoat, Krokhmal and Zhupanska 2007). In addition, experience in Kosovo showed that cueing enhances battle space awareness by making UAVs much more efficient and survivable. The information transmitted in the cue let a UAV know where to look and thus decrease wasted surveillance time. In addition, it reduces the exposure to point air defenses of the UAVs, making them more survivable (Bingham 2001).

1.4 Motivation

There are many situations in which search processes might be facilitated via cueing. For example: a person, choosing from the menu in a restaurant, can receive additional information from the waiter to find the desired food. Or a student, looking up an article, can be helped by a librarian who points out the correct database. Or a team of UAVs, looking for survivors after a hurricane, might receive data about the possible location of the targets from another vehicle with superior capabilities. All these search processes have something in common. First there is an entity looking for something, and then it either finds it or receives additional information that will expedite the search process (cueing). The motivation of this work comes from the need to easily characterize these search processes and to measure system's performance.

The existing research in quantifying the performance gains arising from cueing utilizes continuous time Markov chains (CTMC) to model and analyze the system under study (Alexander and Jeffcoat 2007). CTMCs are a state orientated modeling formalism which requires the modeler to determine the state space of the complete system and assign transition probabilities between each of these states as a part of the modeling effort. This is viable for small cooperative search systems (Jeffcoat, Krokhmal and Zhupanska 2007); however, it is not practical for large systems due to the difficulty in visualizing apriori the interaction among all the components and determining its state transition probabilities. In addition, the sojourn time in CTMC is restricted to the exponential distribution.

This work proposes the use of Petri Nets to model and analyze search processes. Petri Net formalism allows us to visualize the structure of the rules-based system, making the model easier to understand, and to express the behavior of the system in mathematical forms (Lundell, Tang and Nygard 2005). Finally, Petri Nets and its extensions (general stochastic PNs (GSPN), extended SPNs (ESPNs) and deterministic SPNs (DSPN)) offer an activity-oriented formalism that facilitate the use of discrete event system analysis tools, including simulation and numerical analysis, to study the performance of systems.

1.5 Research Objective

In this thesis, we:

- Develop a Petri Net model to quantify the performance gains from cueing in cooperative search systems.
- Define and analyze system's performance measures for the proposed model

1.6 Proposal Organization

The rest of the proposal is organized as follows: Chapter 2 reviews the previous work in literature concerning search theory and performance gains due to cueing in cooperative search systems. Additionally, it introduces the theoretical foundations of Petri Nets**.** Chapter 3 describes the specific problem to be addressed and defines the performance measures to be used. Chapter 4 argues the relevance of general distributions in the problem addressed, and Chapter 5 shows how to analyze them in the proposed Petri Net model. Chapter 6 introduces software to analyze Petri Net models. Finally, Chapter 7 summarizes the contributions and outlines the future research directions.

Chapter 2

Literature Review

This chapter is divided into two sections. Section 2.1 defines search theory and presents several problems that have been addressed in this area throughout the years. In addition, it presents the research done concerning the performance gains due to cueing in cooperative search systems. Section 2.2 introduces the theoretical foundations of Petri Nets and its extensions.

2.1 Search Theory

Research in search theory was initiated during the Second World War by Bernard Koopman; who derived the probability of detection as a function of time, and studied the optimal allocation of search effort to detect a stationary target (Verkama 1996). Many applications of search theory have been orientated towards research with autonomous vehicles (Cole, et al. 2009), (Schultz, Parker and Schneide 2002), (Chandler and Pachter 2002). Depending on the level of human interaction, there are three types of autonomous vehicles (Committee on Autonomous Vehicles in Support of Naval Operations 2005):

- Scripted autonomous systems: use a preplanned script to accomplish the mission objective. These systems do not have human interaction after they are deployed. As an example consider guided rockets.

- Supervised autonomous systems: do most of the functions of planning, sensing, and networking to carry out activities. Human operators, via communication link, make decisions based on the data sensed by the vehicle.
- Intelligent autonomous systems: use intelligent technology to embed attributes of human intelligence in the software of autonomous vehicles and their controlling elements.

This research focuses on intelligent autonomous systems. Throughout the thesis, they are referred to as unmanned autonomous vehicles (UAVs). Some advantages of UAVs over manned aircraft systems include: no casualties, easier to store and ship, less expensive per aircraft and can fly longer missions.

Three different scenarios have been studied in search problems: single searcher, cooperation, and coordination. Uryasev et al. formulated the single searcher problem as a stochastic program. Their objective function was to minimize the expected search time before a target is found (Uryasev and Pardalos 2001). The total search area was divided in sub-regions and determined the average time that a searcher would require spending in a specific sub-region (assuming x targets within the search region). The work was extended into a cooperative search concept in which the search was concurrently performed by more than one vehicle. It was found that cooperative searching is not only dividing search effort among each agent; particularly when the target is able to detect and evade searchers. The cooperative search problem was approached with two opposing objectives: maximize the effectiveness of a single searcher and maximize the effectiveness of the group with multiple searchers.

The second scenario studied in search problems is cooperative search. Some missions are better performed by a team of UAVs instead of only one single agent. For instance a dangerous and/or extensive mission, where it is unlikely that a single UAV survives to complete the task, a team is more suitable to perform it (Cole, et al. 2009). Several authors concentrated on cooperative search. Polycarpou et al. developed and evaluated the performance of strategies for cooperative search with autonomous vehicles that seek to gain information about the environment (Marios, Yang and Liu Yang 2003). The vehicles share the information that they have to enable cooperation. No vehicle tells another what to do nor are there any negotiations among them. Each seeks to enhance a global goal, not only its own goal.

Chandler and Pachter looked at cooperative rendezvous and cooperative target classification and attack in a hierarchical distributed control system (Chandler and Pachter 2002). The vehicle doing path planning and trajectory generation is at Decision Level 1. At Decision Level 2 is the sub-team that coordinates the activities of classification and attack. When more than one vehicle is used to search and attack, the decision whether to continue the search or go attack previously found targets has to be made. This decision making process leads to the work done in *optimal stopping*. The theory of optimal stopping studies the problem of choosing a time to take a particular action, in order to maximize an expected reward or minimize an expected cost. In this work, optimal stopping is not considered. However, a good direction for future research on the problem addressed in this thesis might come from this area.

The third and last scenario studied in search problems is coordinated search. While cooperation entails more than one entity working toward common goal, coordination implies a coupling between entities that is designed to achieve the common goal (Hsieh, et al. 2007). An example of coordinated task execution is provided in (Schultz, Parker and Schneide 2002). A robot should not start analyzing a rock until two others have moved into place to provide assistance. Thus, a distributed executive facilitates one robot monitoring the execution of another robot and helps it recover from faults.

The search problem scenario studied in this research is cooperation. The effects of cueing in cooperative search system have been studied in (Alexander and Jeffcoat 2007). It is demonstrated that cueing increases significantly the probability of detection over a fixed period of time and that its effect on system's effectiveness is bounded. Continuous time Markov chain is used to model the cooperative search system and Kolmogorov equations are solved to determine the effects of cueing on the system's effectiveness. This is viable for small cooperative search systems; however, it is not practical for large systems due to the difficulty in visualizing apriori the interaction among all the components and determining its state transition probabilities. Hence, this work proposes the use of Petri Nets to model and analyze search processes.

2.2 Petri Nets

2.2.1 Motivation for the Use of Petri Nets to Model and Analyze UAV Systems

Petri Nets have proven to be very useful in the modeling, analysis, simulation, and control of UAV systems (Cao, Fukunaga and Kahng 2004), (Palamara, et al. 2009). They provide very useful models for the following reasons:

- Petri Nets capture the precedence relations and structural interactions of stochastic, concurrent, and asynchronous events. In addition, the graphical interface helps to visualize such complex systems (Desrochers and Al'Jaar 1995).
- Petri Net models represent a hierarchical modeling tool with a well-developed mathematical and practical foundation.
- Petri Nets and its extensions (general stochastic PNs (GSPN), extended SPNs (ESPNs) and deterministic SPNs (DSPN)) allow for both qualitative and quantitative analysis of performance measures (Ajmone, et al. 1994).
- The analysis of timed Petri Nets can be automated and several software tools such as SPNP and TimeNET are available for this purpose.
- Finally, Petri Net models can also be used to implement real-time control systems for UAVs (Cao, Fukunaga and Kahng 2004).

2.2.2 Formal Definition and Basic Terminology of Petri Nets

A Petri Net is graphically represented by a directed graph with two kinds of nodes: *places* and *transitions*. Place nodes model states or conditions, while transition nodes model events of functions of the system (Ajmone, et al. 1994). Petri Nets (PNs) are intended to visualize the dynamics of a system. The state of a Petri Net is called *marking*, and is defined by the number of tokens in each place. Each place may be considered as a local state of the system; it describes the condition of a resource. Places and transitions are connected by arcs. According to certain rules, the transition can move the tokens from one place to another, and thus change the state of the system. Formally, a Petri Net can be defined as follows:

- PN = *(P, T, I, O, Mo)*; where
- $P = \{p1, p2, \ldots, pm\}$ *is a finite set of places,*
- $T = \{t_1, t_{2,\ldots,} tn\}$ is a finite set of transitions, $P \cup T \neq \emptyset$ places, and $P \cap T \neq \emptyset$,
- $I: (P \times T) \rightarrow N$ is an input function that defines the arcs from places to transitions, where N *is a set of nonnegative integers,*
- $O: (P \times T) \rightarrow N$ is an output function which defines directed arcs from transitions to places, and
- $M_o: P \rightarrow N$ is the initial marking. It gives the numbers of indistinguishable tokens which are initially in each place.

In the graphical representation places are drawn as circles, transitions are drawn as rectangles, and arcs have an arrowhead at their destinations. Tokens are drawn as black dots; larger number of tokens in a place is represented by their number. A simple example of a Petri Net is shown in Figure 1.

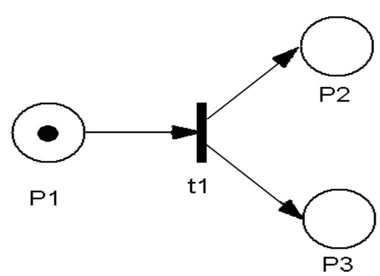


Figure 1 - Petri net example

The occurrence of events or execution of operations in a Petri Net model changes the distribution of tokens in places. Thus, one can study dynamic behavior of the modeled system.

The following rules are used to govern the flow of tokens:

- A transition t is said to be enabled in marking M, if at least one token is in all input places.
- An enabled transition t can fire by removing a token from each input place and putting one token in each output place.

2.2.2.1 Reachability Set and Reachability Graph

The firing rule defines the dynamics of Petri Net models. From initial marking is possible to determine the set of all markings reachable from it and all the paths that the system may follow to move from marking to marking. The initial state must be completely specified for the computation of the set of reachable markings. The representation of all reachable markings (state space of the net) is called reachability graph.

2.2.2.2 Stochastic Timed Petri Nets (STPN)

STPN are Petri Nets in which stochastic firing times are associated with transitions. The transitions times are allowed to be random variables.

2.2.2.3 Generalized Stochastic Petri Nets (GSPN)

A GSPN is an extension of an SPN. The Petri Net contains two types of transitions: immediate transitions and timed transitions.

- Timed transitions are associated with random, exponentially distributed firing delays.

- Immediate transitions fire in zero time with firing probabilities.

In the graphical representation, timed transitions are drawn as thick bars and immediate transitions as thin bars. When a new marking is reached, it can be classified into two types. A marking that enables only timed transitions is called tangible, whereas a marking that enables at least one immediate transition is called vanishings. Markings of the latter type have zero sojourn time. An example of a GSPN is discussed in Section 3.2.

Chapter 3

Problem Description

This chapter is divided into 4 sections. Section 3.1 discusses the basis of the work in this thesis. The problem addressed in this section was first introduced in (Jeffcoat, Krokhmal and Zhupanska 2007). Section 3.2 presents the Petri Net model proposed to model the system studied. Section 3.3 shows how to analyze the system using the Petri Net model. Finally, Section 3.4 introduces the performance indices used to evaluate the system's performance.

3.1 System's Description

The cooperative search mission that is considered throughout this thesis presents a search and engage scenario. It includes two types of UAVs: (i) a dedicated search-only vehicle and (ii) n-search-engage vehicles. The job of the search-only vehicle is to provide cues to all search-engage vehicles. It is assumed that the search-only vehicle has better search capabilities than the search-engage vehicles; thus, it has a higher detection rate. The search-engage vehicles can engage one target only and it is assumed that search-engage vehicles do not cue each other. The mission is completed when the n-search-engage vehicles have engaged n targets. Therefore, it is assumed that there are at least n targets within the search area.

