Fall 2016

Improvement of Work Process Performance with Task Assignments and Mental Workload Balancing

Cansu Kandemir
Old Dominion University

IMPROVEMENT OF WORK PROCESS PERFORMANCE WITH TASK ASSIGNMENTS AND MENTAL WORKLOAD BALANCING

by

Cansu Kandemir
B.S. June 2010, Izmir University of Economics, Turkey
M.A. July 2012, Izmir University of Economics, Turkey

A Dissertation Submitted to the Faculty of
Old Dominion University in Partial Fulfillment of the
Requirements for the Degree of

DOCTOR OF PHILOSOPHY

ENGINEERING MANAGEMENT

OLD DOMINION UNIVERSITY
December 2016

Approved by:

Holly A. H. Handley (Director)

Resit Unal (Member)

Ghaith Rabadi (Member)

Roland Mielke (Member)

ABSTRACT

IMPROVEMENT OF WORK PROCESS PERFORMANCE WITH TASK ASSIGNMENTS AND MENTAL WORKLOAD BALANCING

Cansu Kandemir
Old Dominion University, 2016
Director: Dr. Holly A. H. Handley

The outcome of a work process depends heavily on which tasks assigned to which employees. However, sometimes-optimized assignments based on employees’ qualifications may result in an uneven and ineffective workload distribution among them. Likewise, an even workload distribution without considering the employee's qualifications may cause unproductive employee-task matching that results in low performance of employees. This trade-off is even more noticeable for work processes during critical time junctions, such as in military command centers and emergency rooms that require being fast and effective without making errors.

This study proposes that optimizing task-employee assignments according to their capabilities while also keeping them under a workload threshold, results in better performance for work processes, especially during critical time junctions. The goal is to select the employee-task assignments in order to minimize the average duration of a work process while keeping the employees under a workload threshold to prevent errors caused by overload. Due to uncertainties inherent in the problem related with the inter-arrival time of work orders, task durations and employees' instantaneous workload, a utilized simulation-optimization approach solves this problem. More specifically, a discrete event human performance simulation model evaluates the objective function of the problem coupled with a genetic algorithm based meta-heuristic optimization approach to search the solution space.

This approach proved to be useful in determining the right task-agent assignments by taking into consideration the employees' qualifications and mental workload in order to minimize the average duration of a work process. Use of a sample work process shows the effectiveness of the developed simulation-optimization approach. Numerical tests indicate that the proposed approach finds better solutions than common practices and other simulation-optimization methods. Accordingly, by using this method, organizations can increase performance, manage excess-level workloads, and generate higher satisfactory environments for employees, without modifying the structure of the process itself.

This dissertation is dedicated to my mom, Nuray, and my dad, Erdem.

ACKNOWLEDGMENTS

Many people have contributed to the successful completion of this dissertation. First and foremost, the untiring efforts of my advisor, Dr. Holly. Handley, deserve special appreciation. Not only has Dr. Handley shaped who I am as a scientist, but she also shaped my teaching, communication and collaboration skills. Without her guidance and persistent help this dissertation would not have been possible. I extend many, many thanks to committee members, Dr. Resit Unal, Dr. Ghaith Rabadi and Dr, Roland Mielke for their patience and guidance on my research and editing of this manuscript.

I especially would like to thank Dr. Resit Unal and Dr. Mujde Unal for their numerous areas of support and encouragement. They are family to me!

My very precious thanks to my mom Nuray Kandemir, my dad Erdem Kandemir. Whatever I do to thank them, it will not be a satisfactory acknowledgement. I am deeply grateful for their limitless moral support throughout my study. I feel very blessed to have such a wonderful family.

I am very thankful to all of my friends and colleagues in the Department of EMSE. I also owe a special note of appreciation to Elnaz Dario, Gulsah Hancerliogullari, Tunca Tabaklar, Nima Shahriari, Matthew Amissah, Mustafa Canan, and Kevin Muchiri. They were always there giving me feedbacks and helping me to stay in a good mood. Lastly, Erik Test, who has given me the biggest support during my doctoral study deserves the most special thanks. He was always there when I needed. There are no words that can express my appreciation to him.

NOMENCLATURE

ABOA Assignment based on availability

ABOC Assignment based on capability

ABOGA Assignment based on genetic algorithm

C3TRACE Command, Control, and Communication: Techniques for Reliable Assessment of Concept Execution

CR Cross-over Rate

GA Genetic Algorithm

GAP Generalized Assignment Problem

IMPRINT Improved Performance Research Integration Tool

IPME Integrated Performance Modeling Environment

MRT Multiple Resource Theory

MR Mutation Rate

OT Operational Tempo

PS Population Size

VACP Visual, Auditory, Cognitive, Psychomotor

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Figure Page

1. INTRODUCTION

The effects of globalization, increased competition, complex tasks, limited number of employees and time restrictions have transformed work processes from simple task sequences into more challenging networks that can benefit from continuous improvements. Impacted by these increased challenges, decisions on task assignments and considerations of employees' workload have become even more difficult while inaccuracies resulting from such decisions can have consequences on the productivity of the organization. These consequences include failed tasks, reduced efficiency, and inability to meet deadlines.

The problem of improving the performance of work processes in organizations has been handled in different ways (April, Better, Glover, Kelly, & Laguna, 2006). Introducing automation to the processes, providing education and training for the employees, improving the quality of the management of organization, and reforming the structure of the organization are some of the practices used. These practices may require significant changes, such as hiring new employees, and can be very costly for the organizations. However, by assigning the tasks to the most qualified available employee, increases the performance of an organization.

In general, work processes consist of different tasks, which require different expertise. Employees usually have various degrees of qualifications and their performance may vary for different tasks. Therefore, the outcome of the work process depends heavily on which tasks assigned to which employees (Kamrani, Ayani, & Moradi, 2012). However, sometimes-optimized assignments based on qualifications may result in overload conditions for highly qualified employees while the rest remain under-loaded or idle. On the contrary, even distribution of workload without taking into consideration the qualification of the employee may cause unproductive employee-task matching resulting in low performance of employees. Accordingly, there is a tradeoff between optimized assignment of employees according to their capabilities and balancing employee workload levels. This tradeoff is even more noticeable for work processes in critical time junctions, such as work processes in an emergency room or military command center.

This study proposes that optimization of task-employee assignment based on their qualifications while also keeping their workload under a threshold results in better performance. The aim is to propose a computational model to evaluate the potential improvements in outcomes of work processes in critical time junctions by optimizing task-employee assignments regarding their qualifications without overloading them. Critical time junctions are the time ranges when work orders come very frequently. Because of that, work processes intersecting and force employees to multi-task. By employing the proposed approached, organizations can manage excess levels of workload, increase employees' performance, and generate higher satisfactory environments for members of the organization, without modifying the structure of the process itself.

The research reported in this dissertation developed a computational model that assigns tasks to employees according to their capabilities while considering their mental workload level. Use of the model, evaluated the performance variations in the outcome of the work process. Here, as the workload measure, *mental workload* of the employee seems to be appropriate. Mental workload reflects how difficult it is for the brain to accomplish task demands. Humans have a limited capacity for processing resources allocated to task performance. They are capable of multi-tasking until task demands exceed available resources. In other words, mental workload is the perceived relationship between the amount of mental processing capability or resources and the amount required by the task. This is an important measurement because it provides awareness to incidents where increased task demands within work processes may lead to poor or unacceptable performance (Cain, 2007).

It is worth noting that the assignment problem and mental workload analyses are not new in the literature. A number of researchers have performed mental workload analysis within military and health-care environments that require problem recognition and diagnosis, formulation and implementation of plans of actions, prioritisation of plans of action, making prompt decisions based on the integration of experience and an understanding of current situations, and coping with unexpected situations. Commonly, simulation modeling through task network representation is being used for such analysis. In a task network model, performance of an individual can be analyzed by decomposing an assignment into a series of main tasks and then into series of sub-tasks. In Human Factors Engineering, this process is called *task analysis* and a task network is constructed by defining the sequence of the tasks (Dahn & Laughery, 1997). Furthermore, human performance modeling simulation modeling tools such as IMPRINT (Improved Performance Research Integration Tool) and IPME (The Integrated Performance Modeling Environment) that have the capability to include the effects of the employee's education, experience, or the condition of the workspace while analyzing their *mental workload level* have been used extensively. The literature review section outlines the summary and references to these studies.

It has generally been found that task assignment problems based on employee qualifications are solved with deterministic optimization (Carley, 2002; Cheng & Chu, 2012), while mental workload analyses are studied employing simulation modeling (Bierbaum, Fulford, Hamilton, & Fort, 1990; Mitchell, 2000). Deterministic optimization, which ignores uncertainty in order to come up with a unique and objective solution, relies on linear algebra and is fast in converging to a solution (Cavazzuti, 2013). However, the nature of a work process is stochastic. In real life, the chance that an employee performs the tasks every time exactly the same way is not very likely. Even the duration of a simple task may vary according to employee’s mood, workload, current working conditions, difficulty of the task etc. In situations where uncertainty is at the center of the problem, a different strategy is essential.

Given that simulation approximates reality, it also permits the inclusion of various sources of uncertainty and variability into tasks that affect work process outcomes. However, simulation generally answers "what-if" questions and it is not possible to find optimal solutions in reasonable time for the problems where the solution space increases exponentially as the number of independent variables increases.

The problem in this study is a variation of the assignment problem, with the solution space growing exponentially as the number of independent variables (the number of possible task and employee pairs) increases. Therefore, it is difficult and inefficient to try to evaluate every single task-and-employee pair with the workload outputs. Nevertheless, the stochastic nature of the organizational environment cannot be ignored.

The introduction of a two-step model helps to overcome this problem and provide a simulation environment with which to study task assignment and workload balance tradeoffs. The first step is the optimization tool to guide the search for the best configuration. The second step, the stochastic part, evaluates results of the configuration suggested from the first part. A Simulation Optimization approach can resolve problems related to the utilization of employees by merging optimization and simulation. Thus, the optimization algorithm and the evaluation function of the stochastic simulation method are integrated. Use of this approach will find the most beneficial task-employee assignments and insure that mental workload of the employees stays under the threshold. Additionally, the output of this two-step model will test hypotheses based on employee-task assignment with various parameters and the impact on performance of work process, such as timeliness of the process in the case where no employee is overloaded.

This study will contribute to the literature in several ways. First, the results of this study will help organizations address task assignments and employees' overload problems using a methodological approach. The right matching of employee-task based on qualifications is hypothesized to be as important as keeping the employees in appropriate workload limit. For instance, an employee could get highly loaded and as a result cannot perform well even with the best qualifications. Additionally, a merged simulation-optimization approach may help researchers gain insights into the effectiveness of alternative research designs. The novel areas of this work are:

- A Simulation Optimization (utilizing a discrete event human performance simulation-genetic algorithm) approach that reaches optimal/near optimal solution of employee-task assignments (by reorganizing human resources) in reasonable time in order to minimize average duration (timeliness) of the work processes in critical time junctions without overloading the employees and without making any major changes in the structure of the work process.
- An employee-task assignment tool that can handle large solution spaces (high number of employees and tasks).
- A simulation modeling framework that embraces the stochastic nature of work processes such as task durations, inter-arrival time of work orders, and most importantly, employees' instantaneous workload.

- A human performance simulation-modeling tool, which seamlessly integrates with other software. Current human performance simulation modeling tools (for commercial use) are only capable of getting input from the user such as IMPRINT, IPME that limit the analysis to only what-if analysis. The need of a human performance simulation-modeling tool that communicates with other software is satisfied.
- A flexible tool, which managers can use to evaluate different work processes with different task-agent sizes, capabilities, workload combinations and operating rules.

The performance of the developed simulation-optimization approach is tested through computational analysis. The results from the approach are compared to the results of common practices and other simulation-optimization approaches such as commercial simulation-optimization packages. The findings are discussed in detail. Computational results provide managerial insights as well as highlighting the importance of such a simulation-optimization tool for assignment problem for work processes at critical time junctions. Overall, the developed simulation optimization was found to be effective and efficient in finding solutions to the problems considered.

1.1 Problem Statement

The purpose of this study is to propose a simulation optimization approach to find the task-employee pairs that improves performance of work processes in critical time junctions by regarding employees' capabilities while keeping their workload under the upper workload limit. The combination of several tasks belonging to a process defines a work process. For instance, a software development process consists of main tasks such as coding the core program, planning validation and verification, and developing the interface. Then, each main task decomposes to several subtasks. Each task requires different capabilities at different levels, which mean employees, must meet the necessary qualifications of each task.

The critical aim of this dissertation is to provide a method to improve the outcome of the work process (such as timeliness) by reorganizing the existing human resources without making any major changes in the structure of the work process (such as hiring new employees or changing the task structures).

It is important to mention that measuring work process performance is a challenging task and there is no single universal method. There may be various performance measures for different work processes. For this study, performance measurement of the modeled organization and work process is related to the duration of the process. Since being as fast as possible is such an important factor of success for *tasks processes in critical time junctions* (i.e. emergency rooms processes that represent life or death situations and require fast decision making), a performance measure representing the timeliness of the process is an appropriate one. As a result, finding the task-employee combinations that minimize the average duration of the work process is chosen to be the objective of this study. Another reason is that both capability and skill level of employee can affect the work process duration. For instance, for an employee that has a lack of experience on a task, the time it would take him to finish the task would be longer. In addition, the employee cannot start a new, parallel task unless he/she has enough residual capacity (in other words, he/she is not going to be overloaded). Otherwise, the employee would be prone to errors or inefficiencies. Hence, not being able to start to a new task would increases the total duration of the process. Consequently, timeliness is an indicator of a successful work process in a critical time junction.

Several components of this research are worth noting. First, the following entities will be evaluated to assist in characterizing the work processes:

- The number of available employees (set of agents) to assign the tasks;
- The agents' capabilities and capability levels (depending on the problem scenario: education/training, qualification or experience level);
- The number and type of tasks belonging to the work process, with a required level of capabilities;
- Mental workload demand of each task and mental workload threshold of the agents; and
- The time that the agent takes to finish a task, this is a function of his capabilities and the capability requirements of the task.

For example, if an agent’s experience level for a specific task is low, then the completion time for that specific task that requires higher experience level will be longer. Second, the independent variables for this study will be the task-agent assignments. Lastly, the dependent variable will be the timeliness of the work process. The average duration to finish a work process will be used to determine the timeliness. A brief summary of the parameters is provided in Table 1.

Table 1. Summary of the Parameters

Parameters		Brief explanation
Constants	A work process	A fixed ordered sequence of tasks with defined time duration, mental workload demands and required capability types and levels.
	A team	Number of agents with capability types and levels

Table 1. Continued

Independent Variables	Task-Agent Assignment	Each task-agent pair
Constraints	Instantaneous Mental Workload Threshold	Agents should stay under the defined mental workload level (mental workload threshold). An agent cannot start to an additional parallel task if it is going to increase his mental workload above the mental workload threshold level.
	Agent-Task Assignment Constraint	An agent can work on no, one or more task and a task can only be assigned to one agent.
Dependent Variables	Timeliness	The aim is to minimize the average duration of the work process by finding the right task-agent assignments.

This study is applicable to a given number of agents and a task flow. In order to describe the work process, it should be modeled at a low level of abstraction. The characteristics of the agents, such as their qualification level, have a critical effect on their performance. For instance, for a software development task that requires specific programming language skills, the employee that has the required capability level in that language needs to be chosen. Capabilities can be ranked by using a ranking scale (such as 1 to 5) where low ranks (such as 1) indicates lacking of the capability and high ranks (such as 5) indicates that the employee is at an advanced level in terms of this capability. In the case of experience, these ranks can be measured by number of years in the organization or in that specific process. In other cases, the ranks may be subjective values assigned by the decision maker.

Mental workload level is an essential indicator for work processes in critical time junctions. These include the operation of safety critical systems requiring active and passive vigilance tasks, problem recognition and diagnosis, formulation and implementation of plans of actions, prioritisation of plans of action, remembering to do things, making prompt decisions based on the integration of experience and an understanding of current situations, and coping with unexpected events. It is the perceived relationship between the amount of mental processing capability or resources and the amount required by the task. In other words, mental workload reflects how difficult it is for the brain to accomplish task demands. This is an important measurement because it provides awareness as to where increased task demands within user operations may lead to poor or unacceptable performance (Cain, 2007). People are capable of multi-tasking until task demands exceed available resources. The human mind either can devote to task demands individually or collectively through several resources: visual, auditory, cognitive, and psychomotor. Therefore, the resources concept is based upon the assumption that human operators have a limited capacity for processing resources that may be allocated to task performance (Wickens, 2008).

There are several mental workload measurement methods in the literature. For this study, Visual Auditory Cognitive Psychomotor (VACP) method will be used. According to this method, all tasks decomposes into different processing resources: visual, cognitive, auditory, and psychomotor (Mitchell, 2000). A specific scale represents the workload value for each processing resources. Note that this scale is task based and is not sensitive to human's personal characteristics. The literature review section gives further information on VACP, comparison between different mental workload measures, and reasons for selecting this method.

The simulation optimization process developed for this study will consist of two iterative and repeated sub-processes. The first sub-process will utilize a search procedure to guide and find satisfactory solutions of the agent-task assignment in order to minimize the average duration.

The assignment problem, in general, aims to determine the best assignment of tasks to agents according to a predefined objective function and constraints (Kamrani et al., 2012). Metaheuristic search algorithms have the capability to guide the search to near optimal or optimal solutions (Better et al., 2008). Genetic Algorithm (GA), which is a metaheuristic method, (Holland, 1975), appears to be the right search procedure for this problem. The second sub-process will utilize a simulation model to obtain the performance result (average work process duration) of the suggested solution from the optimization engine. In conducting simulation optimization, the output of each sub-process will be used as an input for the other. This iteration will continue until the stopping criteria are satisfied.

The model created for this study will serve as a test-bed to evaluate different hypotheses on the method to assign agents to tasks based on capabilities, while still maintaining a workload balance among them in order to prevent errors and preserve or improve the level of performance. Data from an example work process will populate the model. The outcome of the virtual experiments will provide guidance on the tradeoffs between task assignment and workload balance and identify the region where both goals successfully meet. The outcome from these results will provide input to both organizational design and engineering management fields.

Some examples of work processes in critical time junctions that this methodology can be applied to are summarized as follows:

- A team of pharmacists working on medication reconciliation and order verification (Metzger, Chesson, & Momary, 2015). They have to accomplished the tasks such as:
    - identify any issues that need to be solves for medication reconciliation,
    - classify medications by disease state,

  - review the verified medications and assess each new order for accuracy, appropriateness, and safety;

  in addition to numerous other job functions. Several studies show that as the orders increases, the likelihood of the pharmacists to make an error increases (Reilley, Grasha, & Schaffer, 2002). Moreover, their error rate and the time they spend on a patient found relates with their education and experience level (Gorbach et al., 2015).

- Supervisory controllers in the operations of unmanned aerial vehicles (UAVs). In this work process, human operators monitor a system and intermittently interact with a computer interface to transform operator commands to detailed control actions on the system (Sheridan, 2012; Cummings & Guerlain, 2007). One critical aspect of the UAV pilot's task requirements is the ability to manage multiple modes of communication. Pilots control the vehicle through radio and satellite communications. In addition to these demanding communication tasks, they a host of other tasks to accomplish including vehicle routing, which involves creating emergency and operational inputs, sensor manipulation to evaluate weather, and vehicle system checks.
- General practitioners in an emergency room with too many patients to see in a short space of time. Number of patients, level of training and experience of physicians has found to have an important effect on patient waiting time (Levin et al., 2007).
- A submarine team that is asked to take on the challenge of incorporating unmanned aerial systems (UAS) as a sensor in support of their current mission (Cook, Heacox, Averett, & Handley, 2012; Smallman, Cook, Beer, & Lacson, 2009). Generally, there are no explicit defined assignments for the submarine team during UAS launch and

flight control. The team member has different roles, qualifications, and availability. Incorporating the new tasks to their existing schedule is a challenge with the time constraints.

The remainder of this dissertation is structured as follows. In Chapter 2, the literature review discusses the assignment problem in organizational design and mental workload analysis for work processes and simulation optimization methodology. In Chapter 3 includes a description of the simulation optimization methodology (task-agent assignments for work processes in critical time junctions). Chapter 4 discusses the use of this method to create a case study model for a hypothetical “Air Interdiction Planning” mission. Chapter 5 includes an analysis of the results under various operational tempos by comparing the developed simulation optimization method to common practices and other simulation optimization methods from the case study. Finally, in Chapter 6 provides concluding remarks and possible areas for further work.

2. LITERATURE REVIEW

In order to approach the problem of assigning tasks to agents for work processes in critical time junctions in an efficient way, the literature was reviewed in three parts. First, the assignment problem in work process design, and second, mental workload analysis for work processes, was reviewed with their applications. Lastly, simulation optimization techniques were reviewed as the intended method to be applied the problem under consideration.

2.1. Assignment Problem in Work Process Design

In the field of operations research, correct assignment of tasks to employees based on evaluation of their suitability and resource constraints is known as the "assignment problem". The assignment problem and different variants of it have been discussed for more than 55 years. This section focuses on applications and the solution methods of assignment problems in organizations.

As stated in the definition of the assignment problem, the aim is to assign agents to tasks based on evaluation of their suitability (Kamrani, 2012). A literature review focused on the assignment problem shows that skill or capability level is a commonly used way to measure suitability. The measure of skill level generally changes according to the type of organization. For instance, in a military organization, the skill level relates to the rank of the military employees, while for a nurse working in an emergency room, skill level relates with experience and the ability to cope with stress. Some studies use ranking scales for skill levels and ask subject matter experts to rate them. In addition, there are studies that use experience (such as time spent utilizing a specific skill) as the capability level.

Minxin, Gwo-Hshiung, and Liu (2003) proposed a multi-criteria assessment model capable of evaluating the suitability of individual employees for a specified task according to their capabilities, social relationships, and existing tasks. Candidates are ranked based on their suitability scores to support workflow administrators in selecting appropriate employees to perform the tasks assigned to a given role. The proposed assessment model overcomes the lack of role-based task assignment in current workflow management systems.

Similarly, Eiselt and Marianov (2008) developed a mathematical model for the assignment of tasks to individual employees with different capabilities. They defined a skill space where an employee's position represents the level acquired in each skill. Tasks can also be mapped into the skill space. Once feasible task assignments are determined, tasks are assigned to employees. The objectives are to minimize inequity between the individual employees' workload and minimize employee-task skill differences to avoid boredom and costs. Both Eiselt and Marianov (2008) and Minxin et al. (2003) measure workload as the total number of hours that the employee works.

Otero, Otero, Weissberger, and Qureshi (2010) claim that completing reliable software products within the expected time frame is a major problem for companies that develop software applications, the reason attributed to inadequate resource allocation. Consequently, they state that it is beneficial to generate systematic employee assignment processes that consider the complete candidate skill set and provide the best fit in order to increase quality, reduce cost, and reduce training time. Moreover, Tsai, Moskowitz, and Lee (2003) argue that software development projects are often unsuccessful because of inadequate human resource project planning. A major contributor to this problem is the inefficient allocation of resources that may result in schedule overruns, decreased customer satisfaction, decreased employee morale, reduced product quality, and negative market reputation. The inevitable consequence is a decrease in potential profit for companies. Accordingly, Otero et al. (2010) proposed a multi-criteria decision making approach for allocating resources to software engineering task assignment. They used a Desirability Function developed by Derringer and Suich (1980) to provide a unified metric representative of the suitability between the complete set of skills available from employees and skills required for tasks to assign quantitatively resources to tasks even when the most desirable skills are not available from the existing workforce. They took into consideration project specific capabilities, such as years of experience, level of perceived expertise on a particular language, operating system, domain knowledge, etc.

In the case where optimum skill sets are not available, Otero, Centeno, Ruiz-Torres, and Otero (2009) developed a linear programming assignment model to match resources to tasks that considers existing capabilities of employees, required levels of expertise, and priorities of required skills by the task. Also, Acuna and Juristo (2004) and Acuna, Juristo, and Moreno (2006) developed procedures for assigning employees to software tasks according to the assessment of behavioral competencies. Tsai et al. (2003) proposed the critical resource diagram (CRD) method and the Taguchi’s parameter design approach for the selection of employees. The CRD method used resource scheduling to represent human-resource workflow and tasks’ precedence. The Taguchi’s parameter design approach obtained a scheme that would optimize the selection of engineers for tasks under dynamic and stochastic conditions.

Kamrani et al. (2012) considered the tasks to be part of a business process model interconnected according to defined rules and constraints (a more complex form of assignment problem). Business process modeling refers to "describing business processes at a high abstraction level, by means of a formal notation to represent activities and their causal and temporal relationships, as well as specific business rules that process executions have to comply with" (Kamrani, Rassul, & Karimson, 2010, p.1). Business process modeling focuses on the representation of the execution order of activities. They used two main categories of business processes: assignment-independent and assignment-dependent. In the first category, different assignments of tasks to employees do not affect the flow of the business process. In the second category, processes contain critical tasks that may change the workflow, depending on who performs them. Combination of the Hungarian Algorithm with either the analytical method or simulation to provides an optimal solution. They conducted a series of tests, which showed that the proposed algorithms efficiently found optimal solutions for assignment-independent and near-optimal solutions for assignment-dependent processes.

In the last two decades, several papers have appeared in the literature where the use of the Multi-Resource Generalized Assignment Problem (MRGAP) solved employee allocation problems (Alidaee, Gao, & Wang, 2010). In these problems, the number of variables grew exponentially. In their research, they consider a generalization of MRGAP and show the improvement upon several published models based on MRGAP where the number of variables were exponentially large. They used computational experiments to demonstrate the advantages of the new model over existing ones.

A summary of the methods and their applications can be found in Table 2. The checklist of methods, considerations and applications of these studies are shown in Table 3. All of the approaches, mentioned in this section, aim to assign tasks to limited resources (employees) in an efficient way. In general, task assignment problems for business environments are solved with deterministic optimization where the uncertainty is ignored in order to come up with a unique and objective solution. However, the nature of an organization that embraces work processes is stochastic. Moreover, not every approach mentioned here take into consideration the workload of the employee. The studies that take the workload of employee into consideration measured it as the hours that the employee works. Since this study investigates high tempo work processes that forces employees to multi-task, a more sensitive workload measurement method is necessary. As a result, the next section focuses on mental workload analysis.

Table 2. Summary of Task-Employee Assignment Methods

Approach-Method	Description	Application	Reference
Multi-Criteria Optimization	Evaluates the suitability of individual employees for a specified task according to their capabilities, social relationships, and existing tasks. Candidates are ranked based on their suitability scores.	Uses a simulated example to illustrate the application of the proposed assessment model with 5 employees, 7 skills, and 5 tasks.	Minxin, Gwo-Hshiung, & Liu (2003)
Mixed Integer, Non-Linear Mathematical Model	Aims to assign tasks to individual employees with different capabilities. The objectives are to minimize inequity between the individual employees' workload, minimize employee-task distances to avoid boredom and costs.	The approach has applied in DICTUC S.A., a company owned by the Pontificia Universidad Católica de Chile, to a subset of 15 employees, 14 skills, and 22 (recurring) tasks.	Eiselt & Marianov (2008)

Table 2. Continued

Multi-Criteria Decision Making	Allocates resources to software engineering task assignment. They used a Desirability Functions They took into consideration project specific capabilities, such as years of experience, level of perceived expertise on a particular language, etc.	The case study assumes a scenario where 10 candidates are available. The identified required skill set involves 5 skills.	Otero, Otero, Weissberger, & Qureshi (2010)
Linear Programming Assignment Model	Matches resources to tasks that consider existing capabilities of employees, required levels of expertise, and priorities of required skills the task.	A sample scenario is used. Survey analysis was conducted to test its validity.	Otero, Centeno, Ruiz-Torres, & Otero (2009)
Critical Resource Diagram (CRD) and Taguchi's Parameter Design	Develops a model to use for the selection of employees. The CRD was used for resource scheduling to represent human-resource workflow and tasks' precedence. The Taguchi's parameter design was used to obtain a scheme that would optimize the selection of engineers for tasks under dynamic and stochastic conditions.	They used a scenario that contains 3 jobs; each job has 2 possible candidates.	Tsai et al. (2003)
A conceptual model/ procedure	Assigns employees to software tasks according to the assessment of behavioral competencies	They used statistical tests for validation.	Acuna & Juristo (2004) and Acuna, Juristo, & Moreno (2006)

Table 2. Continued

Hungarian algorithm combined with either the analytical method or simulation	They used two main categories of business processes, assignment-independent and assignment-dependent. In the first category, different assignments of tasks to employees do not affect the flow of the business process. In the second category, processes contain critical tasks that may change the workflow, depending on who performs them.	They used a model inspired by a work process of military staff. They conducted a series of tests, which shows that the proposed algorithms efficiently find optimal solutions for assignment-independent and near-optimal solutions for assignment-dependent processes.	Kamrani, Ayani, and Moradi (2012), Kamrani, Rassul, & Karimson, (2010)
Multi-resource generalized assignment problem (MRGAP)	Proposes a compact generalized assignment problem model that can be used to solve employee allocation problems.	They used computational experiments to demonstrate the advantages of the new model over existing ones.	Alidaee, Gao, & Wang, (2010)

Table 3. Checklist of methods, considerations and applications in assignment problem for work process design

Reference	Method			Consider		Application	
	Mathematical Modeling (Deterministic)	Heuristic Methods	Simulation Modelling	Level of capability of employees	Workload	Real Life Application	Sample / Artificial Scenario
Minxin, Gwo-Hshiung, & Liu (2003)	x			x	x	x	
Tsai et al. (2003)	x	x					x
Eiselt & Marianov (2008)	x			x	x	x	
Otero, Centeno, Ruiz-Torres, & Otero (2009)	x			x			x
Otero, Otero, Weissberger, & Qureshi (2010)	x			x			x
Kamrani, Rassul, & Karimson, (2010)	x	x	x	x			x
Kamrani, Ayani, & Moradi (2012)	x	x	x	x			x
Alidaee, Gao, & Wang, (2010)	x						x
This Study	**x**	**x**	**x**	**x**	**x**		**x**

2.2. Mental Workload Analysis for Work Processes

As discussed in the previous section, the total number of hours that an employee works generally is used as a measure for workload. Even though it is very meaningful for business work processes (such as software development), another type of measure is needed for the work processes in critical time junctions and those requiring multi-tasking (such as intelligence analyst work processes and emergency room tasks). In that case, Hart and Staveland (1988) defines mental workload measures as the supposed relationship between the amounts of mental processing capability or resources and the amount required by the task are appropriate measures.

The main objective of measuring workload is to quantify mental cost of performing tasks in order to predict operator and system performance (Cain, 2007). Wickens (1992) states that *"... performance is not all that matters in the design of a good system. It is just as important to consider what demand a task imposes on the operator's limited resources*" (p. 390). As task difficulty increases, performance usually decreases, response times and errors increase, control variability increases, fewer tasks are completed per unit time, and task performance strategies change (Huey & Wickens, 1993); there is less residual capacity remains to deal with other issues.

There are three different measurements techniques of mental workload. These are psychophysiological, subjective, and performance measurement techniques (Miller, 2001). Psychophysiological measurement of workload is a concept based on evidence that increased mental demands lead to increased physical responses from the body. Psychophysiological workload measures rely on continuous measurement of the physical responses of the body using sensors. Subjective measurement is based on the use of rankings or scales to measure the amount of workload a person is feeling. Subjective workload measures rely on the question-answer type response to varying levels of workload. Performance measurement of workload relies on examining the capacity of an individual by means of a primary or secondary task. An estimate of mental workload can be determined by measuring how well a person performs on the task, or how their performance worsens as workload increases. The summary table of the mental workload measurement techniques can be found in Figure 1.

Criteria to select the appropriate mental workload measurement technique for the study follow O'Donnell and Eggemeier (1986) who suggests that:

- The method must be consistently sensitive to changes in task difficulty or resource demand and distinguish between significant variations in workload.
- The method should be diagnostic, indicating the source of workload variation and quantify contribution by the type or resource demand.
- The method should not be intrusive or interfere with performance of the operator's tasks, becoming a significant source of workload itself.
- The method should be acceptable to the subjects, having face validity without being onerous.
- The method should require minimal equipment that might impair the subject's performance.

Cain (2007) adds that:

- The method should be timely and sufficiently rapid to apply to capture transient workload changes.
- The method should be reliable, showing repeatability with small variance compared with main effects.

- The method should be selectively sensitive to differences in capacity demand and not to changes unrelated to mental workload.
- Measurement techniques should be designed to capture the individual differences and reflect them in the values obtained from a sound theoretical framework.

Moreover, Casali and Wierwille (1983) claim that;

- The method should be insensitive to other task demands, such as physical activity beyond the conduct of the tasks.

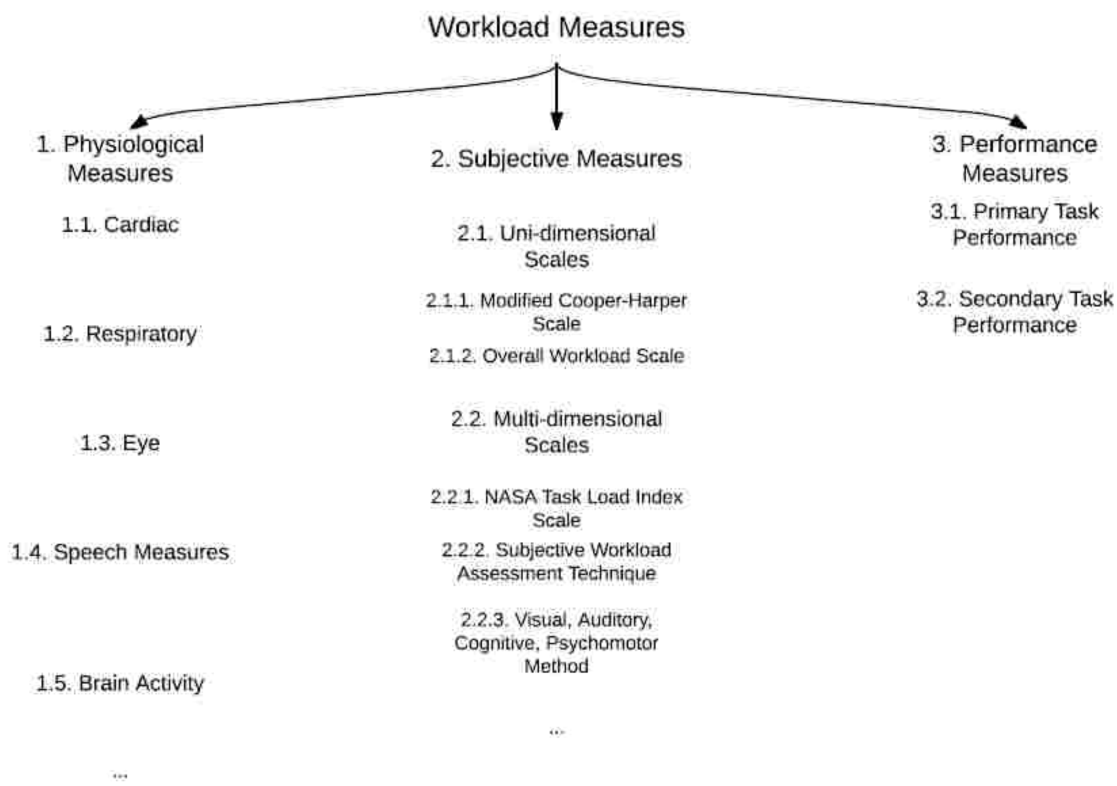


Figure 1. Summary figure of workload measures (Miller, 2001)

This literature review will focus on subjective workload measurement techniques, as it appears suitable that mental workload can be measured by subjective means. Subjective measures have a long history. They are popular since they are reliable and transferable to new systems or new task conditions. They also have high face validity. Moreover, for subjective workload measures there is no need for costly, time-consuming laboratory set-ups. Furthermore, Xie and Salvendy (2000) state that the most progress has been made in subjective measures. They also added that the analytical subjective models are the most attractive since they can be applied early in system. In general, input on workload for these models is gathered from subject matter experts (SMEs). To sum up, subjective measures are considered to be more practical, easiest, more flexible and most convenient form of evaluating workload (Yeh & Wickens, 1988).

There are a number of different methods for subjective measures such as rating scales, questionnaires, or interviews, that system designer can use to collect subjective data of workload. Hart and Wickens (1990) subdivide rating scale methods into unidimensional and multi-dimensional ratings. Unidimensional ratings are easy to understand and use but considered too simple to measure the complexity of workload. They lack combining ratings for predicting workload in different situations involving similar tasks. While unidimensional measures are more sensitive, multi-dimensional measures are more diagnostic. Moreover, most of the multi-dimensional scales have a predictive capability through constructive modeling.

The scales typically used to obtain multi-dimensional subjective ratings of workload are the subjective workload assessment technique (SWAT) (Reid & Colle, (1988); Reid, Potter, & Bressler (1989)); the National Aeronautics and Space Administration (NASA) task load index (TLX) (Hart & Staveland, 1988); and the visual, auditory, cognitive and psychomotor (VACP) model (McCracken & Aldrich, 1984). Description of each rating is shown in Table 3. According to Wickens (2002), they are the most sensitive, most transferable, and the least intrusive techniques for workload estimation. For instance, SWAT and NASA TLX, can provide appropriate workload indications when a mock-up of the proposed system exists. On the other hand, the analytical techniques can be used to predict mental workload when no mock-up exists and the system is just a concept. The greatest value of such measures is to ensure that task demands can remain within the residual capacity region (Wickens, 2008).

Table 4. Brief Explanation of Subjective Workload Measurement Scales

Subjective workload measurement scales	Reference	Brief explanation
Subjective workload assessment technique (SWAT)	Reid et al. (1989)	Uses three levels (low, medium, and high) for each of the three dimensions of time load, mental load, and physiological stress load to assess workload. The three steps that used to analyze workload: 1. Scale development, 2. Rate the workload. 3. Convert the scores into a 0 to 100 scale using the scale developed in step one.
National Aeronautics and Space Administration (NASA) task load index (TLX)	Hart & Staveland (1988)	Uses six dimensions to assess workload: mental demand, physical demand, temporal demand, performance, effort, and frustration. The workload scale is obtained for each task by multiplying the weight by the individual dimension scale score, summing across scales, and dividing by the total weights.
Visual, auditory, cognitive and psychomotor (VACP) model	McCracken & Aldrich (1984)	Any task performed by a person can be broken down into these components. Rating scales provide a relative rating of the degree to which each resource component is used. The steps are: 1. Identify tasks that are necessary to operate the proposed system. 2. Identify the operators to system. 3. Assign tasks to operators. 4. Estimate workload values using the scales.

Subjective workload measures that support predictive modeling, such as VACP, usually focus on task demand in multiple channels. When coupled with task duration in simulations, these approaches produce aggregate measures that are sensitive to both task difficulty and time. When combined with task analysis, simulation models give the best results (Wickens, 2002). Some simulation software, such as IMPRINT, do have a mental workload component (i.e. VACP scale), with task competition based on multiple resource theory (developed by Wickens (2002)) and with workload channels defined to correspond to the different dimensions in multiple resource theory. With the help of these simulation models, the system designer can predict task and procedure execution and mental workload. These models contain the tasks needed to accomplish a particular process, the amount of time it takes each task to perform in the process, the sequence of the tasks, and the person who performs each task. Nevertheless, the time and effort needed for inputs (e.g. tasks, operators, time, and resources) are high. In addition, validation of the simulation model is a major issue.

In general, the application of mental workload analysis is seen in military and health-care environments for critical processes that require immediate attention and decision-making.

For instance, Carayon and Gürses (2005) proposed a conceptual framework of intensive care units nursing workload that defines causes, consequences and outcomes of workload. They identified four levels of nursing workload: unit level, job level, patient level, and situation level and discuss measures associated with each of the four levels. Holden et al. (2011) states that reviews of nursing workload measurement show that workload is defined most often in terms of staffing ratios, and added these ratios are not clearly representative of the nurses’ actual or perceived workload. Both Carayon and Gürses (2005) and Holden et al. (2011) concluded suggesting using situation level (subjective) workload measures, since errors may be best described by task level workload. Moreover, they are reliable and transferable to new system or new task conditions.