The search-engage vehicle searches (uncued) for a target until either it finds it, or it is cued. If it is cued, it orientates its search towards the specified location. Eventually, it will find a target at the location provided by cue from the search-only vehicle. Figures 2 and 3 illustrate all the possible states of each type of UAV. As shown in Figure 2, the search-only vehicle has two possible states: (i) it can be either searching for targets or (ii) cueing search-engage vehicles. Traditionally the exponential distribution is used as detection function to model and analyze search processes (L. D. Stone 1983). The rate at which the search-only vehicle detects and cues search-engage vehicles is λ. As soon as it cues a search-engage vehicle, it starts over to look for new targets.

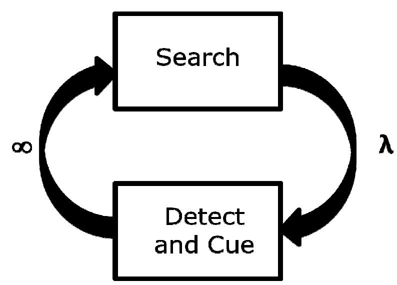


Figure 2 - Search-only vehicle states

As shown on Figure 3, the search-engage vehicle has three possible states: (i) search uncued, (ii) search cued or (iii) detect and engage a target. Initially, it is searching uncued and then it either engages a target with a rate θ_u or it is cued by the search-only vehicle and starts searching based on this cue. Similar to the search-only vehicle, the rate θ_u of the detection function comes from an exponential distribution. The rate in which the search-engage vehicle goes from searching uncued to searching cued is λ/n (It is assumed that there are *n* search vehicles and cues are equally distributed). From the search cued state, the vehicle engages a target with a rate θ_c. Since there is detailed information about the target's location in the search cued state, the rate $\theta_c > \theta_u$.

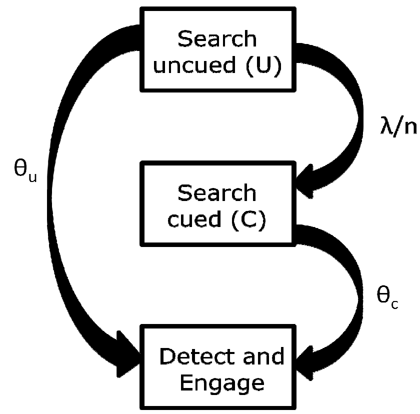
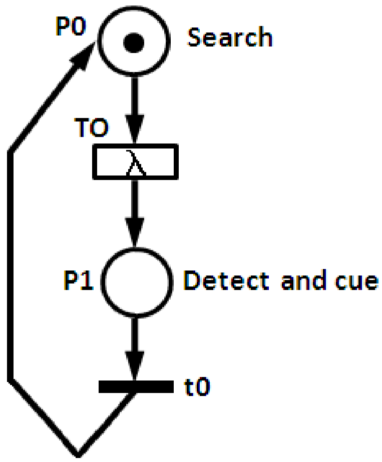


Figure 3 - Search-engage vehicle states

3.2 Petri Net Model

To develop a Petri Net model for the system described in Section 3.1, the two components of the search team are modeled individually. The Petri Net model for the search-only vehicle is shown in Figure 4. Place P0 represents the search state of the search-only vehicle. The time transition T0 represents the search-only vehicle detecting a target. Finally, the immediate transitions t0 corresponds to the event where the search-only vehicle returns to the search state after it cues a search-engage vehicle.

Similarly, the Petri Net model for a search-engage vehicle is shown in Figure 5. In this model, immediate transition t1 represents the cueing of this vehicle, and the time transitions T1 and T2 represents detection of a target before and after cueing respectively. Note the similarity in the structure of the two Petri Net models with the conceptual representation of the vehicles in Figures 2 and 3.



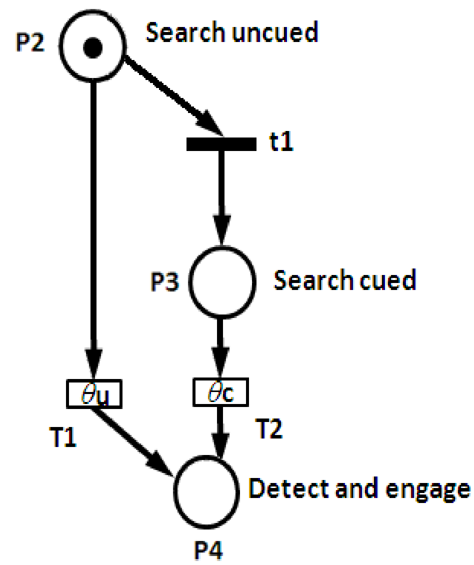


Figure 4 - Petri net search-only vehicle   Figure 5 - Petri net search-engage vehicle

Transition t0 in Figure 4 and transition t1 in Figure 5 represent the same event, namely cueing of a search-engage vehicle. Through this common event (transition), the component models are merged. In addition, a second search-engage vehicle can be integrated by adding an identical model to the one shown in Figure 5. Following the same approach, the Petri Net model of a system with several search-engage vehicles can be readily created. Figure 6 shows a Petri Net model for a system with two search-engage vehicles.

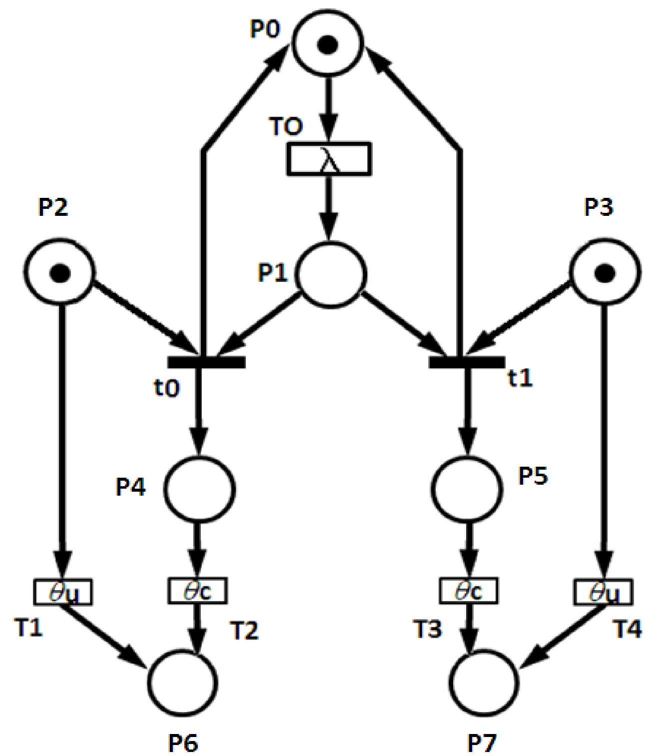


Figure 6 - One search-only vehicle and two search-engage vehicles

The complete model allows obtaining performance measures for a system of n search-engage vehicles and one search-only vehicle. An advantage of the proposed Petri Net is that the same model can be used to obtain system's performance measures with cueing or without cueing. If the token on P0 is removed, there is no cueing in the system. Thus, the same model structure can be used to quantify the effect of cueing in the performance of the system. The next sections explain the numerical analysis of the proposed model and introduce the performance indices of interest.

3.3 Analytical Solution

A GSPN describes an underlying stochastic process, captured by the reachability graph (RG). The analysis of a GSPN is, in principle, the analysis of its underlying process, which has been shown to be reducible to a CTMC. To make the reachability graph isomorphic, with a transition rate diagram of a CTMC, the vanishing markings have to be eliminated (Ajmone, et al. 1994).

3.3.1 Eliminating the Vanishing Markings

The following procedure is based on a system composed by one search-only vehicle and two search-engage vehicles (Figure 4). Table 1 shows the specification firing rates of the transitions in the GSPN of Figure 6.

Table 1 – Transition rates/weights of Figure 6

Transition	Rate/Weight
T0	λ
T1 = T4	θ_u
T2 = T3	θ_c
t0	α
t1	$1-\alpha$

The RG (Figure 7) contains 18 markings. The label on the arcs connecting two markings represents the time distribution to go from one marking to another one. There are only two types of distributions in the RG: exponential with rate μ_i , $\mu_i = \lambda, \theta_u$, or θ_c. and constant with k_0= 0. In addition, the label in square brackets corresponds to the probability that the arc is traversed. The markings represented with a dashed line are vanishing markings.

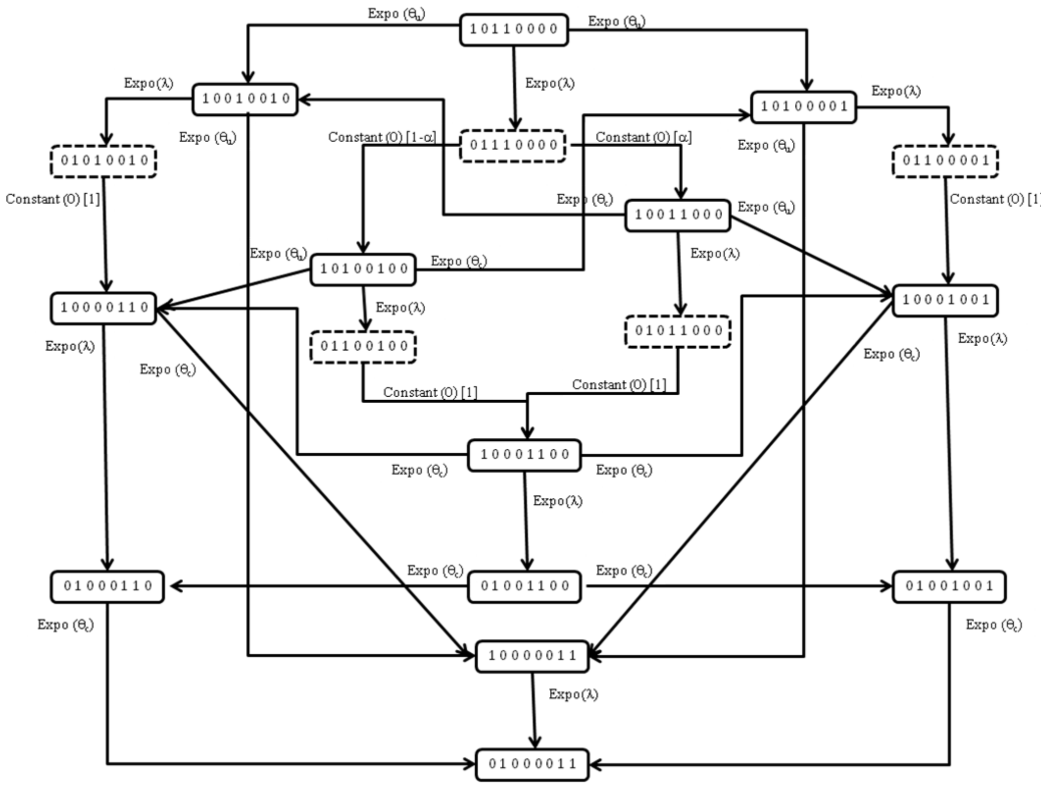


Figure 7 – Reachability graph of the GSPN of Figure 6

From the 18 marking of the reachability graph, there are 13 tangible markings and 5 vanishing markings. The vanishing marking can be eliminated by determining the equivalent rates of moving between two tangible markings with intermediate vanishing markings. The rate of moving from the marking (10110000) to the vanishing marking (01110000) is *λ*. The probability of leaving the vanishing marking to the marking (10011000) is *α*. Hence, the equivalent rate of moving from the (10110000) to (10011000) is:

$$r_{(10110000,10011000)} = \lambda \times \alpha \qquad (1)$$

Using the same procedure for each pair of tangible markings with intermediate vanishing markings, the reachability graph of the GSPN can be converted to a transition rate diagram of a CTMC (Figure 8).

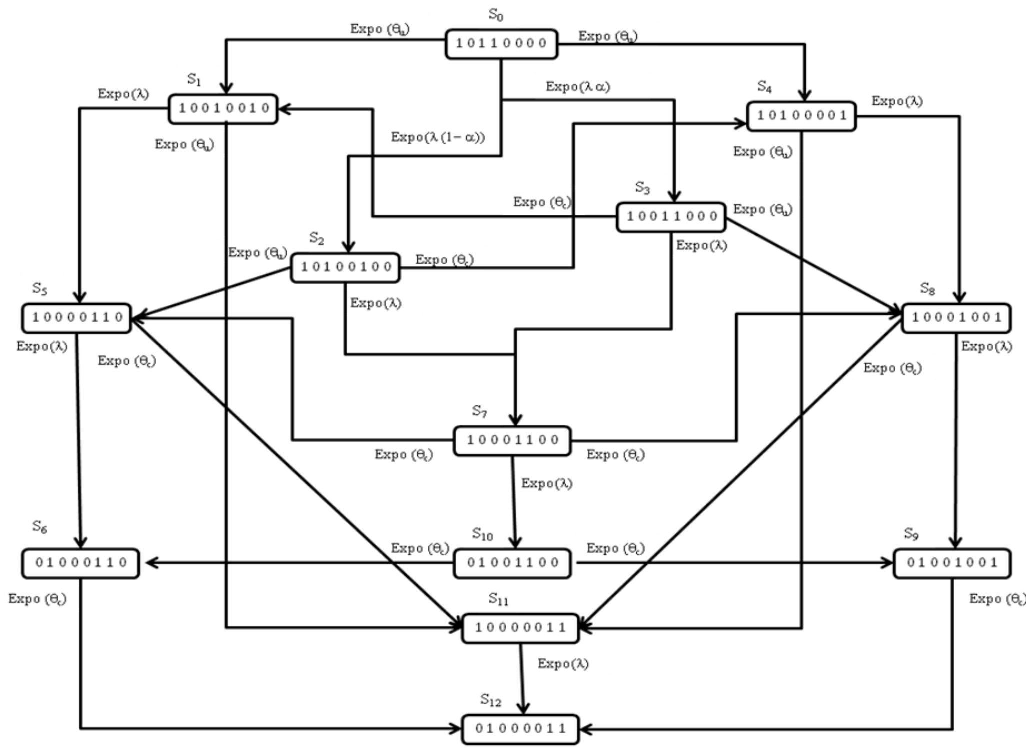


Figure 8 – CTMC rate diagram of the GSPN of Figure 6

The CTMCs allow obtaining system's performance measures such as mean time to complete a mission or probability of engaging a target by time *t*. The procedure of going from the reachability graph to the CTMC can be automated and is computationally acceptable as long as the number of vanishing marking is small compared to the number of tangible markings (Ajmone, et al. 1994). In addition, other procedures that reduce computational complexity have been studied (Miner 2001), (Allmaier, Kowarschik and Horton 1997).

3.4 System's Performance Measures

There are many situations that require performing a task or completing an objective in a certain amount of time. As an example, consider a boat that sank in cold water with 3 fishermen. The targets of the mission are the three possible survivors and they can die from hypothermia in a few hours. Thus, it is imperative to determine the probability that a specific team of UAVs can find the fishermen by time t. The performance measures will allow us to obtain the probability that a team of one search-only vehicle and n-search-engage vehicles engage m targets by time t ($m \leq n$).

The performance indices that are going to be defined to be able to measure the system's effectiveness with one search-only vehicle and n search-engage vehicles are:

- Expected time to engage n targets with n search-engage vehicles; one target for each vehicle.
- Expected numbers of targets engaged by the system as a function of time.

In both cases, it is assumed that there is at least n number of targets and that each search-engage vehicle can engage one target only.

3.4.1 First Passage Times in CTMCs

The first passage times in CTMCs is used to calculate the expected time to engage n targets with n search-engage vehicles (Kulkarni 1999). Let $\{X(\mathrm{t}), \mathrm{t} \geq 0\}$ be a CTMC with state space S={1,…,N} and rate matrix R. The first passage time into state N is defined to be:

$$T = \min \{t \geq 0: X(t) = N\}$$

let,

$$m_i = \mathrm{E}(T \mid X_0 = i) \;\; ; \quad m_n = 0$$

The next theorem gives a method of computing m_i, $1 \leq i \leq N - 1$. Theorem 2[1]: (First Passage Times) · $\{m_i, 1 \leq i \leq N - 1\}$ satisfy the following:

$$r_i m_i = 1 + \sum_{j=1}^{N-1} r_{i,j} m_j, \qquad 1 \leq i \leq N-1 \tag{2}$$

Theorem 2 can also be extended to the expected time to reach a set of states. The transition rate matrix of the CTMC for one search-only vehicle and two search-engage vehicles is shown in Figure 9.

	S_0	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}	S_{11}	S_{12}
S_0		θ_u	$\lambda(1-\alpha)$	$\lambda(\alpha)$	θ_u								
S_1						λ						θ_u	
S_2					θ_c	θ_u		λ					
S_3		θ_c						λ	θ_u				
S_4									λ			θ_u	
S_5							λ					θ_c	
S_6													θ_c
S_7						θ_c			θ_c		λ		
S_8										λ		θ_c	
S_9													θ_c
S_{10}							θ_c			θ_c			
S_{11}													λ
S_{12}													

Figure 9 – Transition rate matrix of the CTMC for one search-only vehicle and two search-engage vehicles

[1] For a proof of Theorem 1 see (Kulkarni 1999).

All the search-engage vehicles engage a target as soon as the CTMC visits the set of states {11, 12}; which is the markings [10000011] and [01000011], respectively. Then, Theorem 2 needs to be extended to the case of reaching a set. The first passage time to reach a set of states, A is:

$$T = \min\{n \geq 0: X(t) \in A\}$$

Let, $m_i\,(A)$ be the expected time to reach the set A starting from state $i \notin A$. Then,

$$r_i m_i(A) = 1 + \sum_{j \notin A} r_{i,j} m_j(A), \quad i \notin A \quad ; \; m_i(A) = 0 \text{ for } i \in A \tag{3}$$

From the transition rate matrix (Figure 9) and Theorem 2 extended to the case of reaching a set of states (3), the following can be obtained:

$$(\theta_u + \lambda(1-\alpha) + \lambda(\alpha) + \theta_u)m_0 = 1 + \theta_u m_1 + \lambda(1-\alpha)m_2 + \lambda(\alpha)m_3 + \theta_u m_4 \tag{4}$$

$$(\lambda + \theta_u)m_1 = 1 + \lambda m_5 \tag{5}$$

$$(\theta_c + \theta_u + \lambda)m_2 = 1 + \theta_c m_4 + \theta_u m_5 + \lambda m_7 \tag{6}$$

$$(\theta_c + \lambda + \theta_u)m_3 = 1 + \theta_c m_1 + \lambda m_7 + \theta_u m_8 \tag{7}$$

$$(\lambda + \theta_u)m_4 = 1 + \lambda m_8 \tag{8}$$

$$(\lambda + \theta_c)m_5 = 1 + \lambda m_6 \tag{9}$$

$$(\theta_c)m_6 = 1 \tag{10}$$

$$(\theta_c + \theta_c + \lambda)m_7 = 1 + \theta_c m_5 + \theta_c m_8 + \lambda m_{10} \tag{11}$$

$$(\lambda + \theta_c)m_8 = 1 + \lambda m_9 \tag{12}$$

$$(\theta_c)m_9 = 1 \tag{13}$$

$$(\theta_c + \theta_c)m_{10} = 1 + \theta_c m_6 + \theta_c m_9 \tag{14}$$

Table 2 summarizes the cueing and detection rates for the system; the same rates were used in (Jeffcoat, Krokhmal and Zhupanska 2007).

Table 2 – Detection rates and cueing weight

Rate	Value
λ	1.5
θ_u	0.10
θ_c	0.19
α	0.5

When the set of equations 4 – 14 are solved, the following values for the expected time to absorption from state m_i are obtained:

$$m_0 = 8.353,\ m_1 = 5.559,\ m_2 = 8.058,\ m_3 = 8.058,\ m_4 = 5.559,\ m_5 = 5.263,$$

$$m_6 = 5.263,\ m_7 = 7.895,\ m_8 = 5.263,\ m_9 = 5.263,\ m_{10} = 7.895$$

Thus, on average it takes 8.353 time units to engage two targets with one search-only vehicle and two search-engage vehicles. Table 3 summarize the average time for different values of λ, θ_u, and θ_c. The value of α was held constant at 0.5; which means that each of the two search-engage vehicles is equally likely to be cued.

Table 3 – Expected time to engage *n* targets with one search-only vehicle and two search-engage vehicles

	θ_u =0.10 θ_c = 0.19 E[T]		λ = 1.5 θ_u =0.10 E[T]		λ = 1.5 θ_c = 0.19 E[T]
λ = 1.5	8.353	θ_c = 0.19	8.353	θ_u =0.10,	8.353
λ = 2.5	8.174	θ_c = 0.29	5.814	θ_u =0.12,	8.246
λ = 3.5	8.095	θ_c = 0.39	4.583	θ_u =0.14,	8.142
λ = 4.5	8.051	θ_c = 0.49	3.859	θ_u =0.16,	8.041
λ = 5.5	8.023	θ_c = 0.59	3.383	θ_u =0.18,	7.943

Results on Table 3 imply that increments on the rate θ_c have a greater impact on the system's effectiveness than improving the cueing rate of the search-only vehicle (λ) or the individual uncued detection rates (θ_u). In other words, if resources were to be allocated toward decreasing the mean time to engage n targets, it is better to improve (increase) the rate θ_c; at least for the scenarios defined in the table.

The rate θ_c can be increased by:

- Improving the quality of the information provided in the cue. For example, if the search-only vehicle provides the exact location of the target to the search-engage vehicles.
- Cueing the closer search-engage vehicle to the location of the target. In the proposed model, the search-only vehicle chooses randomly among the search-engage vehicle to be cued. Choosing the vehicle that is closer to the target will increase the rate □c because it will decrease the time to engage the target after cueing.

3.4.2 Transient Analysis: Uniformization

The uniformization analysis in CTMCs is used to calculate the expected number of targets engaged by time t. Let $\{X(t), t \geq 0\}$ be a CTMC with state space $S = \{1,\cdots,N\}$ and let $R = \{r_{i,j}\}$ be its rate matrix. A CTMC spends an Exp(r_i) amount of time in state i ($r_i = \sum_{j=1}^{N} r_{i.j}$), and if $r_i > 0$, jumps to state j with probability $p_{i,j} = r_{i,j} / r_i$.

Now, let r be any finite number satisfying $r \geq \max_{1 \leq i \leq N} \{r_i\}$.

Define a matrix $P = [p_{i.j}]$ as follows:

$$p_{i.j} = \begin{cases} 1 - \frac{r_i}{r} & \text{if } i = j \\ \frac{r_{i.j}}{r} & \text{if } i \neq j \end{cases}$$

Finally, the transition probability matrix $P(t)$=[$p_{i.j}$(t)] is given by:

$$P(t) = \sum_{k=0}^{\infty} e^{-rt} \frac{(rt)^k}{k!} p^k \tag{15}$$

Using the detection rates in Table 3, we can obtain the following transition rate matrix of the CTMC for one search-only vehicle and two search-engage vehicles (Figure 10).