Lamoureux (1999) and Dixon, Wickens and Chang (2005) used a simulated laboratory setting in order to measure the workload of air traffic controllers and UAV operators, respectively. Mitchell (2009) used mental workload analysis to evaluate changes in a combat system by IMPRINT. She claims that when the program managers add new technologies, these technologies have the potential to change the Soldiers' tasks. The tasks soldiers perform determine the soldiers' workload level and their performance. Too little or excess workload decreases their performance. The design goal for optimum soldier performance is to have an evenly distributed, manageable workload. To meet this design goal, they evaluated the impacts of new technologies on soldier tasks, workload and performance. Mitchell, Samms, Henthorn, and Wojciechowski (2003) examined the mental workload to determine best allocation of some combat functions among two versus three soldier crews. Another application, described by Samms and Mitchell (2010), evaluates the workload of tank crewmembers. They also mentioned the importance of defining a workload threshold level in mental workload analysis.

Mitchell (2009); Mitchell and Brennan (2009); Hunn, John, Cahir, and Finch (2008); Colombi, Miller, Scheiner, McGrogan, Long, and Plaga (2012); and Wong, Walters, and Fairey (2010) employed IMPRINT in their research. Plott, Quesada, Kilduff, Swoboda, and Allendar (2004) used popular human performance simulation software called C3TRACE, which is the abbreviation of Command, Control and Communication Techniques for Reliable Assessment of Concept Execution (Kilduff, Swoboda, & Barnette, 2002) in their study and discussed the theories behind the tool.

A summary of the approaches in mental workload analysis is found in Table 4. A checklist for the methods used in these applications is found in Table 5.

Table 5. Summary of Metal Workload Analysis Methods and Applications

Method	Description	Application	Reference
Conceptual Framework	They identified four levels of nursing workload: unit level, job level, patient level, and situation level and discuss measures associated with each of the four levels.	None included.	Carayon & Gürses (2005)
Survey	A study carried out at six nursing units at two pediatric hospitals provided interesting possibilities for how different types of workload may relate to common patient and employee problems in pediatric clinical settings.	To test this model, they analyzed results from a cross-sectional survey of a volunteer sample of nurses in six units of two academic tertiary care pediatric hospitals.	Holden et al. (2011)
Multiple resource theory-simulation modeling	They evaluated the impacts of new technologies on Soldier tasks, workload and performance	They used IMPRINT as the simulation software. The model has a crew of four Soldiers operating the system (Abrams V2 SEP).	Mitchell (2009)

Table 5. Continued

Multiple resource theory- VACP - simulation modeling	They predicted the mental workload associated with the infantry rifle squad using the common controller to control a small unmanned ground vehicle within an infantry mission.	They used IMPRINT as the simulation software. They proposed mitigation policies for the potential high workload situations.	Mitchell & Brennan (2009)
VACP	They examined the mental workload to determine best allocation of some combat functions among two versus three soldier crews.	The objective of this trade study was to examine the mental workload of the crew to determine the best allocation of the combat functions among two- and three-soldier crews.	Mitchell, Samms, Henthorn, & Wojciechowski (2003)
Multiple resource theory- VACP - simulation modeling	They evaluated the streaming video analysis portion of the geospatial intelligence process associated with an unmanned aircraft system, which provides information to a four person, military intelligence, geospatial analysis cell.	They used IMPRINT as the simulation software. Recommendations are made regarding the level of staffing for this type of system, based on crew workload characteristics discovered.	Hunn, John, Cahir, & Finch (2008)
Multiple resource theory- simulation modeling	They propose a technique that can be applied in any workload analysis.	They applied the technique on a case with tank crewmembers.	Samms & Mitchell (2010)
Multiple resource theory- VACP - simulation modeling	Using system architecture as the foundation, they explored the use of MRT to create representative workload models for evaluating operational system of systems concepts.	They used IMPRINT as the simulation software. An example involving a single pilot controlling multiple remotely piloted aircraft is presented.	Colombi, Miller, Schneider, McGrogan, Long & Plaga (2012)

Table 5. Continued

Multiple Resource theory- Simulation Modeling	They discussed the theory and application of C3TRACE tool by developing two conceptual models.	They used C3TRACE as the simulation software. In support of U.S. Army's premier acquisition program, a baseline and alternate configurations of the Unit of Action Mounted Combat System Company Headquarters are represented and evaluated.	Plott, Quesada, Kilduff, Swoboda, & Allender, (2004)
Simulated Laboratory Setting	They suggest that automation can help alleviate task interference and reduce workload, thereby allowing pilots to better handle concurrent tasks during single- and multiple-UAV flight control.	36 licensed pilots flew both single-UAV and dual-UAV simulated military missions. Pilots were required to navigate each UAV through a series of mission legs in one of the following three conditions: a baseline condition, an auditory autoalert condition, and an autopilot condition.	Dixon, Wickens, & Chang (2005)
Simulated Laboratory Setting	The study outlines an investigation of the impact of aircraft proximity and relationship data on the subjective mental workload of air traffic controllers.	3 participants are used. Study shows that is it possible to quantify the relationship between aircraft relationships and mental workload and eliminate much of what was previously considered to be subjective variation.	Lamoureux (1999)
Multiple Resource theory- TLX – VACP- Simulation Modeling	Discrete Event Simulation (DES) is used as the design method for crew performance of the NASA's Orion Crew Vehicle (CEV).	The results revealed that a majority of the DES model was a reasonable representation of the current CEV design.	Wong, Walters, & Fairey (2010)

In summary, subjective methods are the most used methods in human engineering evaluation to evaluate the employee's rating of a task. These methods, especially those with rating scales, have various advantages in measuring workload relative to other approaches. They have good face validity and general applicability. The VACP method is the most preferred one because it is based on Multiple Resource Theory, developed by Wickens (2002). In multiple resource theory, individuals are viewed as having several different capacities of resources, these resources are differentiated according to information processing stages (encoding and central processing or responding), perceptual modality (auditory or visual) and processing codes (spatial or verbal) (Wickens, 2002). VACP's workload predictions are task-based predictions, and it is applicable through discrete event simulation (Keller, 2002). Furthermore, it can be used in system design early in the concept phase when design changes are less expensive and, therefore, more likely to be implemented (Mitchell, 2009). While the most popular commercial human performance simulation software seems to be IMPRINT (Mitchell & Samms, 2009), alternate commercial workload modelling tools are also available. Those tools include, but not limited to, the Man-Machine Integration Design and Analysis System (Stanton, Salmon, Walker, Baber, & Jenkins, 2006), the Queuing Network-Model Human Processor (Boles & Adair, 2001), Integrated Performance Modelling Environment (Law & Kelton, 1999), Command, Control, and Communications Modelling Environment (C3TRACE) (Kilduff, Swoboda, & Barnette, 2002), and the Integrated Performance Modelling Environment (IPME) (Dahn & Laughery, 1997).

Table 6. Summary of Methods, Considerations and Applications in Mental Workload Studies

	Method/Tool					Consider	Application	
Reference	Ranking Scales-Methods	Survey	Conceptual Framework	Simulator	Simulation Modeling	Multiple Resource Theory	Real Life Application (real life data)	Application Info
Carayon & Gürses (2005)		x	x					-
Holden et al. (2011)		x					x	Define the workload of nurses in six units of two academic tertiary care pediatric hospitals
Mitchell, Samms, Henthorn, & Wojciechowski (2003)	x				x	x	x	Evaluate crew of four soldiers operating the system (Abrams V2 SEP)
Mitchell (2009)	x				x	x	x	Determine the best allocation of the combat functions among two- and three-soldier crews
Samms & Mitchell (2010)	x				x	x	x	A case with tank crewmembers
Mitchell & Brennan, (2009)	x				x	x	x	A case for infantry rifle squad using the common controller to control a small unmanned ground vehicle
Hunn, John, Cahir, & Finch (2008)	x				x	x		A case on streaming video analysis portion of the geospatial intelligence process associated with an unmanned aircraft system

Table 6. Continued

Colombi, Miller, Schneider, McGrogan, Long & Plaga (2012)	x		x		x	x			An example involving a single pilot controlling multiple remotely piloted aircraft is presented
Plott, Quesada, Kilduff, Swoboda, & Allender, (2004)	x		x		x	x			A conceptual baseline and alternate configurations of the Unit of Action Mounted Combat System Company Headquarters are represented and evaluated.
Dixon, Wickens, & Chang (2005)				x					36 licensed pilots flew both single-UAV and dual-UAV simulated military missions in a baseline condition, an auditory autoalert condition, and an autopilot condition
Lamoureux (1999)			x	x					A simulator is used with 3 participants.
Wong, Walters, & Fairey (2010)	x				x	x			A discrete event simulation model developed for NASA's Orion Crew Vehicle. VACP and TLX used to measure mental workload. The model validated by SMEs.

Table 6. Continued

This Study	**x**				**x**	**x**		A simulation optimization model is used to improve the performance of Air Interdiction Planning Mission work process by regarding the capabilities and mental workload of employees.

2.3. Simulation Optimization

As discussed in the previous sections, task assignment problems generally are solved with deterministic optimization and mental workload analyses are studied by simulation models. In deterministic optimization, the uncertainty is ignored in order to come up with a unique and objective solution. On the other side, simulation approaches generally answer "what if" questions and it is time consuming to find optimal or near optimal solutions. According to Kelton (2000) an unplanned experimentation with a simulation model can often be inefficient. Alternatively, carefully planned simulation studies can give important information without unnecessary amount of computational effort time. Building on the capabilities of general simulation modeling, however, one can find the optimal setting of input variables through simulation optimization. The aim of the simulation optimization approach is to find the best input variable values from among all possibilities without explicitly evaluating each possibility (Carson & Maria, 1997). In other words, the objective is to minimize the resources spent while maximizing the information obtained from the simulation model. The differences of the approaches can be seen in Figure 2.

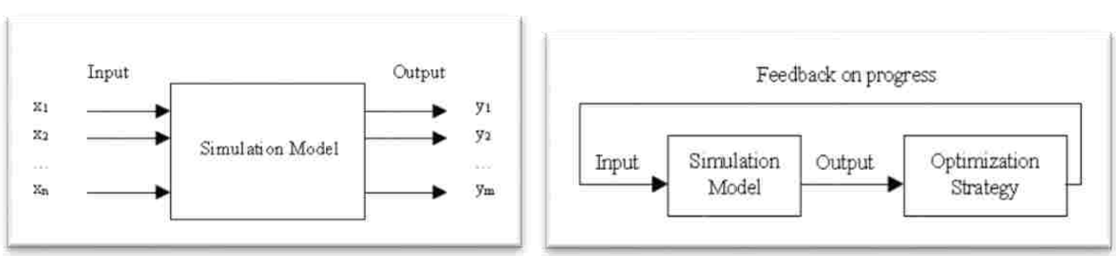


Figure 2. Simulation Model vs. Simulation Optimization Model (Carson & Maria, 1997)

The optimization of a simulation models starts with the need to find a set of model specifications such as input parameters and/or structural assumptions that leads to the optimal performance. In general, the range of parameter values and the number of parameter combinations are too large to test all possible scenarios. For example, many real world problems are too complex to be modeled by tractable mathematical formulations that are at the core of pure optimization methods (e.g. scenario optimization, robust optimization). To address such problems, simulation modeling is a way to guide the search for good solutions. Furthermore, pure optimization models are powerless in their abilities to capture all the complexities and dynamics of a highly sophisticated system. Thus, one must resort to simulation, which cannot easily find the optimal solutions. Simulation optimization resolves this problem of by merging the characteristics of pure optimization modeling and the use of computational simulations (Fu, 2002).

Simulation optimization can efficiently handle a much larger number of scenarios than traditional optimization approaches. Modern simulation optimization tools are designed to solve optimization problems of the form (Better et al., 2008):

$$\text{Minimize } F(x) \qquad \text{(Objective function)}$$

$$\text{Subject to: } Ax < b \qquad \text{(Constraints on input variables)}$$

$$g_l < G(x) < g_u \qquad \text{(Constraints on output measures)}$$

$$l < x < u \qquad \text{(Bounds)}$$

In the context of simulation optimization, a simulation model can be thought of as a "mechanism that turns input parameters into output performance measures" (Law & Kelton, 1991). In other words, the simulation model is a function (whose explicit form is unknown) that evaluates the merit of a set of specifications, typically represented as set of values. Here the vector x of decision variables includes variables that range over continuous values and variables that only take on discrete values. F(x) is the objective function, which is generally very complex. For example, one may be interested in measuring if the likelihood that a cycle time of a process will be lower than a desired duration. Known are the inequality $Ax \leq b$ is usually linear where the coefficient matrix A and the right-hand-side values corresponding to vector b. The constraints represented by inequalities of the form $g_l \leq G(x) \leq g_u$ impose simple upper and/or lower bound requirements on an output function G(x) that can be linear or non-linear. The bounds g_l and g_u are known constants. All decision variables x are bounded and some may be restricted to be discrete. Each assessment of F(x) and G(x) needs an execution of a simulation of the system.

The optimization procedure uses the outputs from the system evaluator, which measures the merit of the inputs that were fed into the model. One of the most preferred optimization methods is based on metaheuristic search algorithms.

The main optimization approaches used in simulation-optimization include random search (Andradottir, 2006), response surface methodology (Barton, 2005), gradient- based procedures (Fu, 2005), ranking and selection (Kinm & Nelson, 2005), sample path optimization (Goodfriend, 1995) and mostly metaheuristics (Ólaffson, 2005) including tabu search (Dengiz & Alabas, 2000; Yang, Kuo, & Chang, 2004), genetic algorithms (Azadivar & Tompkins, 1999; Zen, Wang, Hu, & Chang, 2014; Zeng & Young, 2009; Daniel & Rajendran, 2005; Yeh & Lin, 2007; McCormack & Coates, 2015; Ammeri, Dammak, Chabchoub, Hachicha, & Masmoudi, 2013; Ghazavi & Lotfi, 2016; Persson, Grimm, Ng, Lezama, Ekberg, Falk, & Stablum, 2006) and scatter search (Keskin, Melouk, & Meyer, 2010) or combination of several metaheuristics (Al-Aomar, 2006; Klassen & Yoogalingam, 2009; He, Huang & Chang, 2015). Table 7 shows the major categories of simulation optimization methods (Andradóttir, 2002; Carson & Maria, 1997; Fu, 2002; Kelton, 2000). Note that there is a huge application domain for simulation optimization technique from operations, manufacturing, and logistics to medicine and biology. A recent detailed review of algorithms and applications can be found in Amaran, Sahinidis, Sharda, and Bury (2016).

Table 7. Simulation Optimization Methods

Gradient Based Search Methods	Finite Difference Estimation
	Likelihood Ratio Estimators
	Perturbation Analysis
	Frequency Domain Experiments
Random Search/Heuristic Methods	Greedy Heuristics
	Genetic Algorithms
	Evolutionary Strategies
	Simulated Annealing
	Tabu Search
	Scatter Search
	Simplex Search
Statistical Methods	Importance Sampling
	Ranking and Selection
	Multiple Comparison
Stochastic Optimization	
Response Surface Methodology	
Sample Path Optimization	

Moreover, there are several simulation optimization commercial software programs based on various optimization methodology such as AutoStat, OptQuest, OPTIMIZ, SimRunner, and WITNESS Optimizer (Table 8). The current commercial software is a good start, but fails in two cases (Fu, 2002). First, algorithms that work extremely well are too specialized to be practical, or algorithms that apply very generally often converge too slowly in practice. Second, they do not guarantee local or global convergence (Amaran et al., 2016). Consequently, the optimization approach should be selected according to the problem on hand. In some cases, commercial software can be helpful but one should be aware of their weaknesses.

Table 8. Commercial Simulation Optimization Packages

Optimization Package	**Simulation Software Supported**	**Optimization Methodology**
AutoStat	AutoMod	Evolutionary Algorithms
Evolutionary Optimizer	ExtendSim	Evolutionary Algorithms
OptQuest	FlexSim, @RISK, Simul8, SIMPROCESS, Anylogic, Arena, Crystal Ball, Enterprice Dynamics, ModelRisk	Scatter Search, Tabu Search, Neural Networks
SimRunner	ProModel, MedModel, ServiceModel	Genetic Algorithms, Evolutionary Algorithms
RISKOptimizer	@RISK	Genetic Algorithm
WITNESS Optimizer	WITNESS	Simulated Annealing, Tabu Search, Hill Climbing
Plant Simulation Optimizer	Siemens PLM Software	Genetic Algorithm
ChaStrobeGA	Stroboscope	Genetic Algorithm
Global Optimization Toolbox	SimEvents Matlab	Genetic Algorithms, Simulated Annealing, Pattern Search

To summarize this section, the literature was reviewed in three parts; the assignment problem in work process design, mental workload analysis for work processes, and simulation optimization techniques. It has been found that task assignment problems for organizations are generally solved with deterministic optimization. However, the nature of an organization that embraces work processes is stochastic. In deterministic optimization, the uncertainty is ignored in order to come up with a unique and objective solution. On the other hand, mental workload analyses are generally studied by simulation models. However, simulation approaches generally answer "what if" questions and it is time consuming to find optimal or near optimal solutions. Building on the capabilities of general simulation modeling, however, one can find the optimal setting of input variables through simulation optimization techniques. The aim of the simulation optimization approach is to find the best input variable values from among all possibilities without explicitly evaluating each possibility. The main optimization approaches used in simulation-optimization include random search, response surface methodology, gradient- based procedures, ranking and selection, sample path optimization and mostly metaheuristics.

Mental workload measures are found to be appropriate in order to measure the workload of the employees for work processes in critical time junctions. Subjective mental workload measure methods are the most used methods in human engineering evaluation to evaluate the employee's rating of a task. These methods have good face validity and general applicability. The VACP method is the most preferred one because it is based on Multiple Resource Theory.

In order to solve the agent-task assignment problem for a work process to improve performance, a simulation model that supports VACP measure will be used in coordination with a metaheuristic search algorithm optimization engine.

3. METHODOLOGY

As reported in the literature review, work processes in general consist of different tasks, which require different expertise. Employees usually have various degrees of qualifications and their performance may vary for different tasks. Therefore, the performance outcome of an organization depends greatly on which tasks are assigned to which employees. Moreover, measurement methods and analysis of workload can improve the performance of employees, as well.

In a work process, the employee allocation procedure can be optimized by finding the set of skills that provide the optimal candidate for a particular task. This research claims that the right matching of employee-task is as important as not overloading the employees. The aim is to optimize performance by making sure that the employees stay under the mental workload limit, in other words they are not overloaded.

The upper workload limit represents the point where the proposed system's operator (employee) will be considered overloaded and cannot accomplish the tasks successfully (Huey & Wickens, 1993; Wickens, 2002). The literature review indicates that there is no “one” correct workload threshold and it should be defined according to the task process and workload measurement methodology considered. The upper workload limit can be investigated by analyzing or simulating the system (work process) under normal operational tempo (representing the baseline operation). Also, the upper workload limit can be determined by subject matter experts that are familiar with the work process.

The chosen mental workload measurement technique for this study is VACP. It is a subjective method based on MRT with good face validity and general applicability. Moreover, VACP's workload predictions are task-based predictions, it can be used even when no mock-up systems exist and it is applicable through discrete event simulation (Keller, (2002); Mitchell, (2009)). In addition, since the predictions are task-based, the workload of a task and workload threshold level does not change across agents.

In most of the VACP studies, 28 is used as the upper workload limit (Mitchell, (2000); Pomranky & Wojciechowski, (2007)). The logic behind this value can be explained as follows: According to the VACP scale, people have a limited set of resources available for mental processes. These resources can be thought of as a pool of energy that is used for a variety of mental operations, from sensory-level processing to meaning-level processing. The highest workload value of a resource (visual, auditory, cognitive, psychomotor) can be set is “7”. These VACP scales can be found in Table 7. Since four resources are used and they can be set to 7; the highest possible total workload of a task is 28 (4x7). As a result, based on the task process studied, the upper workload limit can be set to 28 or a value higher than 28 depending on the task process (Mitchell, 2009).

This research focuses on a work process with a team operating under a high operational tempo. Operational tempo can be defined as the frequency of the work orders. Accordingly, “high operational tempo” refers to the very frequent arrival of work orders. A work process with a high operational tempo is a collection of related structured tasks that produce a specific service or product, or serve a particular goal or mission, and should be finished as fast as possible. A work team consists of agents providing that particular service, product, goal or mission in an organization. Agents can be defined as a person that has a set of capabilities with different levels and can be assigned to complete one or more tasks (depending on the problem). The duration that the agent can finish a task depends on his or her capabilities and the capabilities required by the task that he/she is working on. Task is a specific unit of work characterized by a mental workload demand, and required capability levels.