	S_0	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}	S_{11}	S_{12}
S_0		0.10	0.75	0.75	0.10								
S_1						1.50						0.10	
S_2					0.19	0.10		1.50					
S_3		0.19						1.50	0.10				
S_4									1.50			0.10	
S_5							1.50					0.19	
S_6													0.19
S_7						0.19			0.19		1.50		
S_8										1.50		0.19	
S_9													0.19
S_{10}							0.19			0.19			
S_{11}													1.5
S_{12}													

Figure 10 – Transition rate matrix of the CTMC for one search-only vehicle and two search-engage vehicles (rates Table 2)

Then r_1=1.7, r_2=1.6, r_3=1.79, r_4=1.79, r_5=1.6, r_6=1.69, r_7=0.19, r_8=1.88, r_9=1.69, r_{10}=0.19, r_{11}=0.38, r_{12}=1.5, r_{13}=0

hence, r = 1.88

Then, the *P* matrix is:

	S_0	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}	S_{11}	S_{12}
S_0	0.1	0.05	0.40	0.40	0.05	0	0	0	0	0	0	0	0
S_1	0	0.15	0	0	0	0.80	0	0	0	0	0	0.05	0
S_2	0	0	0.05	0	0.10	0.05	0	0.80	0	0	0	0	0
S_3	0	0.10	0	0.05	0	0	0	0.80	0.05	0	0	0	0
S_4	0	0	0	0	0.15	0	0	0	0.80	0	0	0.05	0
S_5	0	0	0	0	0	0.10	0.8	0	0	0	0	0.10	0
S_6	0	0	0	0	0	0	0.9	0	0	0	0	0	0.1
S_7	0	0	0	0	0	0.10	0	0	0.10	0	0.8	0	0
S_8	0	0	0	0	0	0	0	0	0.10	0.8	0	0.10	0
S_9	0	0	0	0	0	0	0	0	0	0.9	0	0	0.1
S_{10}	0	0	0	0	0	0	0.1	0	0	0.1	0.8	0	0
S_{11}	0	0	0	0	0	0	0	0	0	0	0	0.20	0.8
S_{12}	0	0	0	0	0	0	0	0	0	0	0	0	1

Figure 11 – P matrix for one search-only vehicle and two search-engage vehicles (rates Table 2)

Finally, *P(t)* can be computed by:

$$P(t) = \sum_{k=0}^{\infty} e^{-1.88t} \frac{(1.88t)^k}{k!} \, p^k \qquad (16)$$

In numerical computations, *P(t)* is approximated by using the first *M* terms of the infinite series. We compute *P(t)* by using the rule to choose the value of M proposed in (Kulkarni 1999):

$$M \approx max\{rt + 5 * \sqrt{rt}, 20\}$$

P(0.5) =

M = 20

	S_0	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8	S_9	S_{10}	S_{11}	S_{12}
S_0	0.43	0.03	0.16	0.16	0.03	0.02	0.01	0.11	0.02	0.01	0.03	0.00	0.00
S_1	0.00	0.45	0.00	0.00	0.00	0.33	0.16	0.00	0.00	0.00	0.00	0.04	0.02
S_2	0.00	0.00	0.41	0.00	0.04	0.04	0.02	0.30	0.03	0.01	0.15	0.00	0.00
S_3	0.00	0.04	0.00	0.41	0.00	0.03	0.01	0.30	0.04	0.02	0.15	0.00	0.00
S_4	0.00	0.00	0.00	0.00	0.45	0.00	0.00	0.00	0.33	0.16	0.00	0.04	0.02
S_5	0.00	0.00	0.00	0.00	0.00	0.43	0.48	0.00	0.00	0.00	0.00	0.04	0.05
S_6	0.00	0.00	0.00	0.00	0.00	0.00	0.91	0.00	0.00	0.00	0.00	0.00	0.09
S_7	0.00	0.00	0.00	0.00	0.00	0.04	0.04	0.39	0.04	0.04	0.44	0.00	0.00
S_8	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.43	0.48	0.00	0.04	0.05
S_9	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.91	0.00	0.00	0.09
S_{10}	0.00	0.00	0.00	0.00	0.00	0.00	0.08	0.00	0.00	0.08	0.83	0.00	0.01
S_{11}	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.47	0.53
S_{12}	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	1.00

Figure 12 – P matrix for one search-only vehicle and two search-engage vehicles at t =0.5 (rates Table 2)

From the first row of the P matrix in Figure 12, it can be seen that after 0.5 time units and starting at S_0 which is marking [10110000], there is a probability of 0.43 that the system is still on the same state. In addition, there is a 0.03 probability that the system has transitioned to state 1; which means that one of the search-engage vehicles has engaged a target by itself (no cue received). Using the probabilities provided by the matrix and the numbers of targets engaged in each state, we proceed to calculate the expected numbers of targets engaged by a specific time. For example, the expected number of targets engaged at 0.5 time units is:

Let $X(t)$ be the number of targets engaged at time t, then:

$$E(X(0.5)) = 0.43 * 0 + 0.03 * 1 + 0.16 * 0 + 0.16 * 0 + 0.03 * 1 + 0.02 * 1 + 0.01 * 1 + 0.11 * 0 + 0.02 * 1 + 0.01 * 1 + 0.03 * 0 + 0.0 * 2 + 0.0 * 2 = 0.1129$$

Following the same approach, we obtain a graph with the expected number of targets engaged as a function of time (Figure 13).

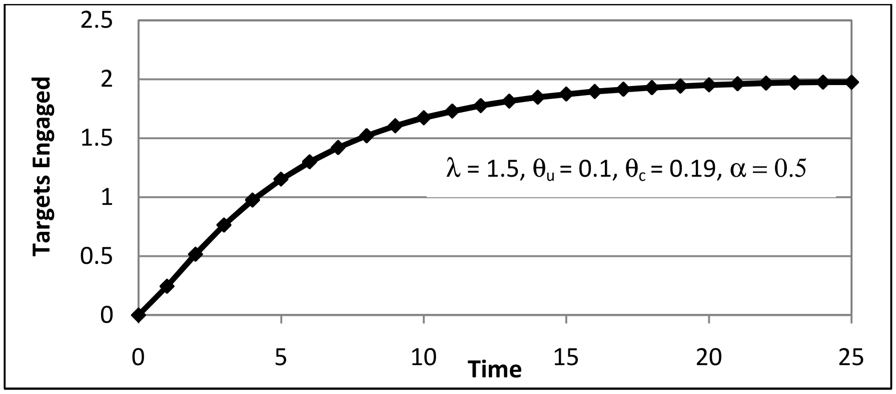


Figure 13 – Expected number of targets engaged with two search-engage vehicles

Figure 14 shows how the curve of the expected number of targets shifts to the left if one of the original rates is increased. The original rates were changed one at a time with an increment of 50%. The 50% is an assumption, and it can represent an improvement on the search capabilities, the speed of the vehicle, or any other factor that affect the rate at which targets are engaged.

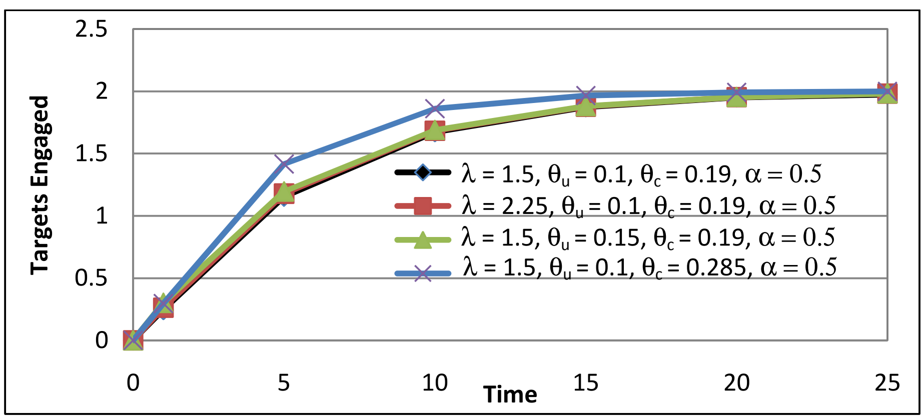


Figure 14 – Expected number of targets engaged with two improved search-engage vehicles

The results from Figure 14 agree with the findings of the previous section. Increments on the rate θ_c have a greater impact on the system's effectiveness than improving the cueing (λ) or the individual detection rates (θ_u).

Chapter 4

General Distributions

This chapter is divided into 3 sections. Section 4.1 describes the simulation developed with one search-only vehicle and one search-engage vehicle. Section 4.2 presents the results of the simulation and argues the relevance of general distributions in the problem addressed. Finally, Section 4.3 presents how to incorporate general distributions into the Petri Net model proposed.

4.1 Simulation Description

In Section 3.1 all the time distributions in the model follow an exponential distribution. A simulation was developed to determine whether this assumption is valid for systems with stationary targets and cues that provide exact target location. Since there is precise information about a target's location, better fits may come from bounded distributions. A cue with the precise location of a stationary target eliminates the need for any additional search by the search-engage vehicle and simplifies the process to traveling from one location to another. Arguably, this no longer is a search process; however, to preserve the association with the general cooperative search model introduced in Chapter 3, this process is still referred to as cued search in the rest of the thesis.

The environment in which the vehicles are searching is shown in Figure 15. The region simulated is a grid of *m* by *n* cells; the values of *m* and *n* are inputs. There are two vehicles in the simulation: one search-only vehicle, shown as a "2"; and one search-engage vehicle, shown as a "1". There is only one target, and it is represented as a "-1". It is assumed that the search-only vehicle covers more area than the search-engage vehicle; the area of coverage is represented by the shaded region around the vehicle and it is an input of the simulation. The search-engage vehicle has to be in the same cell with the target to find it. The initial positions of the vehicles and the target are randomly selected, between each replication, with a uniform distribution over the region of the search environment. Once the simulation starts, both vehicles look for targets until either the search-engage vehicle finds it, or the search-only vehicle detects it. If the search-only vehicle detects the target, it transmits the target's location to the search-engage vehicle. Then, the search-engage vehicle moves to specified position. The simulation ends when the search-engage vehicle finds the target.

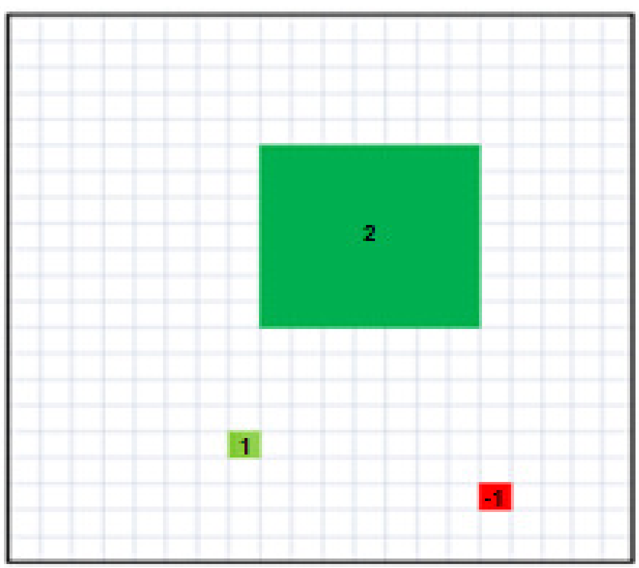


Figure 15 – Simulation environment

The vehicles do not follow any pattern nor have memory of the places they have visited. They are free to move in any direction within the limits of the search environment. However, once the search-engage vehicle receives the information about the target's location, it moves directly to the specified location via the fastest way to reach that position.

4.1.1 Assumptions for the Model

- The search-only vehicle provides the exact target's location to the search-engage vehicle.
- The time to move between cells is the same regardless of the direction. For example: moving to the north direction takes the same time as moving to the north-east direction.
- Both vehicles move at the same speed. Constant speed
- Target is stationary.
- Both vehicles can be in the same cell at the same time.

4.2 Simulation Results

The simulation was constructed to analyze the time distribution of the three search processes. Table 4 shows the histograms of the time distribution associated with each search process and vehicle. The results are based on 5,000 replications. From Table 4, it is seen that the assumption of the exponential distribution is valid for the time of the first two search processes. However, the histogram of the third search process (cued search) indicates that the exponential distribution is not a good fit. The exponential distribution is a good fit for the first two search processes because there is no information about the target's location. In the third search process, once the target's location is known, the time to engage a target is a function of the distance between the target and the search-engage vehicle, and its speed.

Table 4 – Histograms of the distributions of the time associated with each search process

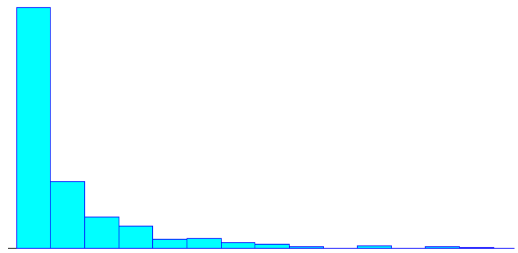
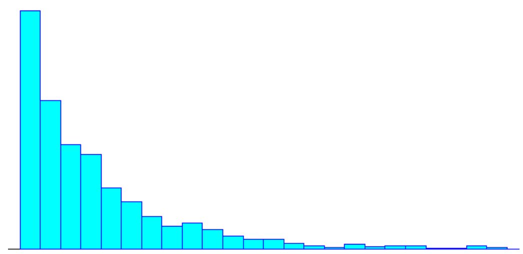
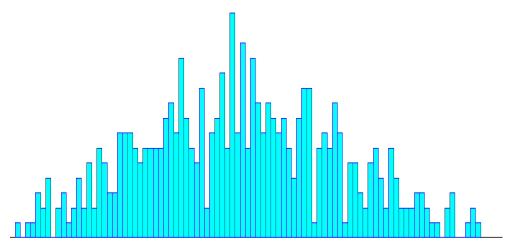
	Vehicle	Search Process	Histogram
1	Search-Only	Search and Cue	
2	Search-Engage	Uncued Search	
3	Search-Engage	Cued Search	

Table 5 shows the *p*-values of the chi-square test for the time of the cued search process. Corresponding *p*-values less than 0.05 indicate that the distribution is not a very good fit; larger *p*-values indicate better fits. It can be seen from Table 5 that the exponential distribution is not a good fit. In contrast, the triangular, the weibull and the normal distributions are better suited to model the underlying process time. Since the search time is a positive and bounded value, the triangular distribution is the most appropriate distribution to model the time of the cued search process.

Table 5 – *P*-values of the chi square test for the time of the cued search process

Functions	***p*-value**
Triangular	> 0.75
Normal	0.473
Weibull	0.454
Erlang	0.0092
Gamma	0.0869
Lognormal	< 0.005
Uniform	< 0.005
Exponential	< 0.005

The simulation results indicate that the time of the cued search process is better represented with a triangular distribution. The time to engage a target once the vehicle receives a cue is the distance traveled to the location provided times the speed (the cue transmitted gives the exact location). Thus, the time distribution of the cued search process is basically the distribution of the distance between two random points times a constant (the speed). It can be proven, using Manhattan metrics, that the distance between two uniformly distributed random points within a rectangle follows a triangular distribution. The proof is shown in (Gaboune, et al. 1993) and it is summarized below.

Denote (X_i, Y_i) a point in a rectangle. Consider two random points and define $X = |X_1 – X_2|$, $Y = |Y_1 – Y_2|$. Also let L denote the average distance between two uniformly randomly distributed points in the rectangle. For $0 \leq x \leq a$, the distribution function of X is given by:

$$F_x(x) = P(|X_1 - X_2| \leq x)$$

$$= 1 - [\,P(X_2 \geq X_1 + x) + P(X_2 \geq X_1 - x)]$$

$$= 1 - \left[\int_0^{a-x} \int_{x_1+x}^{a} f(x_1, x_2)\, dx_2 dx_1 + \int_x^{a} \int_0^{x_1-x} f(x_1, x_2)\, dx_2 dx_1 \right],$$

where f (x1, x2), the joint probability function of X1 and X2, is defined by:

$$f(x_1, x_2) = \begin{cases} \frac{1}{a^2} & if \;\; 0 \le x_1 \le a \quad and \quad 0 \le x_1 \le a \\ 0 & otherwise \end{cases}$$

since X1 and X2 are independent. Therefore

$$F_x(x) = \begin{cases} 0 & if \; x \le 0 \\ 1 - \dfrac{(x-a)^2}{a^2} & if \; 0 \le x \le a \\ 1 & if \; x \ge 0 \end{cases}$$

hence, the density function of X is triangular over [0, a]:

$$f_x(x) = \begin{cases} \dfrac{2}{a}\left(1 - \dfrac{x}{a}\right) & if \; \le \; x \; \le a \\ 0 & \text{otherwise} \end{cases}$$

The next chapter discusses how to incorporate general distributions such as the triangular distribution to the proposed Petri Net model and calculate performance measures.

Chapter 5

General Distributions Analysis in the Petri Net Model

It has been demonstrated that cueing increases the performance of a cooperative search system and that the proposed Petri Net model captures the interactions among the vehicles in the system. Performance indices are defined and computed to measure system's performance. In addition, these indices can be used to decide how to best allocate resources to improve the system's performance. Finally, the cued search process time is shown to be accurately represented by a triangular distribution.

The feasible techniques to obtain performance measures in Petri Nets with general distributions are simulation and approximation (Van der Aalst, Van Hee and Reijers 2000). Simulation will not be addressed in this thesis. This chapter discusses how to approximate general distributions. Section 5.1 derives a general expression for the coefficient of variation for the general distribution using the search area dimensions to determine the type of approximation. Section 5.2 discusses how the triangular distribution can be incorporated into the proposed Petri Net model.

5.1 Analysis of General Distributions

Agner Erlang conceived the notion of decomposing general distributions into phase-type distributions (Yee and Ventura 2000). He showed that a distribution with a coefficient of variation (CV) less than one can be represented by a series of $k \geq 2$

exponential stages; this is known as an Erlang-k distribution. On the other hand, a distribution with a CV greater than one can be represented by $k \geq 2$ parallel exponential stages; this is known as the hyper-exponential distribution (Chen, Bruell and Balbo 1989). This procedure will be used to approximate a triangular distribution to an Erlang-k or hyper-exponential distribution (depending on the CV) and incorporate it into the proposed Petri Net model.

The density function of the distance between the target and the search-engage vehicle was derived in Section 4.2 as function of the size of the search environment. The density function is:

$$f_x(x) = \begin{cases} \frac{2}{a}\left(1 - \frac{x}{a}\right) & if \ \leq \ x \ \leq \ a \\ 0 & \text{otherwise} \end{cases}$$

Then, the expected value *E(X)* and the variance *V(X)* can be computed to obtain a general expression for the coefficient of variation.

$$E[X] = \int_{-\infty}^{\infty} x\, fx\, dx = \int_0^a x\left(\frac{2}{a}\left(1 - \frac{x}{a}\right)\right) dx = \int_0^a \frac{2x}{a}\, dx - \int_0^a \frac{2x^2}{a^2}\, dx$$

$$= \left.\frac{2x^2}{a}\right|_0^a - \left.\frac{2x^3}{3a^2}\right|_0^a = a - \frac{2a}{3} = \frac{1}{3}a$$

E[X] refers to the expected distance traveled along the *x* axis. The same way can be computed for the *y* axis.

$$E[Y] = \frac{1}{3}b$$

Hence,

$$E(L) = E\ (\ X + Y\ ) = E\ (X) +\ E(Y) =\ \frac{a}{3} + \frac{b}{3} =\ \frac{(a+b)}{3}$$

$$V[X] = E[x^2] - (E[X])^2$$

$$E[x^2] = \int_{-\infty}^{\infty} x^2\ fx\ dx = \int_0^a x^2\ \left(\frac{2}{a}\left(1 - \frac{x}{a}\right)\right) dx =\ \ \int_0^a \frac{2x^2}{a}\ dx - \int_0^a\ \frac{2x^3}{a^2}\ dx$$

$$= \left.\frac{2x^3}{3a}\right|_0^a - \left.\frac{x^4}{2a^2}\right|_0^a = \frac{2a^2}{3} - \frac{a^2}{2} = \frac{1}{6}a^2$$

$$(E[x])^2 =\ \left(\frac{a}{3}\right)^2 =\ \frac{a^2}{9}$$

$$V[X] = \frac{a^2}{6} - \frac{a^2}{9} = \frac{a^2}{18}$$

V[X] refers to the variance on the distance traveled along the *x* axis. The same way can be computed for the *y* axis.

$$V[Y] = \frac{b^2}{6} - \frac{b^2}{9} = \frac{b^2}{18}$$

$$V[L] =\ V[X+Y] = V[X] +\ V\ [Y] -\ 2\ cov\ (x+y)$$

X and *Y* are independent; thus, cov (x + y) = 0

$$V[L] =\ \frac{a^2}{18} + \frac{b^2}{18} =\ \frac{a^2+b^2}{18}$$

Hence, the coefficient of variation is:

$$C_v[L] = \frac{V[L]}{E[L]} = \frac{\frac{a^2 + b^2}{18}}{\frac{(a + b)}{3}} = \frac{a^2 + b^2}{6\,(a + b)}$$

The general expression of the coefficient of variation allows for associating the type of approximation needed for the general distribution of the cued search process to the search boundaries. The next section discusses how to approximate general distributions and to incorporate the approximation into the proposed model.

5.2 Incorporating General Distributions into the Proposed Model

Any type of general distribution with support on [0, ∞) can be approximated by a phase-type distribution (Asmussen, Nerman and Olsson 1996). Phase-type distributions have been successfully used for modeling non-exponential activities due to their versatility and relative ease of numerical implementation (Shaked and Shanthikumar 2006).

Several methods have been utilized for approximating general distributions. A general statistical approach called the EM algorithm is presented in (Asmussen, Nerman and Olsson 1996). EM algorithm can be used to approximate incomplete data and continuous distributions with support on [0, ∞). Approximating a continuous distribution by a phase-type distribution is similar to fitting a phase-type distribution to a sample. In parametric estimation, the methods that minimize the divergence between the assumed model density and the true density underlying the data, include maximum likelihood, chi squared methods based on families of chi-squared distances and Hellinger distance, among others (Basu, et al. 1998). A benchmark for phase-type estimation algorithms is presented in (Bobbio and Telek 1994).

To illustrate how a triangular distribution is approximated by a phase-type distribution and incorporated into the proposed Petri Net model, let us assume the following parameters for a triangular distribution (2, 10, 18). First, the coefficient of variation is estimated:

$$c_v = \frac{\sigma}{\mu}.$$

The standard deviation and the mean of a triangular distribution are defined as:

$$\sigma = \sqrt{\left(\frac{a^2 + b^2 + c^2 - ab - ac - bc}{18}\right)} = 3.27$$

$$\mu = \frac{a + b + c}{3} = 10$$

Hence, the coefficient of variation is:

$$CV = \frac{3.27}{10} = 0.327 < 1$$

The CV is less than 1; thus, the triangular distribution can be approximated by a series of $k \geq 2$ exponential stages (Erlang-k distribution). Then, we proceed to estimate the number of stages (k) and the mean time of each one (u). EasyFit software (Technologies, 2004) is used as a tool for estimating the parameters and performing the goodness of fit tests. The following describes how the tool is used.

Initially, a sample set of 5,000 was simulated from the triangular distribution (2, 10, 18). This data was used to estimate the parameters of the Erlang distribution. The larger the sample size the more power[2] the statistical test has (Montgomery and Runger 2002). Thus, with a large sample size the test is more likely to reject the null hypothesis that the Erlang distribution is the true distribution of the data. A second sample of size 200 was simulated to compare the results of the goodness of fit tests.

Figure 16 shows the histograms of the simulated data and the probability density functions of the fitted Erlang distribution for the two samples.

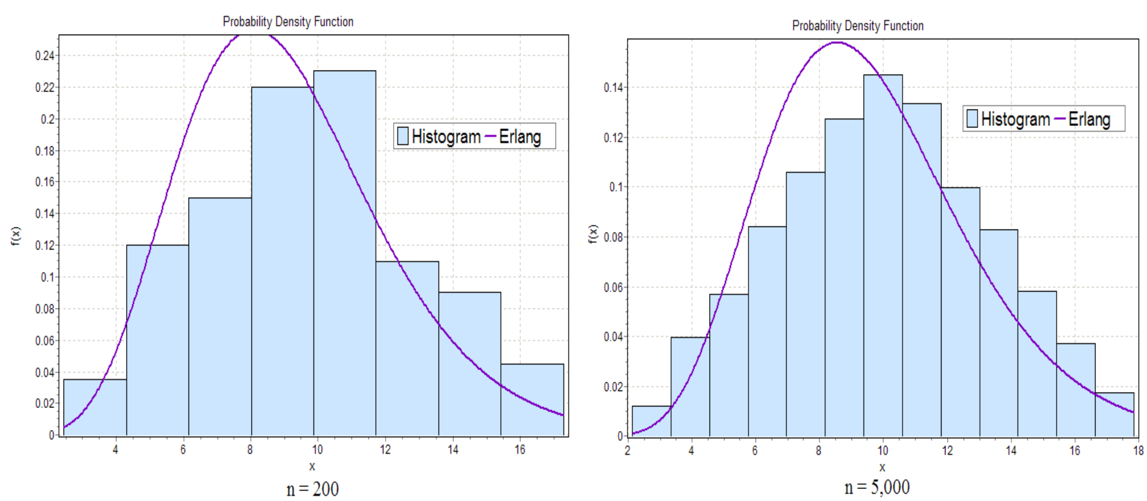


Figure 16 – Erlang-k approximation of the triangular distribution (2, 10, 18)

The parameters estimated for the Erlang distribution are $k = 9$ and $u = 1.0093$ and $k = 9$ and $u = 1.0681$ with the sample size of 200 and 5,000, respectively. Table 6 summarizes the results for the Kolmogorov-Smirnov, Anderson-Darling, and Chi-Squared tests. For the sample set of size 5,000, all the statistical tests reject the null

---

[2] The power of a statistical test is the probability of rejecting the null hypothesis H_o when the alternative hypothesis is true.

hypothesis that the data follow an Erlang distribution. However, for the set of size 200, the Chi-Squared test fails to reject the null hypothesis ($\alpha \leq 0.05$).