Measuring the performance of work processes is a challenging one and there is no universal measure for performance, which is applicable to all work processes (Kamrani, Ayani, Moradi, & Holm, 2009). For this type of work process, the output that helps to measure the performance is *timeliness*. Timeliness is the key to the success for those critical processes such as emergency room tasks or military intelligence tasks. The average duration to finish a work process will be used to determine the timeliness. The aim is to minimize the average duration of the work process.

The accuracy of the work process depends on the agents. For the agent to be successful he/she should be capable enough and be able to handle the tasks. In other words, his capability levels should be equal or more than the required capability levels by the task. If the agent is not capable enough, he/she will not be successful and it will take more time to finish the task. Additionally, agent's workload should be under the workload threshold. Once he exceeds the workload threshold, he will be prone to making errors. Note that the workload of the agent affects the timeliness as well since an agent operating at the threshold workload level can't start a new task until the current task is finished, i.e., until there is enough residual workload capacity for the next task (because of the workload threshold constraint imposed to the problem). Therefore, in the assignment process, both capability level differences (the difference between agent's capability level and the capability level required by the task) and agents' workload levels affects the timeliness of the output. Timeliness is the surrogate for both variables (capability level differences and workload levels). The objective is to find the agent-task pairs in order to minimize the average duration of the task process.

The variables of this problem can be listed as follows:

Constants:

- A work process
  - A fixed ordered sequence of tasks (agents cannot complete tasks other than in the prescribed order)
  - Each task is defined by required capability types and levels
  - Each task has mental workload demands (according to the VCAP scale)
- A team
  - Each agent has capability types and levels
  - Each agent has an upper mental workload limit (workload threshold level which is the same for every agent in the work process)

Independent Variables:

- Task-agent assignments

Constraints:

- Instantaneous Mental Workload Threshold (set-up in simulation model): Agents should stay under the defined upper mental workload level (mental workload threshold). An agent cannot start an additional parallel task if it is going to increase the mental workload above the mental workload threshold level.
- Agent-Task Assignment Constraint (set-up in optimization engine):

  An agent can work on no tasks, one task, or more than one task, and a task can only be assigned to one agent.

Dependent Variables:

- Timeliness of work process: the objective is to minimize the average duration of the work process

To provide a simulation optimization environment with which to study task assignment and mental workload tradeoffs in a work process with a work team, a two-step model that includes (1) "Optimization Engine" and (2) "Simulation Model" will be used (Figure 3). This model will evaluate the hypothesis that the developed simulation-optimization model solves the task-employee assignment problem in order to minimize the duration in a reasonable time and efficient way.

In general, deterministic optimization models disregard the uncertainty in order to come up with a unique and objective solution. However, the nature of a work processes is stochastic, therefore the second step will provide a simulation model and be used to evaluate the results of the inputs suggested by the optimization part; it also permits the inclusion of various sources of uncertainty and variability into tasks that impact work process outcomes. The sources of uncertainty and variability are provided by the task completion time and inter-arrival time of work orders. As a result, it affects both the timeliness of the work process and the incidents when an agent is parallel tasking, which defines the instantaneous workload of an agent. Based on the results obtained from the simulation part, the optimization part will suggest improved input variables for the simulation part. This iterative process will continue until the stopping criteria are satisfied. The stopping criteria for this problem are based on acceptable outcomes of the simulation model. The output of this two-step model will show the timeliness (average duration) of the work process while mental workload of each employee stays under the threshold.

Generally, in simulation optimization studies, the iterative process is stopped when convergence is achieved, which means there is no improvement on the best solution found so far after a defined number of iterations. In this study, a convergence factor and a maximum number of iterations will be defined for the stopping criteria.

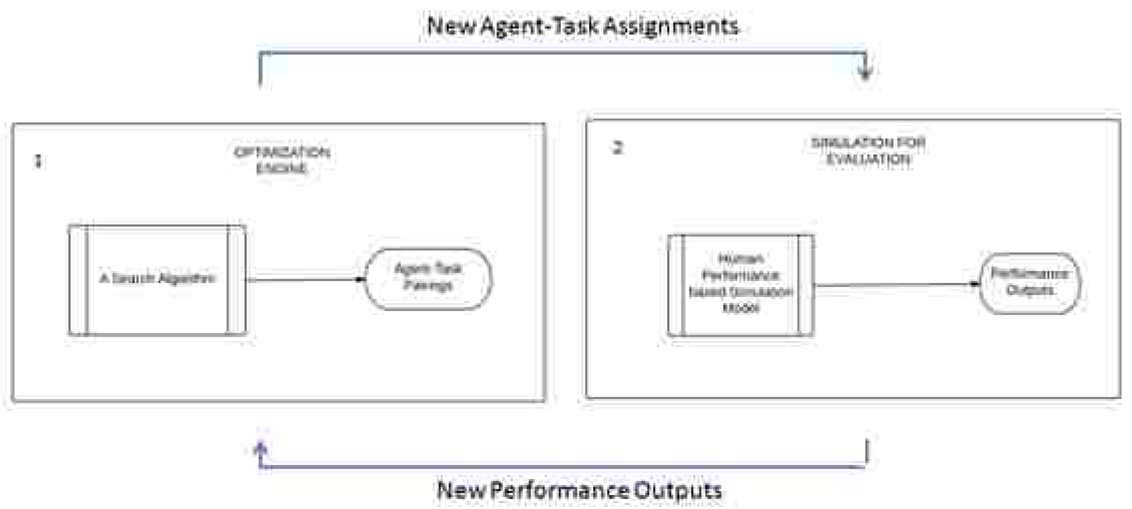


Figure 3. Simulation-Optimization Approach

This simulation optimization approach will improve timeliness (which minimizes average duration) by making more efficient task assignments and keeping the agents under workload threshold to prevent them from making mistakes. Moreover, an experiment will be designed and executed to provide guidance on the tradeoffs between task assignment and workload in terms of different levels of operational tempo (the inter-arrival rate of work orders). The next section explains the details of these two parts, the simulation model and the optimization model with the stopping criteria. Moreover, the software requirements in order to develop such simulation-optimization approach are discussed and the software pair used in this study is described.

3.1 Simulation Optimization Approach for Work Process Design

3.1.1 Methodology for Optimization Engine of Task Assignment

As mentioned in the literature review section, the general simulation optimization problem form (Better et al., 2008) is as follows:

Minimize/ Maximize F(x) (Objective function)

Subject to: Ax < b (Constraints on input variables)

$g_l < G(x) < g_u$ (Constraints on output measures)

l < x < u (Bounds)

In this study, we are dealing with a generalized assignment problem (GAP). GAP involves finding the minimum cost assignment of n tasks to m agents such that each task is assigned exactly to one agent, subject to agent's available capacity. It can be defined as follows (Chu & Beasley, 1997):

Let $I = \{1,2, \ldots, m\}$ be a set of agents, and let $J = \{1,2, \ldots, n\}$ be a set of tasks. For $i \in I$, $j \in J$ define c_{ij} as the cost of assigning task j to agent i (or assigning agent i to task j), r_{ij} as the resource required by agent i to perform task j, and b_i as the resource availability (capacity) of person i. Also, x_{ij} is a 0-1 variable that is 1 if agent i performs task j and 0 otherwise. The mathematical formulation of the GAP is:

$$\text{Minimize} \quad \sum_{i \in I} \sum_{j \in J} c_{ij} x_{ij} \quad (1)$$

Subject to

$$\sum_{i \in I} x_{ij} = 1, \quad \forall j \in J \quad (2)$$

$$\sum_{j \epsilon J} r_{ij} x_{ij} \leq b_i, \quad \forall i \epsilon I \ (3)$$

$$x_{ij} \epsilon \{0,1\}, \forall i \epsilon I, \forall j \epsilon J \ (4)$$

Equation (2) ensures that each job is assigned to exactly one person and Equation (3) ensures that the total resource requirement of the jobs assigned to a person does not exceed the capacity of the agent. Equation (4) is the binary variable constraint.

Table 9 shows the adjustment of the GAP to the simulation optimization problem form of the task-agent assignment problem at hand. Equation (5) is the objective function that represents timeliness, which minimizes the average duration of the work process. Equation (6) ensures that each job is assigned to exactly one agent. The function $G(x_{ij})$ shows the instantaneous mental workload level of each agent. Equation (7) ensures that the instantaneous workload level of an agent stays below the upper bound (g_u) and this constraint is handled in the simulation model. Each assessment of $F(x_{ij})$ and $G(x_{ij})$ needs an execution of a simulation of the system. Equation (8) ensures that x_{ij} is a binary variable.

Table 9. Formulation of GAP for Simulation Optimization

GAP	GAP for Simulation Optimization	
Minimize $\sum_{i \epsilon I} \sum_{j \epsilon J} c_{ij} x_{ij}$	Minimize $F(x_{ij})$	(5)
Subject to: $\sum_{i \epsilon I} x_{ij} = 1, \quad \forall j \epsilon J$	Subject to: $\sum_{i \epsilon I} x_{ij} = 1, \quad \forall j \epsilon J$	(6)
$\sum_{j \epsilon J} r_{ij} x_{ij} \leq b_i, \quad \forall i \epsilon I$	$G(x_{ij}) < g_u \quad \forall i \epsilon I$ $Handled\ in\ simulation\ model$	(7)
$x_{ij} \epsilon \{0,1\}, \forall i \epsilon I, \forall j \epsilon J$	$x_{ij} \epsilon \{0,1\}, \forall i \epsilon I, \forall j \epsilon J$	(8)

The optimization procedure uses the outputs $F(x_{ij})$ from the system evaluator, which measures the merit of the inputs that were fed into the model, see Figure 4.

The literature review indicates that one of the mostly preferred optimization procedures for simulation optimization problems is based on metaheuristic search algorithms. The metaheuristic optimizer chooses a set of values for the input parameters and uses the responses generated by the simulation model to make decisions regarding the selection of the next trial solution. Chu and Beasley (1997) found that one of the heuristics that have been superior to others for solving GAP is the genetic algorithm (GA). Moreover, GA is a popular method and has proved to be effective algorithm in simulation optimization studies.

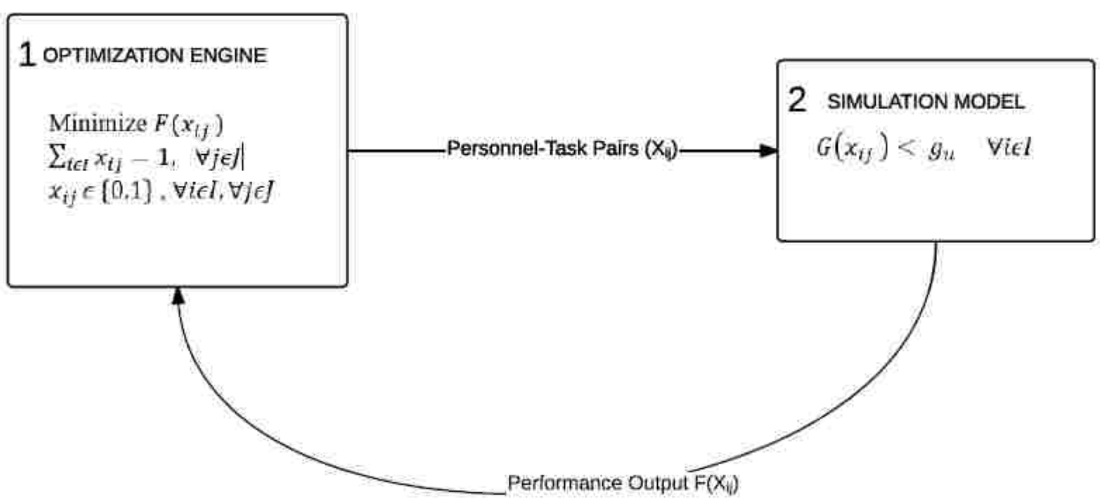


Figure 4. Graphical Illustration of Optimization Engine

GA was first introduced by Holland (1975). GA is an intelligent probabilistic search algorithm which simulates the process of evolution by taking a population of solutions and applying genetic operators in each reproduction. Each solution in the population is evaluated according to some fitness measure. Highly fit solutions in the population are given chances to reproduce. New offspring solutions are generated and unfit solutions in the population are replaced. This evaluation-selection-reproduction cycle is repeated until a satisfactory solution is found.

Genetic algorithms deal with a population of solutions and tend to manipulate each solution in a simple way. In a GA, a potential solution to a problem is represented as a set of parameters known as a gene. These parameters are joined together to form a string of values known as a chromosome. A good representation scheme is essential in a GA. It should clearly define meaningful crossover, mutation and other problem-specific operators in order to minimize computational effort is involved in these procedures.

The detailed description of developed GA for the simulation optimization assignment problem is as follows:

Chromosome definition: An efficient representation is used in which the solution structure is an ordered structure (n-dimensional vector) of integer numbers. These integer numbers identify the agents, as assigned to vector elements denoted by the tasks (see Figure 5). This representation ensures that all the equality constraints in equation (6) are automatically satisfied since exactly one agent is assigned to each job.

Task	1	2	3	4	5	…	n-1	N
Agent	2	1	5	M	6	…	3	10

Figure 5. Representation of an Individual's Chromosome

Initialization of population: N randomly constructed initial solutions are generated by randomly assigning an agent to a task without allowing duplicate solutions in the population.

Fitness evaluations of chromosomes: The fitness f_k , of solution k is equal to its objective function value $F(x_{ij})$ which is the output from the simulation (average duration of the work process).

$$f_k = F(x_{ij})$$

The simulation model produces the fitness values for the GA. This fitness values are read and each fitness value assigned to each solution in the population.

Crossover: The binary tournament selection method is used. In a binary tournament selection, two individuals are chosen randomly from the population. The more fit (smaller fitness value) individual is then allocated a reproductive trial. In order to produce a child, two binary tournaments are held, each of which produces one parent. A child solution is created by first applying a crossover operator to the selected parents. The one-point crossover operator is used, in which a crossover point is selected randomly and the child solution will consist of the first p genes taken from the first parent and the remaining (n - p) genes taken from the second parent, or vice versa with equal probabilities.

Moreover, a *similarity ratio* is defined in order to keep the diversity of the solutions in the population. Based on this ratio, if a solution is similar to one of the solutions in the population (based on the ratio defined; such as if the ratio is 0.1, then it is a solution that has only one different gene from another solution in a 10-member population), it is not allowed to enter in the population and a new binary tournament is started.

In the population replacement scheme, the individual in the population with the lowest fitness is replaced. Note that a duplicate child is not allowed to enter the population.

Mutation: The crossover procedure is followed by a mutation procedure. This mutation procedure involves exchanging elements in two randomly selected genes (i.e. exchanging assigned agents between two randomly selected jobs). In addition to that, "*mutation increase rate*" that increases the mutation rate gradually (by 0.01%) as every time the algorithm approaches half way to the convergence (convergence factor/2) is introduced.

Termination Condition: The above-mentioned evaluation process is repeated until the termination conditions have been reached (Figure 6). The termination conditions for this process are a convergence factor and a maximum allowed number of iterations. The convergence factor is achieved when the best individual objective value has not been updated in 100 successive iterations. The maximum number of iterations is set to 1000.

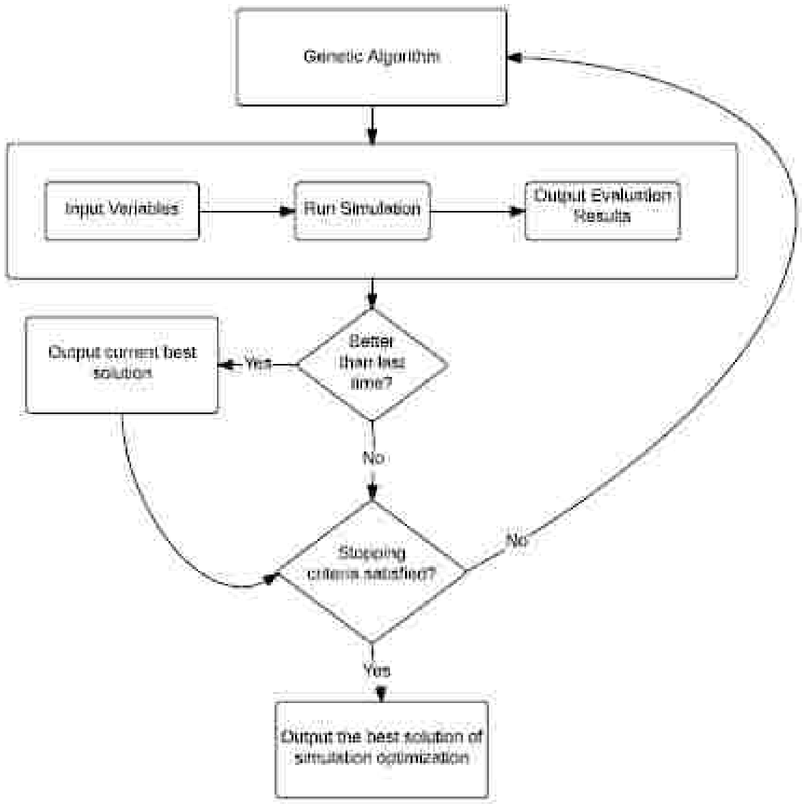


Figure 6. Termination condition flowchart of simulation optimization approach

Note that mutation and crossover processes are used to avoid local optimum solutions. Moreover, additional constraints, namely *similarity rates* and *mutation increase rate*, are added to the genetic algorithm in order to keep the diversity of the solutions in the population high which helps avoid the local optimums as well. It is important to mention that the right population size (PS), crossover rate (CR) and mutation rate (MR) should be set in order to increase the effectiveness of the algorithm. These values can be set by calculating relative changes for different sets of PS, CR, MR.

3.1.2 Methodology to develop a Simulation of a Work Process

For this particular problem, simulation software that has human performance modeling capability is an appropriate one. The steps followed in order to develop a human performance work processes simulation model are shown in Figure 7, and the explanation of the steps are as follows:

2.1. Develop Work Process Flow:

The simulation model for a work process is comprised of a series of tasks, which are connected as a network. As a result, the work process flow is created as a network diagram. All the branching rules, prerequisites in work process flow are defined in this step. Note that the flow of the tasks is fixed and the agents cannot complete tasks other than in the prescribed order. Moreover, required capability and capability levels for each task are defined as variables.

2.2. Create Work Team:

For this step, the agents that are available to participate in the work process are defined. Each available agent is created separately so that in the following steps they will be ready to be assigned to tasks (in step 2.6). In addition, their capabilities and capability levels are defined as variables.