Table 6 – Goodness of fit tests

	n = 200			**n = 5,000**		
Kolmogorov-Smirnov						
Sample Size			200			5000
Statistic			0.13325			0.09453
P-Value			0.00148			0
α	0.05	0.02	0.01	0.05	0.02	0.01
Critical Value	0.09603	0.10734	0.11519	0.0192	0.02147	0.02304
Reject?	Yes	Yes	Yes	Yes	Yes	Yes
Anderson-Darling						
Sample Size			200			5000
Statistic			5.8211			90.929
α	0.05	0.02	0.01	0.05	0.02	0.01
Critical Value	2.5018	3.2892	3.9074	2.5018	3.2892	3.9074
Reject?	Yes	Yes	Yes	Yes	Yes	Yes
Chi-Squared						
Deg. of freedom			7			12
Statistic			13.694			298.38
P-Value			0.05691			0
α	0.05	0.02	0.01	0.05	0.02	0.01
Critical Value	14.067	16.622	18.475	21.026	24.054	26.217
Reject?	No	No	No	Yes	Yes	Yes

Even though the results from the goodness of fit tests may indicate that the Erlang distribution is not consistent with the data, the approximation is widely used to do numerical analysis in Petri Nets with general distributions (Ajmone, et al. 1994), (Yee and Ventura 2000).

The proposed Petri Net model is adjusted to incorporate the approximation of the triangular distribution with a phase-type distribution. The Erlang distribution (9, 1.0093) is used for depicting what the proposed model looks like with a phase-type distribution.

The series of exponential distributions can be incorporated into the model by adding a series of places and time transitions. Figure 17 shows the Petri Net model for a search-engage vehicle with the Erlang (9, 1.0093). The 9 transient states (the phases) are represented in the model with transitions {T2, T3, T4, T5, T6, T7, T8, T9, T10}. The average time of firing each transition is 1.0093. The Petri Net model for a search-engage vehicle with the Erlang-k approximation can be incorporated into the system model to obtain system's performance measures (Figure 18). The next chapter introduces software that allows analyzing more complex systems such as the one depicted in Figure 18.

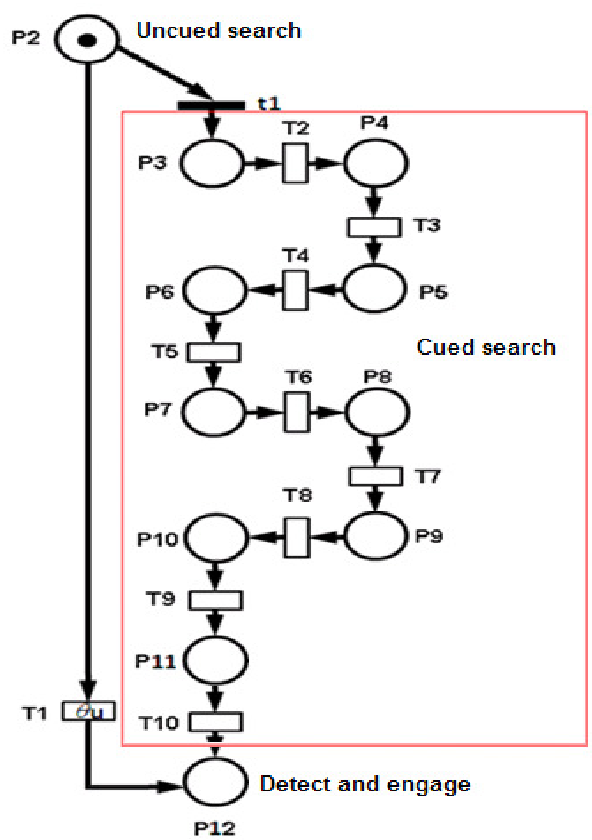


Figure 17 - Petri net model for a search-engage vehicle with Erlang-k approximation

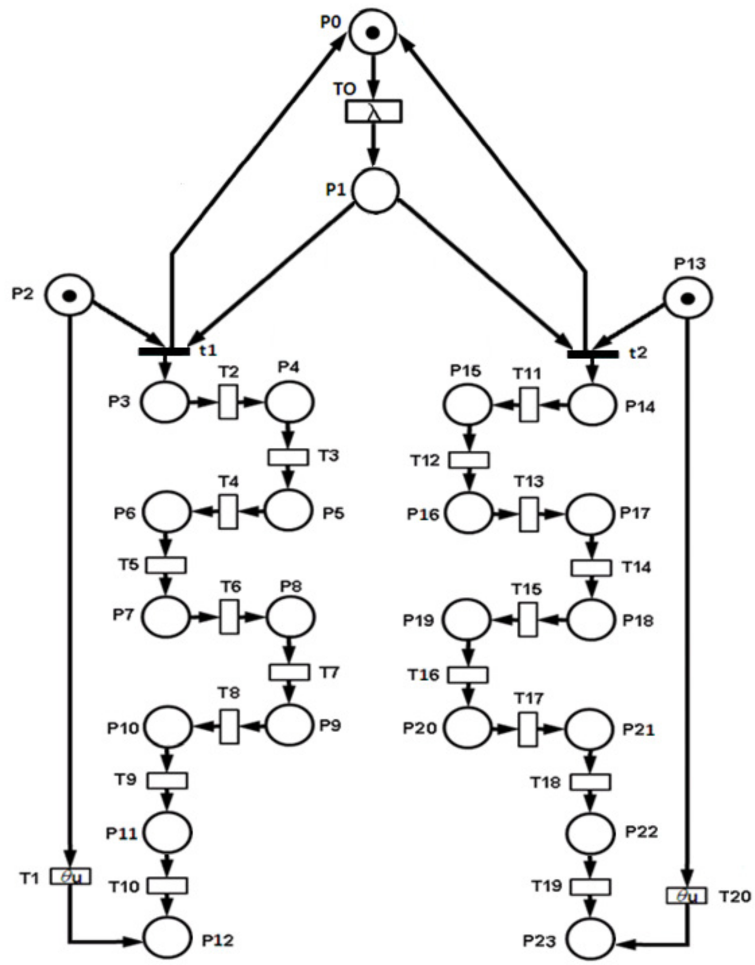


Figure 18 - One search-only vehicle and two search-engage vehicles with Erlang-k approximation

Chapter 6

Stochastic Petri Net Package (SPNP)

It has been demonstrated that cued search processes with stationary targets and cues that provide the exact target's location are better represented with triangular distributions, and it was shown how to incorporate them into the proposed model. This chapter introduces software that allows rapid development of stochastic reward nets (including GSPN) to evaluate performance measures. The name of the software is Stochastic Petri Net Package (SPNP) [3].

6.1 SPNP Description

SPNP is a modeling tool for performance analysis of complex systems. The model type used for input is a stochastic reward net (SRN) and they are specified using CSPL (C based SRN Language) which is an extension of the C programming language with additional constructs for describing the SRN models.

The SRN models are automatically converted into a Markov reward model which is then solved to compute a variety of transient, steady-state, cumulative, and sensitivity measures. For SRNs with absorbing markings, the mean time to absorption and the expected accumulated reward until absorption can be computed.

[3] A full description of the software and its capabilities can be found at:
http://people.ee.duke.edu/~chirel/MANUAL/manual.pdf

6.2 SPNP and Petri Net Model Validation

The same cooperative search system studied in (Jeffcoat, Krokhmal and Zhupanska 2007) was used to replicate its results and consequently validate the proposed Petri Net Model and the output of the software; before using the software to obtain performance measures. The cooperative search system consists of one search-only vehicle and five identical search-engage vehicles. Table 7 summarizes the transitions rates/weights.

Table 7 - Transition rates/weights of Figure 19

Transition	Rate/Weight
T0	λ
T1 = T3 = T6 = T8 = T10	θ_u
T2 = T4 = T5 = T7 = T9	θ_c
t0=t1=t2=t3=t4	α

The transition T0 represents the event where the search-only vehicle detects a target and cues one of the search-engage vehicles. The rate at which this event occurs is λ. The transitions T1,T3,T6,T8,T10 represent the event where a search-engage vehicle searches and engages a target without receiving any information from the search-only vehicle (no cue transmitted). The rate at which this event happens is θ_u. The transitions T2,T4,T5,T7,T9 represent the event where a search-engage vehicle detects and engages a target with information about its location; the rate is θ_c. This event only occurs if the search-engage vehicle receives a cue from the search-only vehicle. A parameter *k* (cueing effectiveness) was defined in (Jeffcoat, Krokhmal and Zhupanska 2007) to associate the increase in the detection rate due to the information transmitted in the cue; hence, $\theta_c = k$ x θ_u. The cues are distributed equally; thus, the probability of firing each immediate transition (t0,t1,t2,t3,t4) is the same. The Petri Net model for one search-only vehicle and five search-engage vehicles in SPNP is depicted in Figure 19.

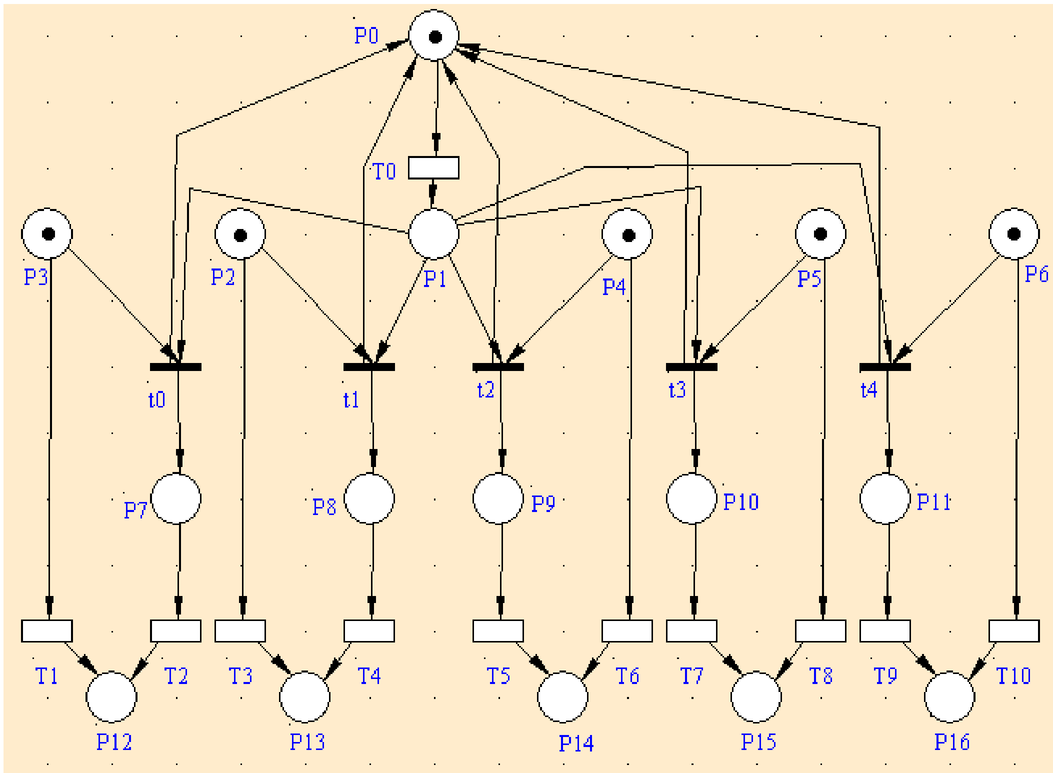


Figure 19 - Petri net model with one search-only vehicle and five search-engage vehicles in SPNP

Jeffcoat et al. measured the system's effectiveness by the probability that all search-engage vehicles have engaged targets by time t. They analyzed two different scenarios and presented their results in two graphs. Both scenarios have the initial detection rate θ_u of the search-engage vehicles equal to 0.1, but the search rate λ of the search-only vehicle varies in the first scenario and the cueing effectiveness k is varied in the second scenario. Figures 20 and 21 show the results for the two scenarios studied, respectively.

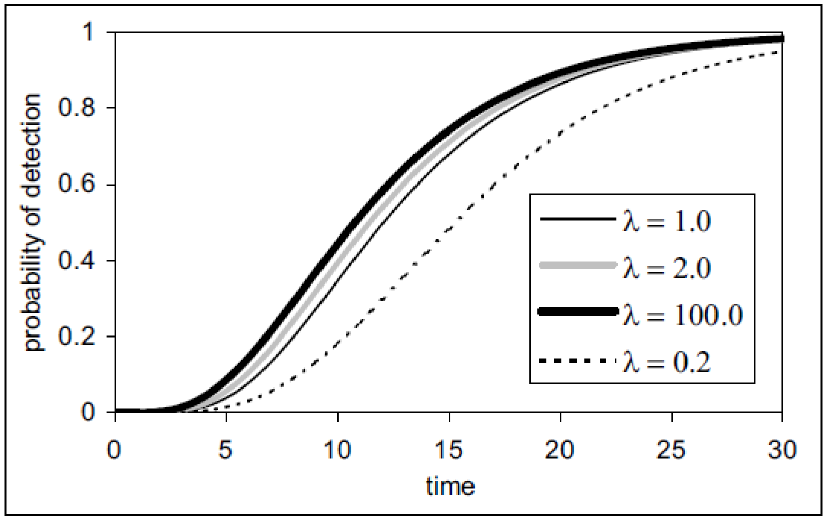


Figure 20 – Probability of detection of all search-engage vehicles varying λ

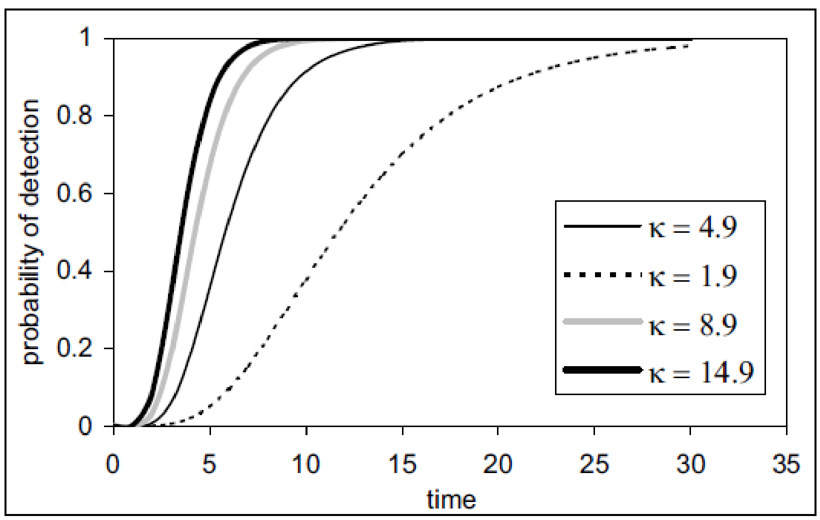


Figure 21 – Probability of detection of all search-engage vehicles varying *k*

In the proposed Petri Net model, all the search-engage vehicles have engaged targets by time t when the places P12, P13, P14, P15, and P16 have a token. The marking of interest is shown on Figure 22.

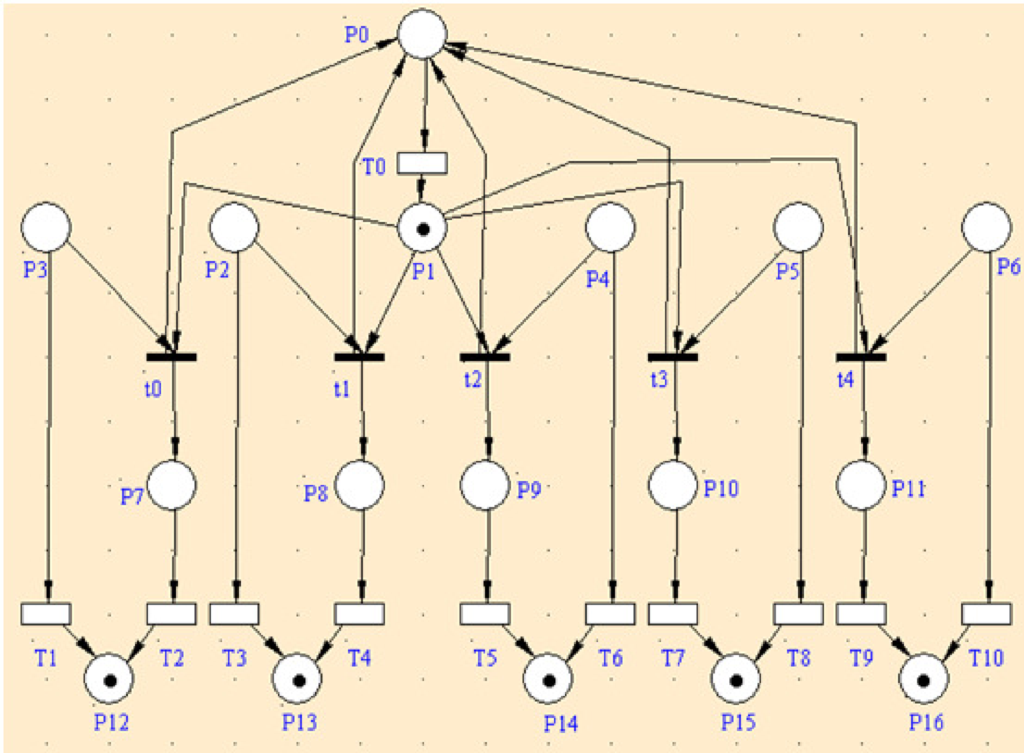


Figure 22 – Marking with targets engaged by all search-engage vehicles

SPNP allows obtaining the probability of reaching this marking as a function of time. The initial detection rate θ_u of the search-engage vehicles is held equal to 0.1, but the search rate λ of the search-only vehicle is varied according to the results presented in (Jeffcoat, Krokhmal and Zhupanska 2007). Figures 23 and 24 show the results obtained using SPNP and the Petri Net model for the two scenarios studied.

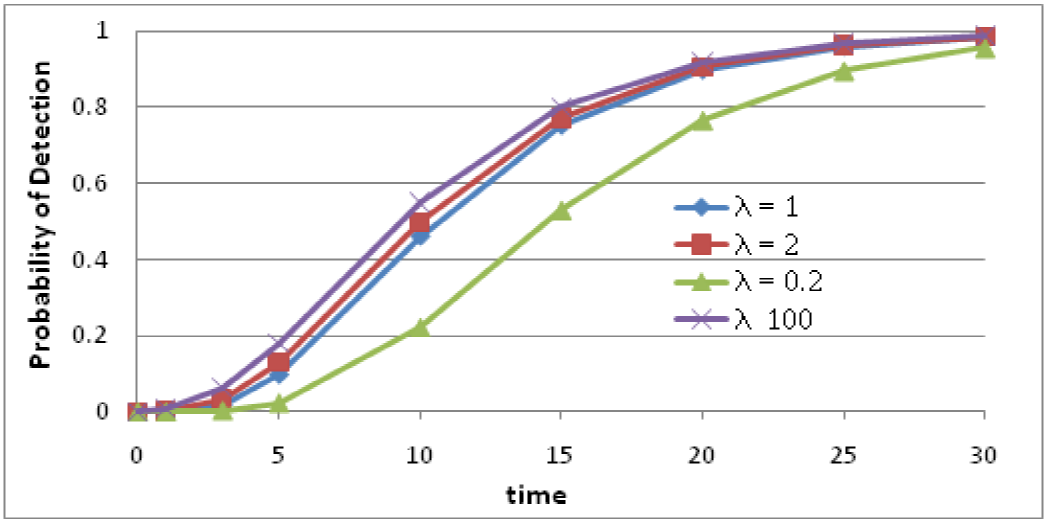


Figure 23 - Probability of detection of all search-engage vehicles varying λ (Petri net)

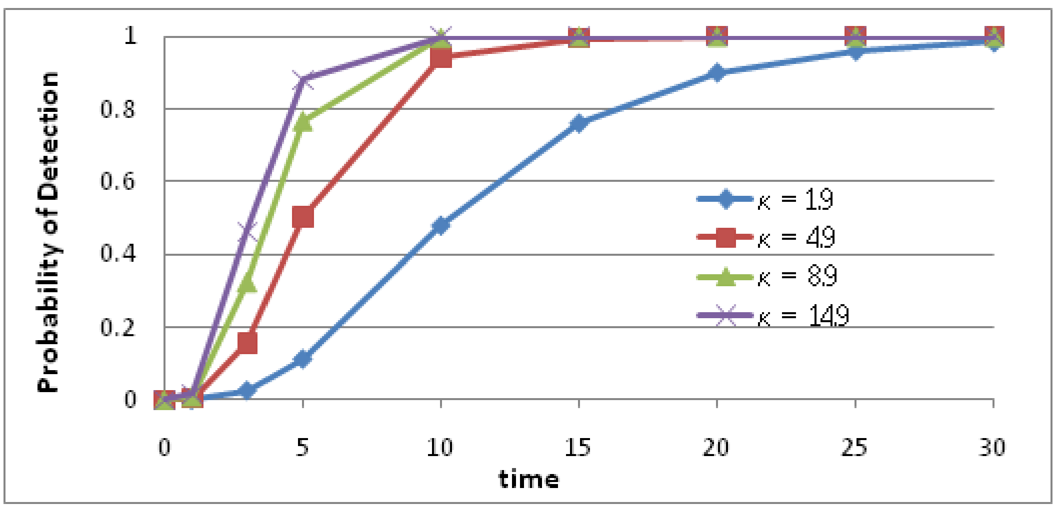


Figure 24 - Probability of detection of all search-engage vehicles varying k (Petri net)

From the comparison of the respective figures, it is clear that the Petri Net model and the results from SPNP are equal to the results in Figures 20 and 21 from (Jeffcoat, Krokhmal and Zhupanska 2007) verifying the correctness of the proposed modeling methodology. The next chapter summarizes the contributions of this thesis and outlines the future work that can be done in this research.

Chapter 7

Contributions and Future Research Directions

7.1 Contributions

This thesis presents a Petri Net based modeling approach to model the interaction among autonomous search vehicles in a cooperative search system. The cooperation among the vehicles involves cueing. Both in previous studies and in this thesis, it was demonstrated that cueing increases system performance. However, the concept of cueing has not been explored in detail and there is a lack of system models and modeling approaches that involve cueing.

The proposed modeling approach based on Petri Nets brings with it the well documented advantages associated with using Petri Net models, such as modularity, hierarchical modeling, well developed mathematical foundation, and a wide range of software available for model development and analysis.

In addition, the approach allows the analysis of similar systems using the same Petri Net structure greatly decreasing model development and verification effort. For example, in Figure 25 by removing the token from place P0, the search-only vehicle becomes inactive in the model (transition T0 is not enabled) eliminating the cueing capability of the system. The transition T0 will not fire and the search-engage vehicles will never be cued. This is an advantage that the Petri Net model has over CTMC because the same model can be easily modified to quantify the performance gains from cueing.

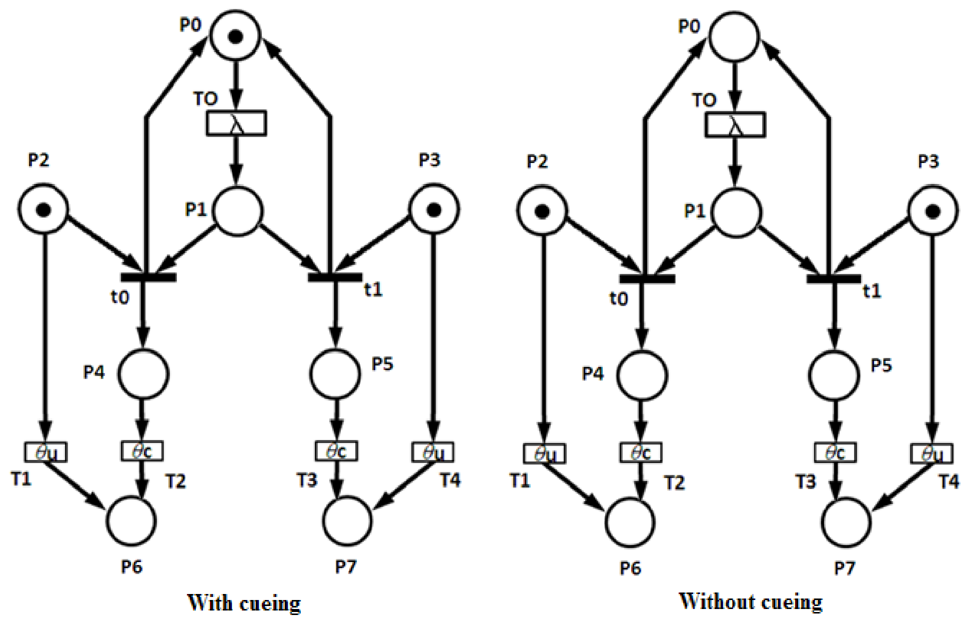


Figure 25 – Petri net model with cueing and without cueing

In the same manner, a subset of search-engage vehicles can be deactivated by removing their corresponding tokens from the Petri Net model. Figure 26 shows a Petri Net model with four search-engage vehicles but only two of them are active corresponding to marked places (P2, P11). Such a modification allows the system modeler to evaluate alternative scenarios with varying number of search-engage vehicles and analyze system's performance measures without constructing a new model for each scenario. The number of vehicles can also change during a mission due to vehicle break-downs or the nature of the mission which may necessitate re-evaluation of the expected system performance.