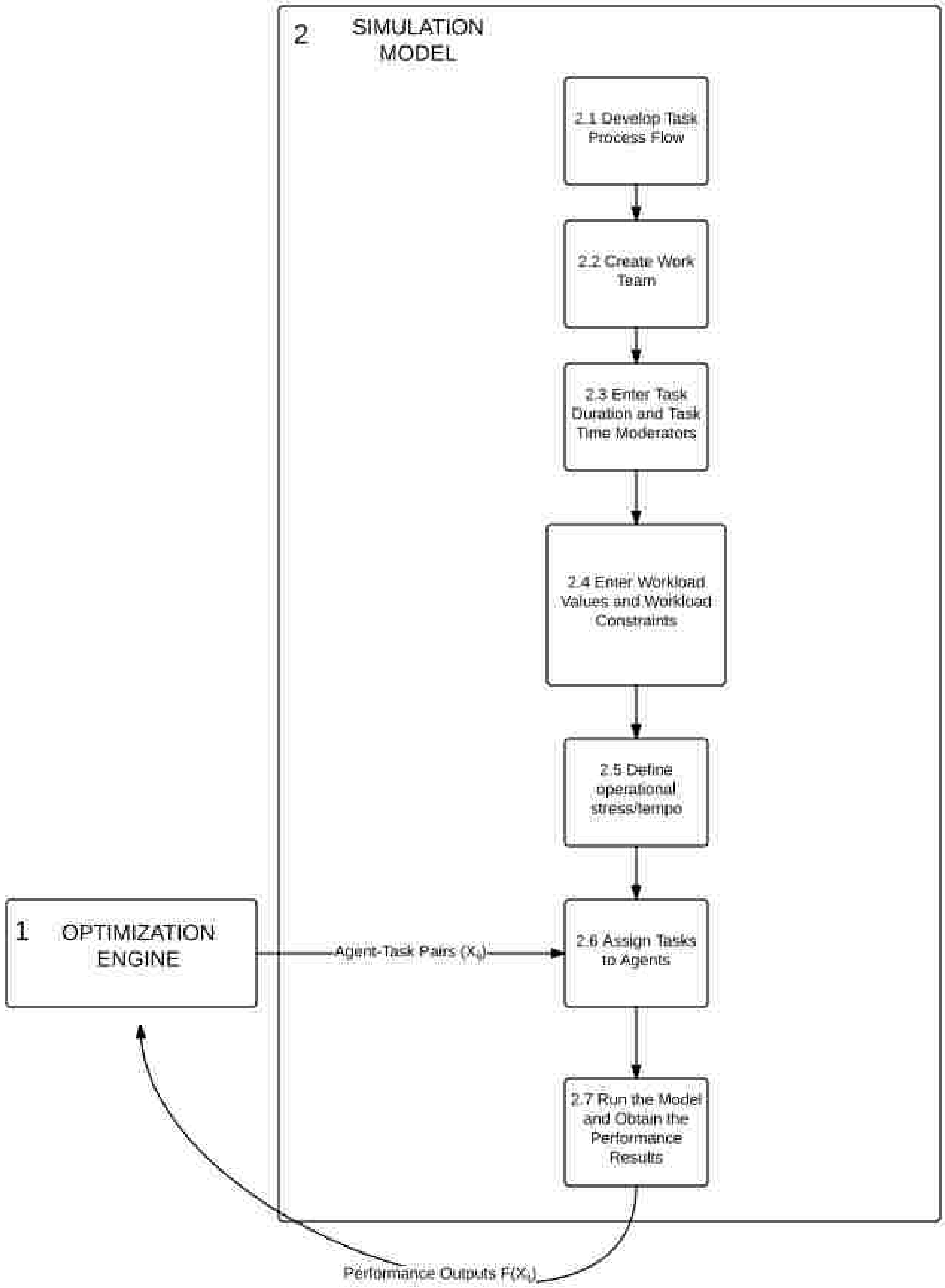


Figure 7. Steps for Developing a Work Process Simulation

2.3. Enter Task Durations and Task Duration Moderators:

The length of time each task usually takes is entered as a probabilistic distribution including the variables that will indicate the change in task duration according to difference between capability level of agent and required capability level by task. If the agent's capability level is lower than the required capability level by task, the duration of the task is increased by the percentage (k) that the developer of the model defined for each capability level difference; this percentage is called *capability level difference coefficient*. Moreover, to avoid an agent to be assigned to a particular task or to prevent an agent from being assigned at all, a very big capability level difference can be defined in advance.

2.4. Enter Workload Values and Workload Threshold Constraints:

In this step, the VACP workload scale that are consistent with well-known and documented theories of workload prediction, including the Multiple Resource Theory (MRT) (Wickens, 2002) is used. The rating scale of VACP can be found in Table 10. The corresponding workload values for each resource from the scale that shown in Table 10 is entered as a variable for each task. The total mental workload demand of a task is the sum of the entered workload values of each resource. The mental workload threshold, which is the same across the agents, is defined. It is important to remember that the mental workload threshold represents the point where the proposed system's operator will be considered overloaded and cannot accomplish the tasks successfully. This constraint enforces the rule that the agents cannot start to a new task if they do not have enough remaining residual capacity. The residual capacity is the difference between agent's instantaneous workload level and the workload threshold level. As soon as an agent has enough residual capacity, he/she can start to a new/parallel task.

Table 10. VACP workload estimation scales

Workload Demand Value
<u>Visual</u>
3.0 - Visually Register/Detect
3.0 - Visually Inspect/Check
4.0 - Visually Locate/Align
4.4 - Visually Track/Follow
5.0 - Visually Discriminate
6.0 - Visually Scan/Search/Monitor
5.1 - Visually Read
<u>Auditory</u>
1.0 - Detect/Register Sound
2.0 - Orient to Sound (general)
4.2 - Orient to Sound (selective)
4.3 - Verify Auditory Feedback
3.0 - Interpret Semantic Content (speech) Simple (1-2 words)
6.0 - Interpret Semantic Content (speech) Complex (sentence)
6.6 - Discriminate Sound Characteristics
7.0 - Interpret Sound Patterns
<u>Cognitive</u>
1.0 - Automatic (simple association) All values below 7.0 map to
1.2 - Alternative Selection Solving
3.7 - Sign/Signal Recognition
4.6 - Evaluation/Judgement (single aspect)
5.0 – Rehearsal
5.3 - Encoding/Decoding, Recall
6.8 - Evaluation/Judgement (several aspects)
7.0 - Estimation, Calculation, Conversion
<u>Fine Motor</u>

Table 9. Continued

2.2 - Discrete Actuation (button, toggle trigger)
2.6 - Continuous Adjustive (flight control, sensor control)
4.6 - Manual (tracking) Fine Motor Discrete
5.5 - Discrete Adjustive (rotary, vertical thumb wheel, lever position)
6.5 - Symbolic Production (writing)
7.0 Serial Discrete Manipulation (keyboard entries)
Gross Motor
1.0 - Walking on Level Terrain
2.0 - Walking on uneven terrain
3.0 - Jogging on Level Terrain
3.5 - Heavy Lifting
5.0 - Jogging on Uneven Terrain
6.0 - Complex Climbing

2.5. Define Operational Tempo:

Operational tempo is the frequency of the inter-arrival times of the work orders. As the inter-arrival times of the work orders decreases, operational tempo increases. Additionally, as the operational tempo increases, the likelihood of an agent working on parallel tasks increases (which increases the likelihood of increasing his workload level). In this step, the operational tempo ranges can be defined according to the experimental design.

2.6. Assign Tasks to Operators:

Creation of the logic that changes the assignments according to the suggestions from the optimization module is necessary. According to the task-agent pairs from the optimization module, each task will be assigned to the suggested agent. The assignments are made automatically depending on the communication structure between simulation model and optimization engine, i.e., the assignments can either be read from a text file or obtained directly from the optimization engine (in case there's a developed connection structure between optimization engine and simulation model). Explained in section 3.2 are the software requirements in order to achieve this communication.

2.7. Run the Model and Obtain Results:

During execution, the simulation model calculates task durations (implementing the task time moderators) and the workload of an agent over time. The output file shows the *average duration of the work process*, which is the indicator of the *timeliness*.

3.2 Software Requirements

There are two different requirements in order to develop a simulation optimization that assign agents to tasks and consider agents' capabilities and mental workload levels. These are the "capability of the simulation software" and "communication requirements between optimization engine/code (language that is used) and simulation model".

3.2.1 Required capability of the simulation software

In order to develop a simulation model for a work process that captures workload level of agents and reflect the changes in agent's capability levels on task durations, the following steps should be able to be implemented using the simulation software:

- Create task network
- Assign agents to tasks
- Assign capabilities and capability levels to agents

- Assign required capabilities and capability levels to tasks
- Define task time moderators (capability levels impacts the duration of the task)
- Assign workload to tasks
- Calculate instantaneous workload levels of the agents
- Calculate average duration of the work process

If these criteria can be satisfied with the simulation software on hand, the second step is finding the right optimization code and/or tool in order to achieve the communication between the two.

3.2.2 Communication requirements between simulation and optimization

There are three crucial requirements in order to achieve the communication between the simulation and the optimization engine. These requirements can be summarized as follows:

1. Optimization code or software starts the simulation run automatically**:** The simulation program should be reachable from the command line on the computer (should be able to be saved as an .exe file) or has a coding environment or add-in that has already been integrated (such as ARENA and Visual Basic for Applications (VBA)). Most of the coding language can call a command line prompt (in Windows). That way the simulation run can be started from the command line.

2. Optimization receives output from simulation: The optimization engine should be able to get the fitness values from the simulation. This can be achieved either reading the simulation outputs from a file that the simulation model created after each run or, in the case where an add-in is available, the optimization engine can be integrated with the simulation model so simulation model can pass the output values to optimization engine.

3. Simulation receives input from optimization: The simulation model should be able to get the agent-task configuration list from the optimization. It can be achieved either by reading the configuration list from a file that the optimization engine created or the optimization engine should be integrated to the simulation model through an add-in, so that optimization engine can pass the configuration list to simulation model.

3.2.3. The chosen simulation software and coding language pair for this study: ARENA-JAVA

For this study, ARENA was the preferred simulation-modeling tool. ARENA because; first, it is flexible enough to model a work process and satisfies all the requirements explained in section 3.2.1. Secondly, it runs from the command line of Windows. Lastly, it has the capability of reading from files and writing to files.

ARENA software's underlying language called SIMAN. When the model developed in ARENA, ARENA produces SIMAN code. Then, the SIMAN code is compiled and executed. First, two files associated with the SIMAN program are generated. These are the mod (model) and exp (experiment) files. The mod file contains the SIMAN code of the flowchart modules in an ARENA model window. The exp file contains the SIMAN code for the data modules and simulation run control parameters that are used during the execution of the simulation. ARENA software uses MODEL.exe to generate "m" file from mod file and EXPMT.exe to generate "e" file from exp file. These generated files are used to create "p" file by combining m and e files by LINKER.exe. The p file is the complete simulation model (flowcharts, data modules and simulation control parameters) that can be executed. The simulation model (p file) can be executed using SIMAN.exe. MODEL.exe, EXPMT.exe, LINKER.exe and SIMAN.exe files and necessary dll files can be found under the installed ARENA software file which located in directory the user downloaded ARENA. The flowchart of creating and executing these necessary files can be found in Figure 8.

Once the simulation model in ARENA completed, mod and exp files can be written using Run> SIMAN> Write, then from the optimization engine the necessary exe files can be called using command line in windows and necessary input files can be entered to these exe files. The sequence in optimization engine should be as follows:

1. Create input file for simulation
2. Run simulation (Call exe files and enter input files)
3. Read the output file generated by the simulation model

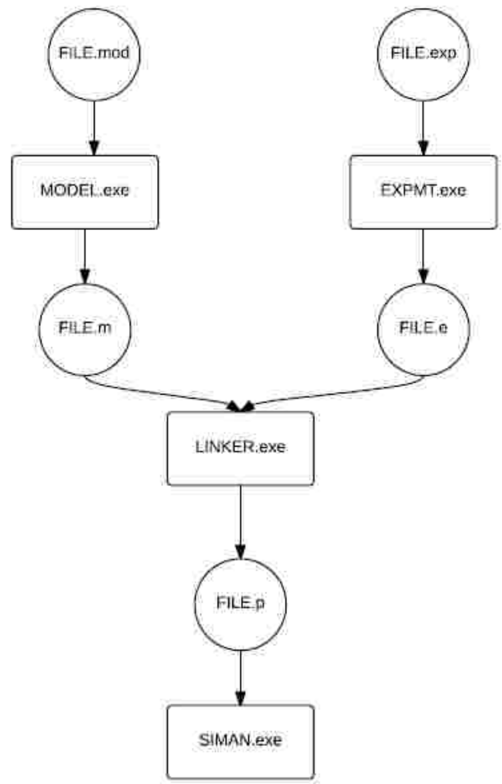


Figure 8. Necessary files to run an ARENA simulation model through SIMAN.exe (Seppanen, 2016)

As the optimization engine, Genetic Algorithm is coded in JAVA. JAVA is an object oriented language that allows creating modular programs and reusable codes. The “cmd line” can be accessed through coding in JAVA then the necessary exe files can be run. Moreover, JAVA is platform independent. It can be moved easily from one computer system to another. Furthermore, it is easy to write, compile and debug than other programming languages.

To summarize, the stages of developing a simulation optimization approach that would improve timeliness of a work process in order to minimize average duration by making more efficient task assignments and keeping the agents under workload threshold to prevent them from making mistakes is explained. This method is comprised of two parts: the simulation model, and the optimization engine with stopping criteria. A genetic algorithm will be used for the optimization engine. Chromosome definition, initialization of population, evaluation of chromosomes, crossover and mutation methods that will be used for this algorithm are explained in detail. The stopping criteria for this process are the convergence factor, which is best individual objective value not updated in 100 successive iterations, and maximum number of iterations, which is 1000. Presented are the steps followed in order to develop a human performance work processes simulation model. These steps are; develop work process flow, create work team, enter task durations and task duration moderators, enter workload values and workload constraints, define operational tempo, assign task to operators, run the model and obtained results. In order to develop this method; GA based optimization engine will be coded in JAVA and the human performance simulation model will be developed in ARENA. The two-way communication between optimization engine and simulation model will be achieved by text files. In the next chapter, an example work process called *Air Interdiction Planning Mission* will be used to develop the GA based simulation optimization tool by following the steps explained in the current chapter.

4. MODEL DEVELOPMENT AND ALTERNATIVES

In the previous chapter, a methodology was described to create a simulation optimization model for a work process at a critical time junction to test hypotheses of improved performance. This method is used to create a model for a hypothetical "Air Interdiction Planning" mission and team. The model will be run under different settings including varying operational-tempo to evaluate the performance of the proposed method. The results will then be compared to the results obtained through applying the common practices that have been generally used in organizations that represents the bounds of the problem. Moreover, the performance of the developed GA based simulation-optimization method will be compared with a commercial simulation optimization engine OptQuest. The next two sub-sections describe the implementation of the methodology on the selected case and explain the alternative methods (common practices and other simulation optimization approaches) that will be compared with the developed GA based simulation-optimization method.

4.1. Implementation of Methodologies in a Prototype Application: "Air Interdiction Planning Mission"

A fictitious case of an Air Force Air Interdiction Mission Planning work process is used to evaluate the performance of the methodology designed for this research. The objective of the air interdiction mission is to divert, disrupt, delay, or destroy the enemy's military potential before it can be brought to bear effectively against friendly forces. Air interdiction is conducted at such distance from friendly forces that detailed integration of each air mission with the fire and movement of friendly forces is not necessary (Grooms, 2009). It requires fast planning and action to be effective. A team that receives real-time information about enemy positions and friendly positions, and requests for air support performs the planning of this mission. The team has to analyze the requests in the context of the prevailing situation and plan missions as fast as possible without any errors.

This system was chosen for the study because it meets the criteria of the problem definition, for instance, this example includes a team with given number of agents and flow of tasks. The process must be accomplished in a critical time, and as fast as possible without any errors. In order to achieve this aim, agents should stay under their workload threshold while parallel tasking. Furthermore, the capabilities of the agents and required capabilities by the tasks have critical effects on the task durations.

In the next two sections, the prototype problem will be modeled following the defined steps in order to develop the GA based simulation optimization method.

4.1.1 Optimization Engine for "Air Interdiction Planning Mission" Process

The Air Interdiction Mission Planning has 10 main tasks that lead to several subtasks as shown in Table 9 (Perdu, 1997). There are 10 agents (decision makers) in the mission with different capabilities. The workload threshold value is set to 28 (see previous explanation). It means that the instantaneous workload of an agent cannot exceed 28. Note that this threshold is constant across all agents since the VACP workload scale is a task based scale and not related with agents' characteristics.

Let $\mathrm{I} = \{1,2, \ldots ,10\}$ be a set of agents, and let $\mathrm{J} = \{1,2, \ldots ,10\}$ be a set of tasks.

The mathematical model is formulated as follows:

$$\text{Minimize } F(x_{ij}) \qquad (9)$$

$$\text{Subject to: } \sum_{i\epsilon I} x_{ij} = 1, \quad \forall j \epsilon J \qquad (10)$$

$$G(x_{ij}) < 28 \quad \forall i \epsilon I \text{ (Handled in simulation model)} \qquad (11)$$

$$x_{ij} \epsilon \{0,1\}, \forall i \epsilon I, \forall j \epsilon J \qquad (12)$$

Objective function (9) represents timeliness, which minimizes the average duration of the work process. Equation (10) ensures that each job is assigned to exactly one agent. The function $G(x_{ij})$ shows the instantaneous mental workload level of each agent. Equation (11) ensures that the instantaneous workload level of an agent stays below the upper bound 28 and this constraint is handled in the simulation model. Each assessment of $F(x_{ij})$ and $G(x_{ij})$ needs an execution of a simulation of the system. Equation (12) ensures that x_{ij} is a binary variable and handled in GA code.

The GA is coded in JAVA following the explanations mentioned in section 3.1.1 on chromosome definition, initialization of population, fitness evaluation of chromosomes, crossover and mutation rules, and termination condition.

4.1.2 Simulation Model for "Air Interdiction Planning Mission" Process

4.1.2.1 Logic and Data of the Simulation Model

The steps that have been indicated in Figure 6 have been followed to develop the simulation of the prototype problem. The simulation model is developed using ARENA simulation software. The steps and the data that have been used in each step are as follows:

1. Develop Work Process Flow:

The decomposition of the Air Interdiction Planning Mission has led to the 10 main tasks and each task further decomposed to several subtasks as listed in Table 11 (Perdu, 1997). The first task, “Analyze request” has a request coming from units on the battlefield, or an intelligence report about some enemy movement behind the enemy lines. This task evaluates the threat of the enemy forces in the battlefield area designated by the request or the report. The second task, "Produce threat characteristic data" evaluates the threats and generates a threat characteristic data report. The third task, "Get enemy data" assesses and evaluates the enemy position in the battlefield, especially in the area designated by the request or the intelligence report and generates an enemy position in the battlefield report. The following task "Generate an enemy posture report" evaluates the enemy positions and generates the enemy posture report. The task "Target development and prioritization" prioritizes the targets. The following task produces an aimpoint report. The task "Perform weaponing" defines the best weapon to destroy the target. "Evaluate air defense capability in the area of interest" defines battlefield environment and determines threat course of actions. It generates an air defense capability report. "Forecast the degree of redundancy" calculates the degree of redundancy necessary for the objective. The last task, "Plan mission" delivers the final output of the team. It combines the target data and the information contained in all the reports to define completely the mission.

Table 11. Main and Subtasks of Air Interdiction Mission Planning

1	**Analyze request**
	Receive a request for CAS or intelligence report
	Read the request
	Evaluate the threat of the enemy forces in the battlefield area
	Produce a threat report
2	**Produce threat characteristic data**
	Evaluate the threat report
	Weight the threats likelihood
	Generate threat characteristics data
	Produce threat characteristic data report

Table 11. Continued

3	**Get enemy data**
	Receive enemy position data
	Assess and evaluate the enemy position in the battlefield
	Generate the data sheet
	Produce the enemy position in the battlefield report
4	**Generate an enemy posture report**
	Obtain enemy position in the battlefield report
	Assess and evaluate the enemy positions
	Generate the data sheet for report
	Generate an enemy posture report
5	**Perform target development/prioritization**
	Obtain enemy posture report
	Mark the targets
	Prioritize the targets
	Generate target development/prioritization report
6	**Perform aimpoint construction**
	Analyze target report
	Construct the aimpoint
	Generate the aimpoint data
	Produce aimpoint report
7	**Perform weaponeering**
	Obtain target report
	Obtain aimpoint report
	Perform weapon selection
	Produce weaponeering report
8	**Evaluate air defense capability in the area of interest**
	Define battlefield environment
	Describe the battlefield's effect
	Determine threat courses of actions
	Produce air defense capability report
9	**Forecasts the degree of redundancy necessary for the objective**
	Obtain air defense capability report
	Obtain aimpoint report
	Calculate degree of redundancy necessary for the objective
	Produce degree of redundancy report
10	**Produce mission plan**
	Obtain all the reports
	Check the reports

Table 11. Continued

	Combine the reports
	Produce mission plan

The flow chart representation of the process is shown in Figure 9. Inputs from the environment are obtained, respectively, by tasks 1 and 3. The output of task 1 is processed by task 2. The output of task 3 is processed by task 4, and so forth. The flow chart on Figure 9 shows the predecessor-successor relationships, as well. Finally, task 10 needs the results of tasks 7 and 9 to produce the team output. Each task has 4 required capabilities in different levels. These capabilities are numerical analysis, problem solving and decision-making, communication, and computer skills (Kamrani et al., 2009). The capability level scale is from 1 to 5. 1 means low level and 5 means high level of capability required. The capability levels required by tasks are shown in Table 12.

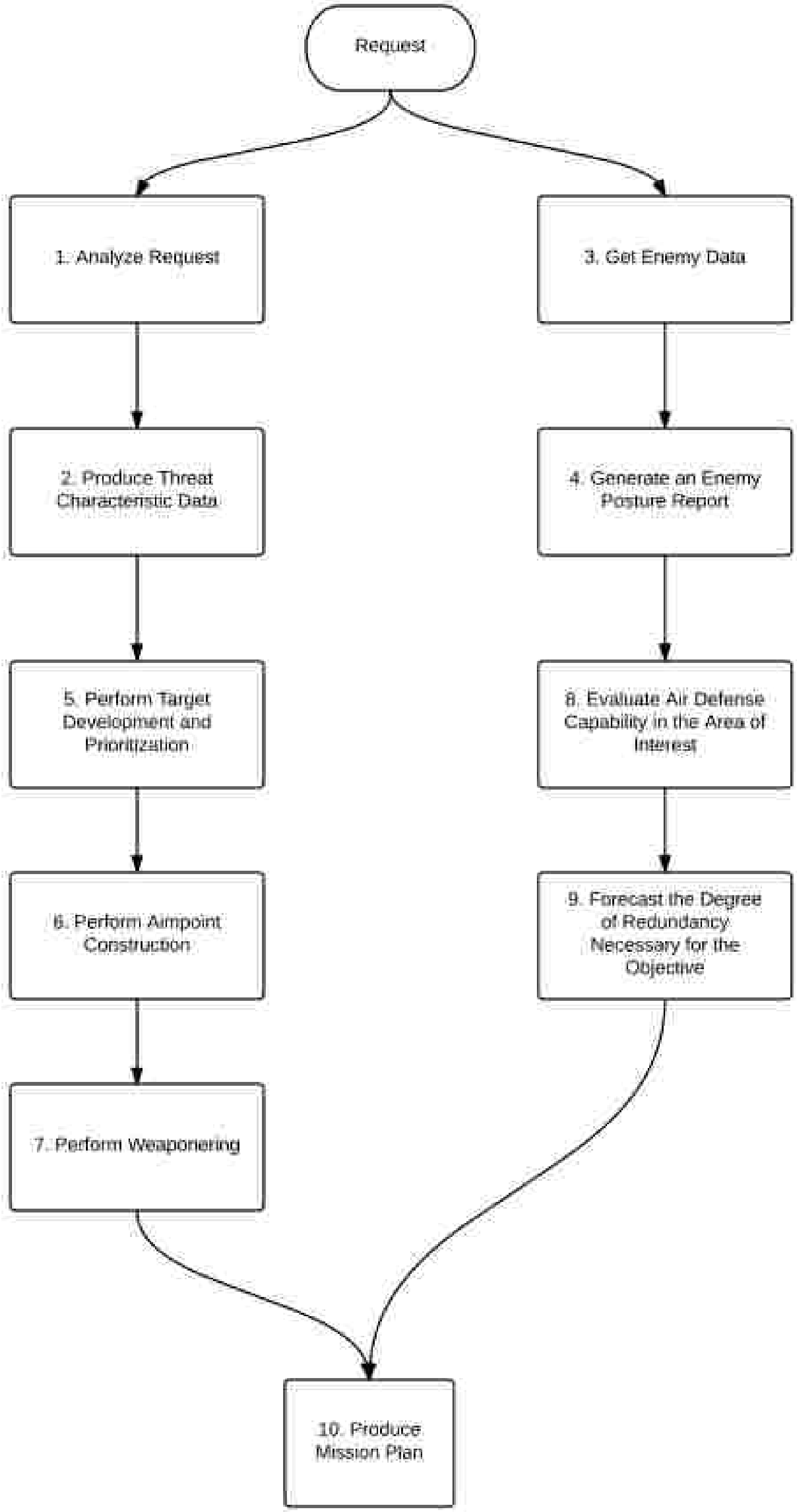


Figure 9. Flow chart of the mission

Table 12. Required Capability Levels by Tasks

Task / Required capability level	Numerical Analysis	Problem Solving and Decision Making	Communication (Read and Write)	Computer Skills (Software Experience)
Analyze request	1	5	3	3
Produce threat characteristic data	3	4	3	2
Get enemy data	1	1	3	5
Generate an enemy posture report	1	1	5	3
Perform target development/prioritization	3	3	3	3
Perform aimpoint construction	4	3	2	4
Perform weaponeering	3	3	2	2
Evaluate air defense capability in the area of interest	4	4	2	3
Forecasts the degree of redundancy necessary for the objective	5	3	1	3
Produce mission plan	4	3	5	5

2. Create Agents:

There are 10 agents (DM1, …, DM10) in the mission with 4 capabilities in different levels. These capabilities are numerical analysis, problem solving and decision-making, communication, and computer skills (Kamrani et al., 2009). The capability levels of each agent are shown in Table 13.

Table 13. Capability Levels of Agents

	Numerical Analysis	Problem Solving and Decision Making	Communication (Read and Write)	Computer Skills (Software Experience)
Agent1	1	1	5	3
Agent2	3	3	3	3
Agent3	4	3	2	4
Agent4	3	3	2	2
Agent5	4	4	2	3
Agent6	5	3	1	3
Agent7	4	3	5	5
Agent8	1	5	3	3
Agent9	3	4	3	2
Agent10	1	1	3	5

3. Enter Task Durations and Task Time Moderators:

The default duration of each task are shown in Table 14. The durations are characterized by a triangular distribution since only the minimum, maximum and most likely durations are known. Triangular distribution is selected because it is a rough approximation to a random variable with an unknown distribution. The difference between agent's capability level and required capability level by task affects the durations. If agent lacks a capability, it increases the duration on the task. A weight is used for each level difference. It is called *capability level difference coefficient*. Currently this weight is set to 0.5, which means a one level difference between task and agent capacity increases the task duration by 50% regarding the probability distribution of the duration. The duration of work process is calculated as duration= finish time-start time.

Table 14. Tasks and Their Default Durations

	Task	Duration
1	**Analyze request**	
	Receive a request for CAS or intelligence report	Triangular (4,5,6)
	Read the request	Triangular (4,5,6)
	Evaluate the threat of the enemy forces in the battlefield area	Triangular (8,10,12)
	Produce a threat report	Triangular (4,5,6)
2	**Produce threat characteristic data**	
	Evaluate the threat report	Triangular (4,5,6)
	Weight the threats likelihood	Triangular (8,10,12)
	Generate threat characteristics data	Triangular (4,5,6)
	Produce threat characteristic data report	Triangular (4,5,6)
3	**Get enemy data**	
	Receive enemy position data	Triangular (4,5,6)
	Assess and evaluate the enemy position in the battlefield	Triangular (8,10,12)
	Generate the data sheet	Triangular (4,5,6)
	Produce the enemy position in the battlefield report	Triangular (4,5,6)
4	**Generate an enemy posture report**	
	Obtain enemy position in the battlefield report	Triangular (4,5,6)
	Assess and evaluate the enemy positions	Triangular (16,20,24)
	Generate the data sheet for report	Triangular (8,10,12)
	Generate an enemy posture report	Triangular (4,5,6)
5	**Perform target development/prioritization**	
	Obtain enemy posture report	Triangular (4,5,6)
	Mark the targets	Triangular (8,10,12)
	Prioritize the targets	Triangular (12,15,18)
	Generate target development/prioritization report	Triangular (4,5,6)
6	**Perform aimpoint construction**	
	Analyze target report	Triangular (8,10,12)
	Construct the aimpoint	Triangular (8,10,12)
	Generate the aimpoint data	Triangular (4,5,6)
	Produce aimpoint report	Triangular (4,5,6)
7	**Perform weaponeering**	
	Obtain target report	Triangular (4,5,6)
	Obtain aimpoint report	Triangular (4,5,6)
	Perform weapon selection	Triangular (12,15,18)

Table 14. Continued

	Produce weaponeering report	Triangular (4,5,6)
8	**Evaluate air defense capability in the area of interest**	
	Define battlefield environment	Triangular (16,20,24)
	Describe the battlefield's effect	Triangular (16,20,24)
	Determine threat courses of actions	Triangular (20,25,30)
	Produce air defense capability report	Triangular (4,5,6)
9	**Forecasts the degree of redundancy necessary for the objective**	
	Obtain air defense capability report	Triangular (4,5,6)
	Obtain aimpoint report	Triangular (4,5,6)
	Calculate degree of redundancy necessary for the objective	Triangular (28,35,42)
	Produce degree of redundancy report	Triangular (4,5,6)
10	**Produce mission plan**	
	Obtain all the reports	Triangular (8,10,12)
	Check the reports	Triangular (8,10,12)
	Combine the reports	Triangular (16,20,24)
	Produce mission plan	Triangular (4,5,6)

4. Enter Workload Values and Workload Constraints:

Table 15 shows the VACP values of each tasks belongs to these tasks. Those VACP values of each task are gathered from the study of Hunn, Schweitzer, Cahir, and Finch, (2008) that used the same VACP workload estimation scales in Table 10. Since the tasks used in their study are close enough to Air Interdiction Planning Mission tasks, no additional workload estimation procedure is used.

Table 15. Visual, Auditory, Cognitive and Psychomotor Workload Values of Each Task (their summation gives the total workload level)

	Tasks	Workload			
		Visual	Auditory	Cognitive	Psychomotor
1	**Analyze request**				
	Receive a request for CAS or intelligence report	3	0	1	2.2
	Read the request	5.1	0	0	0
	Evaluate the threat of the enemy forces in the battlefield area	3	0	4.6	0
	Produce a threat report	4.4	0	1	6.5
2	**Produce threat characteristic data**				
	Evaluate the threat report	3	0	4.6	0
	Weight the threats likelihood	3	0	1.2	0
	Generate threat characteristics data	3	0	1	2.2
	Produce threat characteristic data report	0	0	0	2.2
3	**Get enemy data**				
	Receive enemy position data	3	0	1	2.2
	Assess and evaluate the enemy position in the battlefield	3	0	4.6	0
	Generate the data sheet	3	0	1	2.2
	Produce the enemy position in the battlefield report	0	0	0	2.2
4	**Generate an enemy posture report**				
	Obtain enemy position in the battlefield report	3	0	1	2.2
	Assess and evaluate the enemy positions	4	0	4.6	2.2
	Generate the data sheet for report	3	0	1	2.2
	Generate an enemy posture report	0	0	0	2.2
5	**Perform target development/prioritization**				
	Obtain enemy posture report	3	0	1	2.2
	Mark the targets	4	0	1.2	2.2
	Prioritize the targets	3	0	6.8	2.2
	Generate target development/prioritization report	0	0	0	2.2
6	**Perform aimpoint construction**				
	Analyze target report	4.4	0	4.6	0

Table 15. Continued

	Construct the aimpoint	6	0	7	0
	Generate the aimpoint data	3	0	0	2.2
	Produce aimpoint report	0	0	0	2.2
7	**Perform weaponeering**				
	Obtain target report	3	0	1	2.2
	Obtain aimpoint report	3	0	1	2.2
	Perform weapon selection	3	0	6.8	2.2
	Produce weaponeering report	0	0	0	2.2
8	**Evaluate air defense capability in the area of interest**				
	Define battlefield environment	6	0	6.8	2.2
	Describe the battlefield's effect	4.4	0	1	6.5
	Determine threat courses of actions	0	0	6.8	6.5
	Produce air defense capability report	4.4	0	1	6.5
9	**Forecasts the degree of redundancy necessary for the objective**				
	Obtain air defense capability report	3	0	1	2.2
	Obtain aimpoint report	3	0	1	2.2
	Calculate degree of redundancy necessary for the objective	3	0	7	2.2
	Produce degree of redundancy report	0	0	0	2.2
10	**Produce mission plan**				
	Obtain all the reports	3	0	1	2.2
	Check the reports	3	0	6.8	2.2
	Combine the reports	3	0	1	2.2
	Produce mission plan	3	0	6.8	6.5

The workload threshold value is set to 28. It means that the instantaneous workload of an agent cannot exceed 28.

5. Define operational stress/tempo:

Operational tempo for this Air Interdiction Planning Mission defines the frequency of inter-arrival time of requests. Various levels of operational tempo will be used to test the model under varying conditions. As the inter-arrival time of work orders decreases, the operational tempo increases. There is a discussion of the detailed information about operational tempo and its effects on the performance output in section 5.1.

6. Assign Tasks to Agents:

Agents are assigned to the tasks based on the optimization engine's outputs.

7. Run the Model and Obtain the Performance Results:

The simulation run is started through the optimization engine once the task-agent pair suggestions are ready. The timeliness (average duration) of the process is calculated by this stochastic simulation model.

4.1.2.2 Basic Analysis for the Simulation Model

Validation and Verification: Before the simulation process developed as part of the simulation-optimization methodology can be used, the validation and verification of this hypothetical model must be completed. Since, it cannot be compared to an actual system, after developing the simulation model; it is evaluated through face validation. The entire predecessor, successor relationships are checked and compared to similar work processes in literature (Perdu, 1997). The assigned task delays and workload components were based on similarly modeled military tasks (Mitchell, 2007). In this study, since the work processes serve as a surrogate to test the simulation-optimization modeling methodology and the hypothesis of improved performance, validating that the simulation model properly captures the work process characteristics is sufficient.

The model verification is performed by using sensitivity analysis to study change in the input, which causes the change in the output correspondingly. Input values were modified to check for the simulation responses as a way to confirm the accuracy of the model implementation. The verification method changes the number of work orders, increasing in one and decreasing in another. Results from the variation are then compared to the baseline simulation.

In the original simulation, the number of work orders is 30. This amount is doubled and halved all the while keeping the inter-arrival time of work orders randomly distributed at 50 seconds. Table 15 illustrates the results. One can observe that by increasing the number of work orders, the waiting time of work orders, average duration of the work process increases. The reverse is true for decreasing the number of work orders. For instance, when the number of work orders doubled, average waiting time of work processes in queues increased by 38% and average duration of work process increased by 13%. In the reverse case average waiting time of work processes in queues decreased by 34% and average duration of work process decreased by 12%. One can see that these percentages in decreases and increases in waiting time and average duration are similar. The small variation between them can be explained by the stochasticity of the simulation model. For example, see the percentage of decrease and increase in average duration when the number of work orders doubled and halved. The increase is 13% and, the decrease is 12%. The variation is as small as 1%.

Table 16. Sensitivity Analysis

	Original Number of Work Orders (30)	Double Number of Work Orders (60)	Half Number of Work Orders (15)
Average waiting time of work orders in queues (sec)	364.89	503.08	241.59
	Original	Increase 38%	Decrease 34%
Average duration of the work process (sec)	529.02	597.85	467.26
	Original	Increase 13%	Decrease 12%

Since the outcomes of the model are reasonable responses to the variations, the model meets the sensitivity analysis requirements.

Termination condition of simulation: Note that Air Interdiction Planning Mission is a terminating system since the team is only working during their shift, and the problem is focuses on a specific period. Therefore, the simulation does not require a warm-up period. The simulation terminates once all the defined number of work orders (30) are completed.

Setting the number of simulation replications: To implement a valid analysis of the simulation model, the number of simulation replications must be determined. The simulation model was executed for five runs using common random numbers under low operational tempo with random task-agent assignments to obtain n_0. The number of work orders (30) defines the run length of each replication. The average duration of the work process obtained is 246.07. The standard deviation (s) is 10.15. According to these results, the half-width of the 95% confidence interval (h_0) is $t_{n-1,1-\alpha/2}\frac{s}{\sqrt{n_0}}$= 11.64. In order to reduce half-width to 5 (h), an approximate required sample size would be $n = z^2_{1-\alpha/2}\frac{s^2}{h^2}$= 15.8. As a result, the simulation model is set at 15 runs to evaluate the fitness value of each chromosome. At this point, the optimization engine and the simulation model (validated, verified, termination condition and number of replications defined) are ready to be integrated in order to complete the simulation optimization model.

4.1.3 Integrating Simulation Model and Optimization Engine, Stopping Criteria of Simulation Optimization, and GA Parameter Setting for "Air Interdiction Planning Mission" Process

Integrating Simulation Model and Optimization Engine: Once both modules are developed, the two-way communication is implemented by integrating the text files.

Stopping Criteria: For the stopping condition, a combination of convergence factor and number of iterations is used. To ensure convergence, GA runs for a predetermined number of generations while progress is monitored using graphs. Once the solutions remain static for 100 generations, it terminates. In case it does not converge, after 1000 iterations it terminates. As mentioned, although GA is a generally applicable meta-heuristic, the crossover rate, mutation rate and population size parameters need to be tuned to suit the problem on hand.

Tuning the GA Parameters (Population Size, Cross-over Rate and Mutation Rate): In general, the population size is advised to be four times bigger than the chromosome size (Daniel & Rajendran, 2005). Since, the chromosome size is 10 the smallest population size is set to be 40 following the population size set to 50 and 60, as well. Moreover, to identify the best set of population size (PS), cross-over rate (CR) and mutation rate (MR) across four different work process settings, two measures, namely relative increase in average duration and average relative increase in average duration are calculated for all sets of PS, CR and MR. The work process settings considered are called W1, W2, W3, and W4. W1 is the air interdiction planning mission work process with randomly distributed inter-arrival time of work orders of 100 seconds. W2 is the air interdiction planning mission work process with randomly distributed inter-arrival time of work orders of 75 seconds. W3 is the air interdiction planning mission work process with randomly distributed inter-arrival time of work orders of 50 seconds. W4 is the air interdiction planning mission work process with randomly distributed inter-arrival time of work orders of 25 seconds. Note that as the inter-arrival time of work orders decreases, operational tempo increases. Decrease in inter-arrival time of work orders means that the frequency of work orders increases. The inter-arrival of work orders with 100, 75, 50, and 25 are selected to test the changes in average duration with respect to varying PS, MR, and CR.

The formula used for relative increase in average duration for given PS, CR and MR:

$$Relative\ Increase\ in\ Average\ Duration =$$

$$\frac{Average\ Duration_{\{PS;CR;MR\}} - \min Average\ Duration_{\{PS;CR;MR\}}}{\min Average\ Duration_{\{PS;CR;MR\}}}\ x\ 100$$

To explain the measure, consider the work process W1 with inter-arrival time of work orders of 100 seconds (randomly distributed) and the PS, CR and MR setting {40; 0.5, 0.01}. The average duration of the work process corresponding to this set of PS, CR, and MR is 307.98 seconds (see Table 16). The minimum average duration obtained through different set of PS, CR and MR for W1 is 244.87 (see Table 15). Hence the relative increase in average duration for the setting of {40; 0.5, 0.01} with respect to work process W1 is computed as 63.11 (i.e. (307.98-244.87) *100/244.87) (see Table 17). Similarly, the relative increase in average duration is computed with respect to all work processes settings (W1, W2, W3, and W4) and PS, CR, and MR settings. The average relative increase in average duration for a given setting of CR and MR across different operational tempo setting is calculated as follows:

$$Average\ Relative\ Increase\ in\ Average\ Duration = \frac{Sum\ of\ relative\ increases}{4}$$

Table 17. Results (Average Duration) from the Simulation Optimization Based on the Given PS, CR and MR

{PS; CR; MR}	Work Process settings with varying inter-arrival time of work orders			
	W1 (inter arrival time of work orders100 sec. randomly distributed)	W2 (inter arrival time of work orders 75 sec. randomly distributed)	W3 (inter arrival time of work orders 50 sec. randomly distributed)	W4 (inter arrival time of work orders 25 sec. randomly distributed)
40 0.5 0.01	307.98	274.53	366.70	762.11
40 0.5 0.02	334.84	287.30	350.05	663.72
40 0.5 0.03	285.39	309.43	763.67	635.22
40 0.6 0.01	296.90	286.37	857.04	658.65
40 0.6 0.02	255.64	282.21	354.75	624.12
40 0.6 0.03	251.30	269.64	325.18	1512.94
40 0.7 0.01	396.57	372.91	309.31	680.03
40 0.7 0.02	266.11	315.70	381.55	620.53
40 0.7 0.03	352.06	548.84	363.75	784.17
40 0.8 0.01	260.46	252.77	352.94	621.16
40 0.8 0.02	244.87	285.99	328.00	725.69
40 0.8 0.03	274.50	267.39	1003.91	686.80
40 0.9 0.01	255.90	283.77	837.40	671.21
40 0.9 0.02	252.03	290.59	312.61	709.33
40 0.9 0.03	271.20	257.99	566.89	1436.98
50 0.5 0.01	534.98	264.46	377.29	773.27
50 0.5 0.02	335.23	291.14	451.00	710.04
50 0.5 0.03	281.48	271.55	379.38	677.93
50 0.6 0.01	333.80	271.55	332.33	1072.81
50 0.6 0.02	282.54	318.26	361.49	703.14
50 0.6 0.03	282.60	339.20	633.94	760.97
50 0.7 0.01	278.49	328.77	867.31	615.82
50 0.7 0.02	266.69	408.67	390.13	638.33
50 0.7 0.03	264.63	287.15	392.41	850.75
50 0.8 0.01	255.47	269.25	780.76	791.99
50 0.8 0.02	291.74	318.78	324.23	923.76
50 0.8 0.03	258.11	337.79	355.84	616.68
50 0.9 0.01	254.77	450.14	360.65	744.79
50 0.9 0.02	304.38	546.85	608.55	690.40

Table 17. Continued

50 0.9 0.03	281.08	304.53	336.29	878.45
60 0.5 0.01	247.22	372.43	339.92	665.37
60 0.5 0.02	272.07	329.83	453.00	782.25
60 0.5 0.03	312.93	414.62	407.79	740.68
60 0.6 0.01	368.08	290.31	372.90	679.02
60 0.6 0.02	274.31	280.81	363.66	808.79
60 0.6 0.03	354.55	310.65	343.51	868.37
60 0.7 0.01	283.77	353.02	393.88	647.01
60 0.7 0.02	245.34	282.61	357.98	902.61
60 0.7 0.03	380.72	319.45	394.89	1073.35
60 0.8 0.01	277.52	340.75	629.14	744.03
60 0.8 0.02	287.34	397.54	841.99	906.29
60 0.8 0.03	298.56	290.92	389.68	991.73
60 0.9 0.01	293.07	335.11	350.02	663.86
60 0.9 0.02	270.88	448.53	413.00	640.09
60 0.9 0.03	297.30	321.28	341.30	771.49
Minimum	244.87	252.77	309.31	615.82

For example, considering the work process settings W1, W2, W3, and W4, the respective relative increases in average duration with the given PS, CR, and MR being {40; 0.5, 0.01} are 63.11, 21.76, 57.39, and 146.28, and the average relative increase in in average duration is 72.14 (see Table 18).