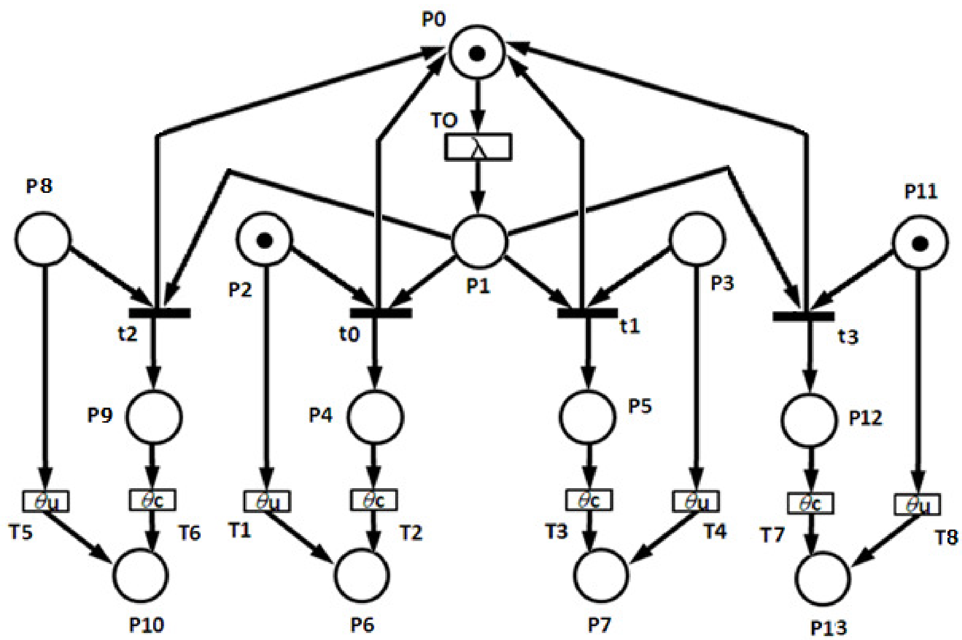


Figure 26 – Petri net model with 2 actives search-engage vehicles

In this thesis, it was demonstrated both through simulation and analytically that the time distribution of the cued search process follows a triangular distribution when the target is stationary and the cues provide the exact target location. Methods to approximate general distributions such as the triangular distribution with phase-type distributions are discussed and Petri Net models incorporating phase-type distributions are developed. Finally, a cooperative search system example from (Jeffcoat, Krokhmal and Zhupanska 2007) is modeled and analyzed to verify the proposed modeling methodology.

The contributions of this thesis can be summarized as follows:

- A novel Petri Net based modeling methodology for modeling cooperative search systems involving cueing is introduced.
- The cued search process for stationary targets is shown to follow a triangular distribution when the cue provides the exact target location. This process is

similar to traveling from one random location to another, namely from the location of the cued search-engage vehicle and the location of the stationary target.

7.2 Future Research Directions

Intelligent cueing is an immediate and natural future research direction to incorporate intelligent target assignment into the proposed Petri net model. The proposed Petri net model assumes that cues are assigned randomly among the vehicles available. However, the decision of what vehicle to cue could be based on several factors such as proximity to the target or elapsed uncued search time. In the case of a system with heterogeneous search-engage vehicles, the decision of what vehicle to cue would also depend on vehicle capabilities. Considering these factors may decrease the time to engage a target after receiving the cue, or may increase system effectiveness by selecting the vehicle(s) with appropriate capabilities for a particular mission.

Controlled Petri Nets are an extension of standard Petri nets in which binary control inputs can be applied as external conditions for enabling transitions in the net. The markings of the external input places can be used to restrict the firing policy on the Petri Net. In the proposed Petri Net model, the status of a search-engage vehicle can enable a transition to make it eligible to be cued.

The theory of fuzzy logic (Carlsson and Fullér 2002) resembles human reasoning in its use of imprecise information to generate decisions. Fuzzy logic does not need exact equations and precise numeric values, it allows expressing the states of the system with subjective concepts which are mapped into exact numeric ranges. Thus, fuzzy logic can be used to classify the status of the search-engage vehicles and consequently determine if eligible to be cued.

Another extension of this research involves modeling cooperative search systems with moving targets and imprecise cues. This extension would not impact the structure of the proposed Petri Net model, however it is anticipated that the cued search time would not follow the triangular distribution since there is still a search process that has to take place once the search-engage vehicle reaches the cued location since the target may have moved and/or an imprecise cue requires the search-engage vehicle to search for the exact location of the target.

References

Ajmone, Marsan, G Balbo, Giuseppe Conte, S Donatelli, and G. Franceschinis. *Modelling with Generalized Stochastic Petri Nets.* John Wiley and Sons, Inc, 1994.

Alexander, Alice M, and David E. Jeffcoat. "Markov Analysis of the Cueing Capability/Detection Rate Trade-space in Search and Rescue." *LECTURE NOTES IN ECONOMICS AND MATHEMATICAL SYSTEMS* (SPRINGER VERLAG KG), 2007: 185-196.

Allmaier, S, M Kowarschik, and G Horton. "State Space Construction and Steady--State Solution of GSPNs on a Shared--Memory Multiprocessor." *Seventh International Workshop on Petri Nets and Performance Models* . 1997. 112-122.

Asmussen, Soren, Olle Nerman, and Marita Olsson. "Fitting Phase-Type Distributions via the EM Algorithm." *Scandinavian Journal of Statistics*, 1996: 419-441.

Basu, Ayanendranath, Ian Harris, Nils Hjort, and M Jones. "Robust and efficient estimation by minimising a density power divergence." *Biometrika*, 1998: 549-559.

Bingham, Price T. Lt Col. "Transforming Warfare with Effects-Based Joint Operations." *Aerospace Power Journal*, 2001.

Bobbio, Andrea, and Miklos Telek. "A benchmark for ph estimation algorithms: results for acyclic-ph ." *Stochastic Models* , 1994: 661-677.

Cao, Y. Uny, Alex S Fukunaga, and Andrew Kahng. "Cooperative Mobile Robotics: Antecedents and Directions ." *Autonomous Robots*, 2004: 7-27.

Carlsson, Christer, and Robert Fullér. *Fuzzy Reasoning in Decision Making and Optimization.* New York: Heidelberg, 2002.

Chandler, Phillip r, and Meir Pachter. "Complexity in UAV Cooperative Control." *Proceedings of the American Control Conference.* Anchorage, 2002. 1831-1836.

Chen, Po-zung, Steven C Bruell, and Gianfranco Balbo. "Alternative Methods for Incorporating Non-exponential Distributions into Stochastic." *In the Proceedings of the Third international Workshop on Petri Nets and Performance Models.* Washington: IEEE Computer Society, 1989. 187-197.

Cole, David T, Thompson Paul, Ali Haydar Göktoğan, and Salah Sukkarieh. "Demonstrating the Benefits of Cooperation for a UAV Team Performing Vision Based Feature Localisation." In *Experimental Robotics*, 105-115. Heidelberg: Springer Berlin, 2009.

Committee on Autonomous Vehicles in Support of Naval Operations, National Research Council. *Autonomous Vehicles In Support Of Naval Operations.* National Academies Press , 2005.

Desrochers, Alan A, and R.Y. Al'Jaar. *Applications of Petri nets in Manufacturing Systems: Modelling, Control and Performance Analysis* . IEEE Press , 1995.

Gaboune, Brahim, Gilbert Laporte, Soumis, and Francis Soumis. "Expected Distances between Two Uniformly Distributed Random Points in Rectangles and Rectangular Parallelpipeds." *The Journal of the Operational Research Society* (Palgrave Macmillan) 44, no. 5 (1993): 513-519.

German, Reinhard. *Performance Analysis of Communication Systems : Modeling with Non-Markovian Stochastic Petri Nets.* New York: John Wiley & Sons, Inc, 2000.

Haulman, Daniel L. *U.S. Unmanned Aerial Vehicles in Combat, 1991-2003.* Research paper, Maxwell: Historical Research Agency, 2003.

Hsieh, Ani, et al. "Adaptive teams of autonomous aerial and ground robots for situational awareness." *Journal of Field Robotics* 24, no. 11-12 (2007): 991-1014.

Iida, Koji, Ryusuke Hohzaki, and Takesi Kaiho. "Optimal Investigating Search Maximizing the Detection Probability." *Journal of the Operations Research Society of Japan* 40, no. 3 (1997).

Inuiguchi, Masahiro, Shoji Hirano, and Shusaku Tsumoto. *Rough Set Theory and Granular Computing.* New York: Springer, 2003.

Jeffcoat, David. "Coupled Detection Rates: An Introduction." In *Theory and algorithms for Cooperative Systems*, edited by Don Grundel, Robert Murphey and Panos M Pardalos, 157-167. New Jersey: World Scientific Publishing, 2004.

Jeffcoat, David E, Pavlo A Krokhmal, and Olesya I. Zhupanska. "A Markov Chain Approach to Analysis of Cooperation in Multi-Agent Search Missions." In *Cooperative Systems: Control and Optimization*, 171-184. Heidelberg: Springer Berlin, 2007.

Kulkarni, Vidyadhar G. *Modeling, Analysis, Design, and Control of Stochastic Systems.* New York: Springer, 1999.

Lundell, Martin, Jingpeng Tang, and Kendall Nygard. "Fuzzy Petri net for UAV decision making." *Proceedings of the 2005 International Symposium on Collaborative Technologies and Systems.* 2005. 347-352.

Marios, Polycarpou, Yanli Yang, and Passino, Kevin Liu Yang. "Cooperative Control Design for Uninhabited Air Vehicles." In *Cooperative control: models, apllications and algotigh*, by Sergiy Butenko, Murphey Robert and Panos M Pardalos, 283-321. Kluwer Academic Publishers, 2003.

Miner, Andrew S. "Efficient solution of GSPNs using Canonical Matrix Diagrams." *Proceedings of the 9th international Workshop on Petri Nets and Performance Models* . Washington: IEEE Computer Society , 2001. 101 - 110.

Montgomery, Douglas C, and George C Runger. *Applied Statistics and Probability for Engineers.* John Wiley & Sons, Inc, 2002.

Oracle Corporation. *Unmanned Vehicle Systems: Leveraging COTS Software.* Redwood Shores: Oracle Corporation, 2007.

Palamara, Pier F, Vittorio A Ziparo, Luca Iocchi, Daniele Nardi, and Pedro Lima. "Teamwork Design Based on Petri Net Plans." *Lecture Notes In Artificial Intelligence* , 2009: 200-211.

Peters, J, A Skowron, Z Suraj, S Ramanna, and A Paryzek. "Modeling Real-Time Decision-Making Systems With Rough Fuzzy Petri Nets." 1998.

Richards, Marc D, Darrel Whitley, Beveridge Ross J, Todd Mytkowicz, Nguyen Duong, and Rome David. "Evolving cooperative strategies for UAV teams." *Genetic And Evolutionary Computation Conference* . Washington: ACM , 2005. 1721-1728.

Schultz, Alan C, Lynne E Parker, and Frank E Schneide. *Multi-Robot Systems: From Swarms to Intelligent Automata, Volume II.* Dordrecht: Springer, 2002.

Shaked, M, and J Shanthikumar. "Phase Type Distributions—I." *Encyclopedia of Statistical Sciences* , August 15, 2006: 709-715.

Stone, Lawrence D. "The Process of Search Planning: Current Approaches and Continuing Problems." *Operations Research* (INFORMS) 31, no. 2 (1983): 207-233.

Stone, Lawrence D. *Theory of Optimal Search, 2nd Ed.* Operations Research Society of America, 1992.

Technologies, MathWave. *MathWave Technologies data analysis & simulation.* http://www.mathwave.com/company.html (accessed June 15, 2010).

Uryasev, Stanislav, and Panos M Pardalos. *Stochastic optimization: algorithms and applications.* Vol. 54. Springer, 2001.

Van der Aalst, W, K Van Hee, and H Reijers. "Analysis of Discrete-time Stochastic Petri Nets." *Statistica Neerlandica* 54, no. 2 (2000): 237-255.

Verkama, Markku. "Optimal Paging - A Search Theory Approach." *5th IEEE Int. Conf. Universal Personal Communications.* 1996. 956-960.

Wang, Jiacun. *Timed Petri Nets: Theory and Application* . Boston: Kluwer Academic Publishers, 1998.

Yee, Shang-Tae, and Jose A Ventura. "Phase-Type Approximation of Stochastic Petri Nets for Analysis of Manufacturing Systems." *Robotics and Automation* 16, no. 3 (2000): 318-322.