Table 18. Relative increase in average duration and average relative increase in average duration

{PS; CR; MR}	Work Process settings with varying inter-arrival time of work orders				Average relative increase in average duration
	W1 (inter arrival time of work orders100 sec. randomly distributed)	W2 (inter arrival time of work orders 75 sec. randomly distributed)	W3 (inter arrival time of work orders 50 sec. randomly distributed)	W4 (inter arrival time of work orders 25 sec. randomly distributed)	
40 0.5 0.01	63.11	21.76	57.39	146.28	72.14
40 0.5 0.02	89.97	34.53	40.74	47.90	53.29
40 0.5 0.03	40.52	56.66	454.36	19.40	142.73
40 0.6 0.01	52.03	33.60	547.73	42.82	169.05
40 0.6 0.02	10.78	29.44	45.44	8.30	23.49
40 0.6 0.03	6.43	16.87	15.88	897.11	234.07
40 0.7 0.01	151.70	120.14	0.00	64.21	84.01
40 0.7 0.02	21.25	62.93	72.24	4.70	40.28
40 0.7 0.03	107.19	296.07	54.45	168.34	156.51
40 0.8 0.01	15.59	0.00	43.63	5.34	16.14
40 0.8 0.02	0.00	33.22	18.70	109.87	40.45
40 0.8 0.03	29.63	14.62	694.60	70.98	202.46
40 0.9 0.01	11.03	31.00	528.09	55.38	156.38
40 0.9 0.02	7.16	37.82	3.30	93.50	35.44
40 0.9 0.03	26.33	5.22	257.58	821.15	277.57
50 0.5 0.01	290.11	11.69	67.98	157.45	131.81
50 0.5 0.02	90.36	38.37	141.69	94.21	91.16
50 0.5 0.03	36.62	18.78	70.07	62.11	46.89
50 0.6 0.01	88.94	18.78	23.02	456.99	146.93
50 0.6 0.02	37.67	65.49	52.18	87.31	60.66
50 0.6 0.03	37.74	86.44	324.63	145.15	148.49
50 0.7 0.01	33.63	76.00	558.00	0.00	166.91
50 0.7 0.02	21.82	155.91	80.82	22.50	70.26
50 0.7 0.03	19.76	34.38	83.10	234.93	93.04
50 0.8 0.01	10.60	16.48	471.45	176.16	168.68
50 0.8 0.02	46.87	66.01	14.92	307.93	108.93
50 0.8 0.03	13.24	85.02	46.54	0.85	36.41
50 0.9 0.01	9.90	197.37	51.34	128.97	96.90
50 0.9 0.02	59.52	294.08	299.24	74.57	181.85
50 0.9 0.03	36.21	51.76	26.98	262.63	94.39

Table 18. Continued

60 0.5 0.01	2.35	119.66	30.62	49.54	50.54
60 0.5 0.02	27.20	77.06	143.69	166.43	103.60
60 0.5 0.03	68.06	161.85	98.48	124.86	113.31
60 0.6 0.01	123.21	37.54	63.59	63.20	71.89
60 0.6 0.02	29.45	28.04	54.35	192.96	76.20
60 0.6 0.03	109.68	57.88	34.20	252.55	113.58
60 0.7 0.01	38.91	100.25	84.57	31.19	63.73
60 0.7 0.02	0.47	29.84	48.67	286.78	91.44
60 0.7 0.03	135.85	66.68	85.58	457.52	186.41
60 0.8 0.01	32.66	87.98	319.83	128.21	142.17
60 0.8 0.02	42.48	144.77	532.68	290.47	252.60
60 0.8 0.03	53.69	38.15	80.37	375.90	137.03
60 0.9 0.01	48.20	82.34	40.71	48.04	54.82
60 0.9 0.02	26.02	195.76	103.69	24.27	87.43
60 0.9 0.03	52.43	68.51	31.99	155.67	77.15
Minimum					16.14

Once the average relative increase in average duration is computed for all combinations of PS, CR and MR across different work process settings (W1, W2, W3, and W4), the set of PS, CR, and MR that results in minimum average relative increase in average duration is then selected. According to the results shown in Table 18 the lowest average relative increase is obtained from PS, CR, and MR setting of {40; 0.8; 0.01}. Consequently, PS, CR and MR are set to {40; 0.8; 0.01} for the current study.

At this point, the simulation optimization model is ready for the experiments. The next section will describe the alternative methods that will be used to compare the results of the simulation-optimization model; the results of the models will be discussed in the next chapter.

4.2. Alternative Methods to Compare with Developed GA Based Simulation-Optimization Method

In order to show the benefit of the GA based simulation-optimization method, the results will be compared to alternative methods. These common approaches are being used in organizations and other simulation-optimization techniques. The common approaches are called “Assignment based on capabilities” and “Assignment based on availability”. Those approaches represent the two extreme cases and the extreme bounds of the problem. The first is only taking into consideration the capability of employees while assigning them to tasks, while the second is only taking into consideration the availability of employees while assigning them to tasks. Moreover, a commercial simulation optimization package OptQuest is selected as an alternative tool to solve the current problem. The information and explanation on these methods are given in the next sub section.

4.2.1 Alternative Methods

4.2.1.1. Assignment methods used in common practices

Two different extreme methods have been used in organizations for assigning tasks to employees. One of these methods represents the extreme case where only capability is used as the assignment criteria. In order to refer this method easily, it is referred to as *assignment based on capabilities.* The other one represents the other extreme case where only availability of the employee is used as the assignment criteria. This method is referred to as *assignment based on availability.*

Note that there is no enough evidence in the literature for other methods or heuristics that represent these extreme cases, which are applicable to the current problem. For instance, the method that Eiselt and Marianov (2008) developed, takes the different skill levels and two different objectives into consideration in order to find the task-agent assignments. However, it is not possible to solve the problem in reasonable time with high number of agent and tasks.

Moreover, their method cannot be applied without being configured for the current problem. In addition, the mental workload availability analysis in the literature are limited with what-if simulation analysis.

Assignment based on capabilities (ABOC): In general, in a work process, the most skilled person is assigned to high number of tasks while the rest of the team is assigned to lower number of tasks (Daskal, 2016; Jackson, 2014; Parker, 2011). This has been a common assignment approach in various organizations and reported as a reason for low performance and unsatisfied employees (Schwartz & Erikson, 2009). In a low tempo environment, this assignment method is not likely to affect the overall task process performance; however, in higher tempo environments, the negative effects start to appear (such as increasing the duration).

The simulation is configured in order to represent this condition. First, the person with the highest capability levels is found. Then this person is assigned as many tasks that are equal or under his capability range. The rest of the tasks are assigned to the rest of employees according to their capability range; some of the employees may remain idle. The steps of the heuristic are as follows:

1. Find the agent with the highest total capability level.
2. Assign agent as many tasks as possible as long as each tasks' capability level is equal or lower than his capability levels.
3. Assign the rest of the Agents in a way that each skill level of the agent is equal or higher than the skill level of the task.
4. If there is no such case, assign agent to the task that is closer to his skill levels.

This assignment method represents the extreme case where only capability of agent is used as the assignment criteria.

Assignment based on the availability (ABOA): In this method, employees are assigned to the tasks based on their availabilities without taking their capabilities in to consideration. In other words, whoever has enough residual mental workload capacity for the next task is assigned to that task. The simulation model is arranged to represent this rule by adding the necessary decision module that shown in the flowchart in Figure 10 (adapted from the availability and assignment algorithm of Cook et al., 2012, pg. 87). Before each task, the current residual capacity of each agent is calculated. The first agent that has enough residual capacity for the following task assigned to the task. An agent is assigned randomly if there is no agent with enough current residual capacity.

This assignment method represents the other extreme case where only mental workload availability of agents is used as the assignment criteria.

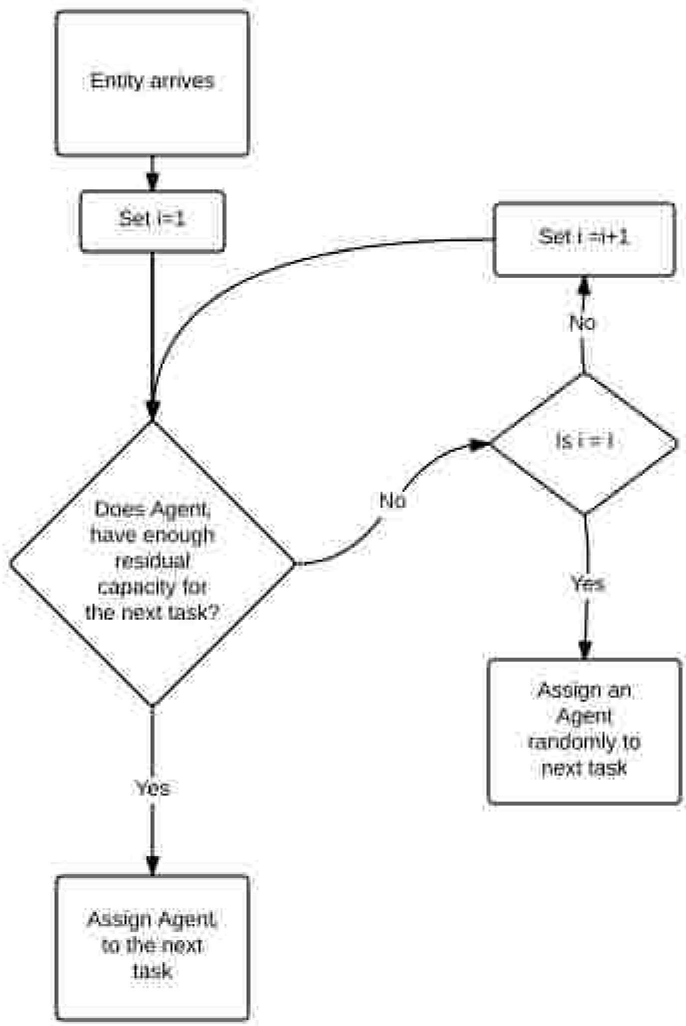


Figure 10. Flow chart for assignment based on availability rule

4.2.1.2. Other Simulation Optimization Methods

Simulation Optimization Package (OptQuest): In this study, a commercial optimization package, OptQuest, developed by OpTek, Inc. is selected to compare to the optimization model. The reason of selecting OptQuest is its compatibility with ARENA simulation software.

Developing identical simulation model with different software is a challenging task (Eskandari, Mahmoodi, Fallah, & Geiger, 2011). By using an optimization package that is compatible with ARENA, the same developed simulation model can be used to compare OptQuest with the developed optimization engine.

OptQuest employs three different search heuristics (Eskandari et al., 2011; Lipski, 2013; Wan & Kleijnen, 2006). Its main search strategy is scatter search (SS). SS applies heuristic processes to generate a starting set of solution vectors and designate a subset of best vectors to be reference solutions. Then the algorithm forms the linear combination of subsets of current reference points and generates new points. In the next step, the SS algorithm selects a combination of the best solutions and uses them as starting points for a new application of the heuristic processes and repeats these steps until a specified number of iteration or reaches stopping criteria. The secondary method that OptQuest uses is tabu search (TS). TS uses adaptive memory to prohibit the search from reinvestigating solutions that have already been evaluated and to guide the search to a globally optimal solution. As the last method, OptQuest employs neural networks (NN). A neural network is used to screen out solutions that are likely to be poor without allowing the simulation to evaluate them. The neural network is used as a prediction model to help the system accelerate the search by avoiding the need for evaluating objective function for a newly created reference point, in situations where the objective value can be predicted to be of low quality.

OptQuest uses three stopping rules: user-specified maximum number of configurations, automatic stop (run until there is no improvement in the value of the objective function for 100 consecutive configurations), and combination of both rules.

As described in literature review section, commercial simulation optimization tools are not developed to solve all types of problems and requires user sophistication. Note that OptQuest is not developed for solving assignment problems and configuring it to solve an assignment problem was a challenge.

In order to configure OptQuest for assignment problem, ARENA's capability of defining resource (agent) capacity as a variable is used. Instead of defining 10 agents, 100 (10x10) agents were created. For every one agent, 10 agents were created to represent which tasks the agent is assigned. The related variables (mental workload and capability) are changed accordingly in order to keep the same simulation logic.

In order to compare OptQuest solutions to the developed GA based simulation-optimization method, the same stopping rules are applied in both solution methods. In both methods the convergence factor is set to 100 which mean the methods run until there is no improvement in the value of the objective function for 100 consecutive configurations. The maximum number of configurations is set to 1000.

Complete Enumeration: Complete enumeration technique is another approach that can be used in order to find the optimum solution/solutions for the problem. In complete enumeration, all the possible solution combinations are enumerated and solved (Rao, 2009). Finally, the solution combination, which has the best objective function value, is selected.

Note that this method is not efficient or cannot be applied to large size combinatorial problems because the number of possible solution combinations will grow exponentially with respect to problem size. This method can only be applied to polynomial or small size combinatorial problems.

In the current case (Air Interdiction Planning Mission), because of the number of tasks and number of agents, the number of all combinations to be tested is 10,000,000,000 (10^{10}). Moreover, when taking into consideration the stochasticity of the problem, the number can be even higher. As a result, it is not possible to apply this method for the current case. However, it is mentioned in this section in order to illustrate the necessity of a reliable and fast search algorithm to solve the mentioned problem.

In summary, the simulation-optimization model is now ready for the experiments. Moreover, two common practices, assignment based on capability and assignment based on availability are introduced in order to make comparisons to developed method's performance. Additionally, in order to show the effectiveness of the developed model, another simulation optimization tool (a commercial one) is presented. In the next chapter, the results from the developed simulation optimization model with comparison to the alternative assignment methods will be introduced and discussed in detail.

5. NUMERICAL RESULTS AND COMPUTATIONAL PERFORMANCE

The developed GA based simulation-optimization model was executed through numerous configurations with respect to varying operational tempo and range of parameters. For simplicity, through the rest of the document the developed GA based simulation optimization model will be referred to as ABOGA, which is the abbreviation of assignment based on genetic algorithm.

The performance of the ABOGA approach is shown by comparing the computational results to the results obtained from applying the previously described common practices and solving the same problem with the commercial simulation optimization tool OptQuest.

Additionally, the effects of parameters namely *Capability Level Difference Coefficient* and *Number of Work Orders* on the average work process duration (performance output) is tested under varying operational tempo in order to further evaluate the performance of the assignment methods.

5.1. The preliminary results from common practice applications

In order to understand the system behavior of Air Interdiction Planning Mission example; firstly, the simulation models that reflect the common practices are run through various operational tempo (inter-arrival time of work orders) at a range of randomly distributed 10-350 seconds with 10 seconds increments. In doing so, the low, medium, and high operational tempo ranges are defined. The ranges in which the extreme common practices perform better than the other are understood. Defining the ranges by applying these common practices was necessary in order to identify the appropriate conditions in which to conduct further reasonable experiments. Once the necessary ranges are defined, developed ABOGA method is used to solve the problem in these ranges.

By following the steps explained in the definition of ABOC rule, the following agent-task assignments is used as an input for the simulation (recall for this case, the agent who meets or exceeds the required capability of the tasks is assigned):

Task	1	2	3	4	5	6	7	8	9	10
Agent	8	9	7	7	7	7	7	5	6	7

For the ABOA rule that has been described in Figure 10, it is applied before each task module initiates in the simulation model. Recall in this case, the agent that is available for the task based on residual mental capacity is assigned.

Figure 11 shows the change in average duration of the work process as the operational tempo increases for both of the assignment techniques. Between 350-150 seconds inter-arrival time of work orders range; there is not a substantial change in the average duration of the work processes under both ABOC and ABOA rules. During this range, the agents are not “stressed,” there is plenty of time between the arrival of work orders to complete the tasks and the likelihood of the need for multi-tasking is very low. However, after inter-arrival time of work orders of 150 seconds, the variations in average duration start to appear. Specifically, there is a slight increase in average duration between the 150-100 range. Between 50-10 range, there is a dramatic change in average duration especially under the ABOC method. Accordingly, the inter-arrival time of work orders 150-100 range is defined as low operational tempo, 90-50 range is defined as medium operational tempo, and 40-10 range is defined as high operational tempo (Figure 12). Further experiments will be conducted using these low, medium, and high operational tempo ranges.

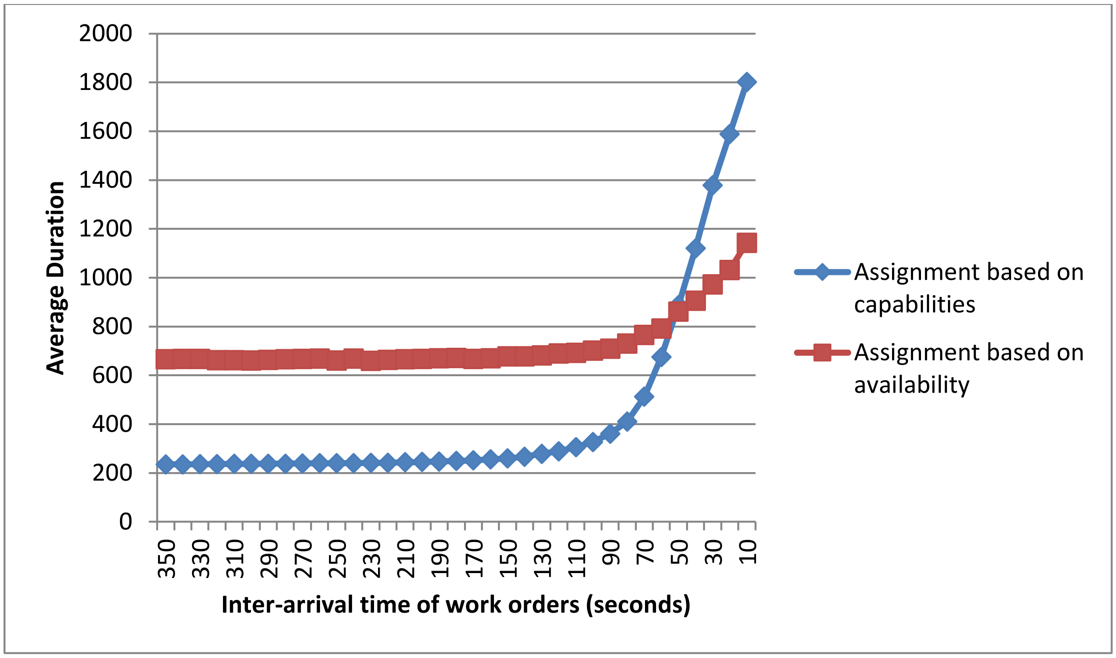


Figure 11. Average Duration Results (output) from Common Practices Under Varying Operational Tempo

Inter-arrival Time of Work Orders (sec.)	150	140	130	120	110	100	90	80	70	60	50	40	30	20	10
Operational Tempo	Low						Medium					High			

Figure 12. The Relationship Between Operational Tempo and Inter-arrival Time of Work Orders

According to these initial experiments utilizing common approaches (Figure 11); ABOC rule leads to better performance (lower average duration) during low operational tempo (150-100) while ABOA rule performs better as the operational tempo increases (50-10). Moreover, ABOC is less sensitive to operational tempo than ABOA.

In ABOC rule, even though all agents are capable enough to accomplish the tasks as fast as possible, since the most capable agent, which is agent 7, works on several tasks. As the operational tempo increases, more and more tasks start to waiting in the queue for the agent to become available. This situation ends up increasing the overall duration of the work process.

Generally, the ABOA rule is more consistent during the changes in operational tempo. Given no consideration for the agents' capabilities, it results in a higher average duration than ABOC rule during low and medium operational tempo.

As the results indicate, both assignment methods are noteworthy for the timeliness of the work process and should be taken in to consideration. As a result of this initial definition of the problem space, the developed method, ABOGA, will next be employed for the problem in the defined area of concern ranges (inter-arrival time of work orders of 150-10 seconds) in order to test the hypotheses of taking both capability and workload level of the agents into consideration for the assignment.

5.2. Results from developed GA based simulation-optimization method (ABOGA)

Now that the application of common practices has identified the area of concern, the proposed ABOGA method is applied to the problem for the 10-150 inter-arrival time of work orders range with 10 seconds of increments. The tuned population size, cross over rate and mutation rate (PS, CR, MR): (40, 0.8, 0.01) are used, as determined in the initial characterization of the model described in section 4.1.3.

Note that as the operational tempo changes, the solution space of the problem changes as well. When the operational tempo is lower (which means the inter-arrival time of work orders is higher), several number of solutions in the solution space are optimal or near optimal. As the operational tempo increases, the number of optimal or near optimal solutions gets lower and lower. As a result, solving the problem with different operational tempo can be considered as solving a new independent problem because the GA is searching for the solutions on a different solution space.

The GA starts immediately after the generation of the initial population, which is then send into the ARENA simulation model through a text file to begin the simulation executions. The average duration of Air Interdiction Planning Mission process obtained from the simulation executions becomes the fitness for each chromosome. After the crossover and mutation, offspring are obtained to replace some of the chromosomes in the mating pool. After the fitness values of chromosomes are evaluated, the results are analyzed to decide whether to stop or continue through the GA generations (Figure 6).

The results of the ABOGA can be found in Table 18 with task-agent assignments and number of generations GA used to find the solution. In case that the assignments showed in Table 18 are used for Air Interdiction Planning tasks and team, the work process would take the corresponding average durations without overloading any agent. For instance, in experiment 1, under low operational tempo with the inter-arrival time of work orders of 150 seconds, the agent task pairs should be

Task	1	2	3	4	5	6	7	8	9	10
Agent	8	7	10	1	7	3	9	5	6	7

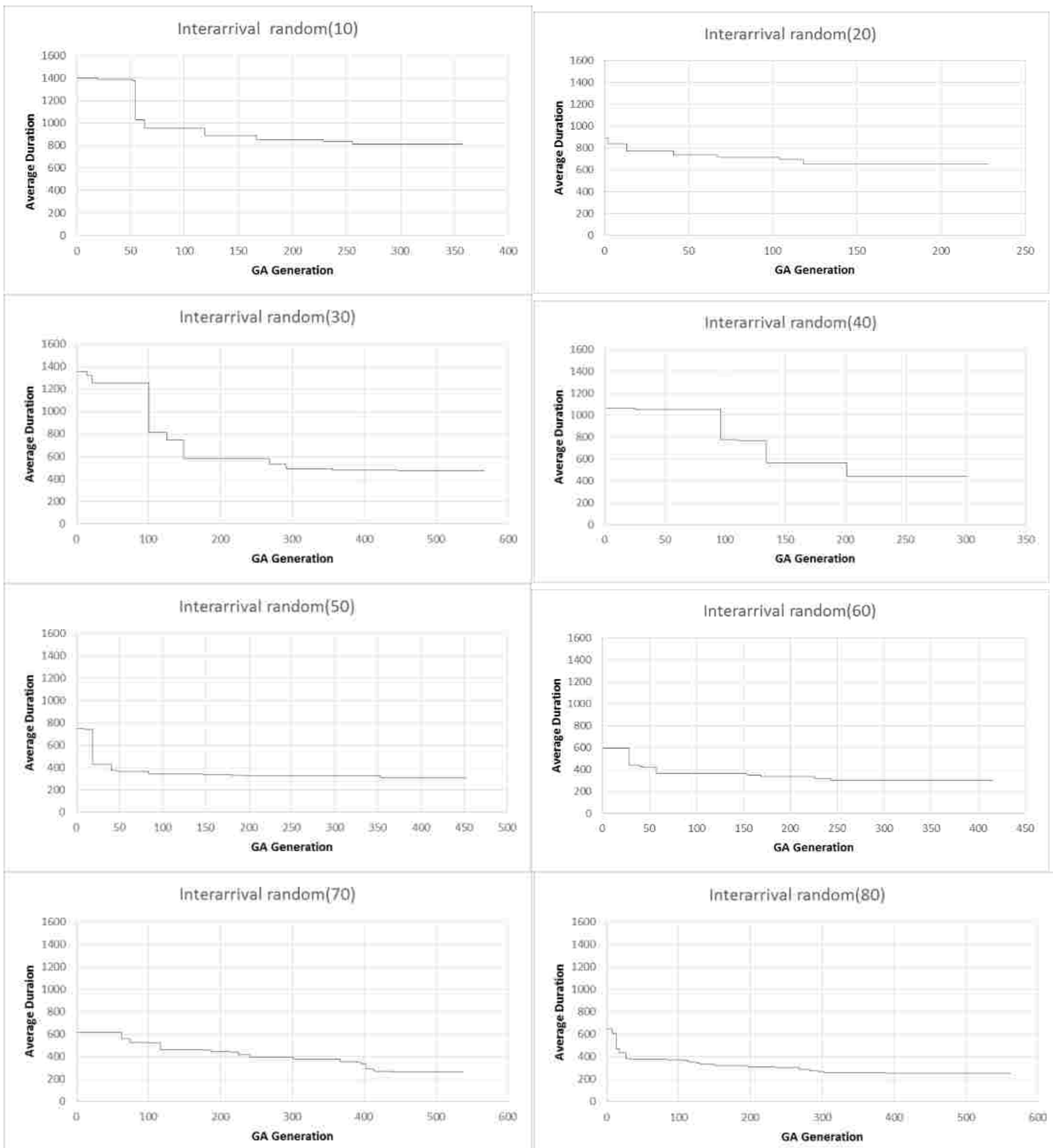
and the average duration would be 359 seconds for the team to get the air interdiction plan ready without making any errors, since no agent is overloaded.

The solutions show that the assignments are changing and the average durations are increasing gradually as the operational tempo increases. The substantial differences in average durations are seen during medium and, especially high operational tempo (Table 17, experiment numbers 9 to 15). The changes in the agent-task assignments is the results of ABOGA's search mechanism. As the operational tempo increases or decreases, different agent-task assignments lead to lower average work process durations. As a reminder, there are 10,000,000,000 possible agent-task assignments for this work process and it is not possible to evaluate all of them in order to decide which pair would lead to lowest duration. As a result, employing the ABOGA was necessary in order to find the agent-task pairs that leads to lower average durations under varying operational tempo while making sure that the agents are not overloaded. The solution evolution processes of GA with average duration and the best chromosomes under varying operational tempo are shown in Figure 13. It is important to point out that all the ABOGA experiments are converged (after 100 generations with no change on the best solution found so far) before completing the maximum number of generations allowed (1000). It took GA at least 59 and at most 552 generations to converge (Table 18, experiment numbers 2 and 3, respectively).

Table 19. Results from the Developed Simulation Optimization Approach Under Changing Operational Tempo

Experiment No	Operational Tempo	Inter-arrival time of work orders (in seconds and randomly distributed)	Average Duration (sec.)/ Timeliness	Agent Assignments to (Task 1, Task 2, …., Task 10)	# of GA generations to find the solution
1	Low	150	238.73685	8,7,10,1,7,3,9,5,6,7	359
2		140	262.94016	8,9,10,1,2,6,9,5,3,7	59
3		130	241.6398	8,4,7,7,2,3,3,5,6,7	552
4		120	242.21153	8,2,10,1,2,3,7,5,6,7	409
5		110	259.0673	8,2,10,7,3,7,5,5,6,7	464
6		100	251.13481	8,4,10,1,7,7,2,5,6,7	234
7	Medium	90	247.05888	8,2,10,1,2,3,4,5,6,7	301
8		80	251.41364	8,9,10,1,2,3,9,5,6,7	462
9		70	263.60068	8,9,10,1,2,7,9,5,6,7	438
10		60	301.40204	2,9,10,8,9,3,4,5,6,7	316
11		50	306.76413	8,9,10,7,2,3,2,5,6,7	353
12	High	40	440.90564	8,2,10,7,6,9,1,5,3,7	201
13		30	473.32428	4,9,10,1,9,3,8,5,6,7	467
14		20	649.88586	9,3,10,10,8,2,9,5,6,7	118
15		10	808.13715	8,10,10,7,2,4,9,5,6,7	258

As one can see some task-agent pairs are most consistent than others. These are Task 1- Agent 8, Task 3- Agent 10, Task 9- Agent 6 and Task 10- Agent 7. Task 1, Task 9 and Task 10 are some of the tasks that required high capability levels (Table 11). Agent 8, Agent 6 and Agent 7 are some of the agents that have high capability levels (Table 12). Accordingly, the GA finds the solutions that include these assignments in order to decrease the average duration. On the other hand, Task 3 is one of the tasks that require low capability levels and Agent 10 is one of the agents that have low capability levels. As a result, GA finds the solutions that include this assignment to avoid Agent 10 to work on a task require higher capability levels. The rest of the agent-task assignments are changing as the operational tempo changes in order to find the pairs that lead to lowest average duration of the work process.



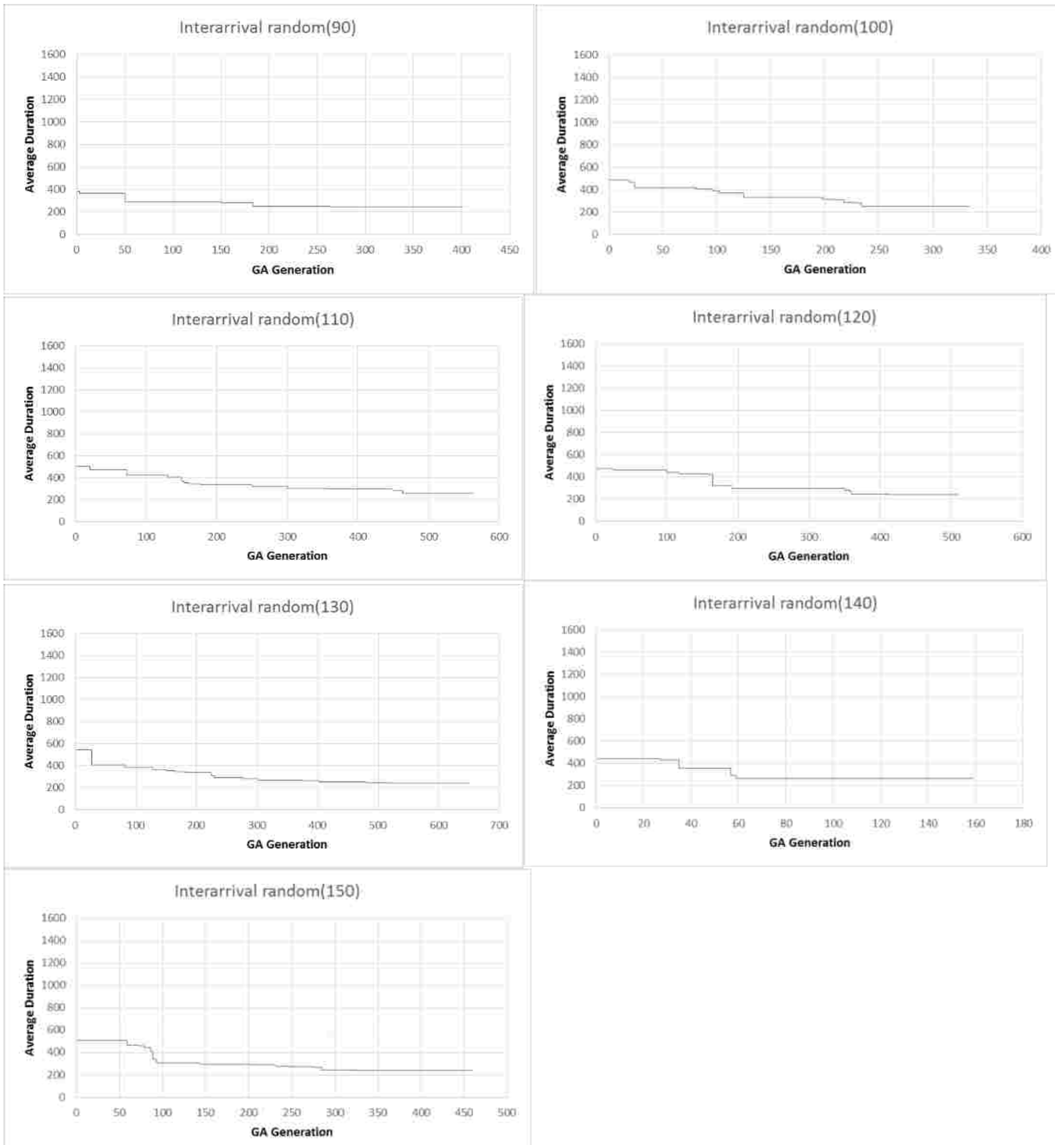


Figure 13. Evolution process and convergence of solutions in the optimization engine (GA)

The longest ABOGA run took approximately 36 hours (Table 18, experiment No 3). Evidence notes that there are several components that affect the computation time. These are the computing environment, initial solution, and simulation run time. Even though the common random number technique is used in order to decrease variability in the output of the simulation replications, 15 replications were decided to appropriate to obtain meaningful results. The significant amount of time in this process is spent in running the simulation experiments.

In order to employ the simulation-optimization on a work process the user should have estimation on the operational tempo of the work process. A work process does not have to have one single operational tempo level. The operational tempo can change according to the time of the day or the occurrence of an event (such as the busiest hours in emergency rooms). In that case, the simulation optimization can be run using the expected operational tempo levels in order to find assignments to minimize the average duration of the work process.

It is shown that the ABOGA method searches the solution space for the right agent-task assignment in order to minimize the work process duration. Further observations on the effectiveness of ABOGA method will be discussed in the rest of the chapter by making comparison with the results obtained from the common practices and OptQuest tool.

5.3. Comparison of the results from ABOGA to common practices

In order to show the effectiveness of the developed ABOGA method, the results that were obtained from ABOGA method are compared to the results that are obtained from common practices.

The results shown in section 5.1 indicated that ABOC leads to lower average duration during low and medium operational tempo and ABOA leads to lower average duration during high operational tempo. However, both variables, capability and mental workload of the agents are important for the duration of the work process. Even though these common practices have operational tempo ranges that they work better under; when compared to the ABOGA results, there are agent-task pairs that lead to better solutions than the common practices under any operational tempo.

Figure 14 shows the results of average duration of Air Interdiction Planning Mission under varying operational tempo from both common practices (ABOC and ABOA) and ABOGA method. In low operational tempo, the solutions found by ABOGA is better but very close to solutions found by ABOC. However, after the point of 100 seconds (randomly distributed) inter-arrival time of work orders, the average duration resulted from ABOC starts to increase gradually. The reason is that the ABOC assigns numerous tasks to the most capable agent, as the inter-arrival time increases, the number of tasks waiting for that particular agent to be available increases. ABOGA method solves this issue and finds the right assignments without over loading one particular agent. Conversely, even though, ABOA takes the workload of the agent in to consideration, it does not consider the capability levels and consequently assigns agents to tasks with unmatched capabilities. As a result, it results in higher average duration than the ABOGA. From the graph in Figure 14, one can see the considerable difference in the average duration results from common practices and the ABOGA method. Especially, during medium and high operational tempo, this difference gap is increases dramatically. ABOGA method finds the task-agent assignments that lead to lower average duration of work process than ABOC and ABOA methods.

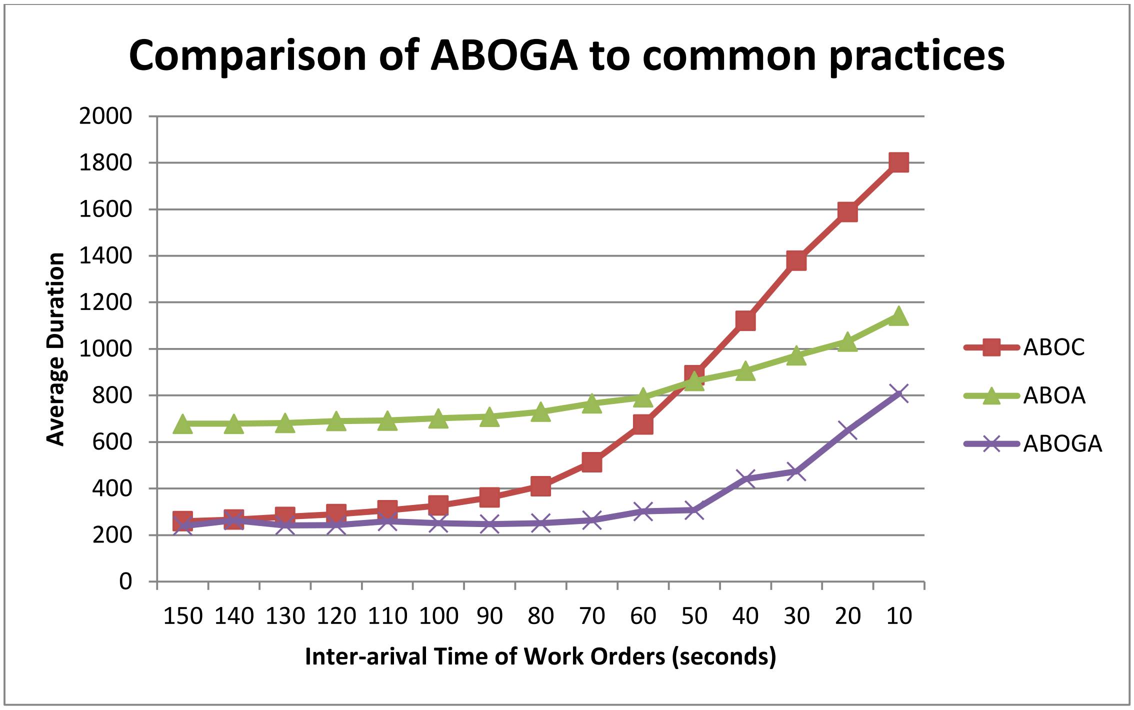


Figure 14. Comparison between common approaches (ABOA and ABOC) and ABOGA

Table 20 shows the percent difference of average duration found between common practices and ABOGA method. Percent difference between ABOC and ABOGA method is very low during low operational tempo (as low as 1.43% in experiment 2), while the differences between ABOA and ABOGA goes as high as 184.83% (experiment 4). In contrast, during high operational tempo, differences between ABOC and ABOGA is as high as 191.29% (experiment 13), and differences between ABOA and ABOGA is as high as 105.52% (experiment 12) and as low as 41.48% (experiment 15). Moreover, under any operational tempo neither ABOC nor ABOA method could find an assignment that leads lower average duration than ABOGA method. ABOGA method found the assignments that lead to lowest average duration for Air Interdiction Planning Mission under every operational tempo setting.

Table 20. Percent Difference Between ABOC, ABOA and Developed ABOGA Average Duration Results

Experiment No	Operational Tempo	Inter arrival time of work orders (sec., randomly distributed)	Average duration from ABOC assignments	Average duration from ABOA assignments	Average duration from ABOGA assignments	%difference between ABOC and ABOGA	%difference between ABOA and ABOGA
1	Low	150	259.42	678.06	238.74	8.66	184.02
2		140	266.71	678.03	262.94	1.43	157.86
3		130	278.36	681.7	241.64	15.20	182.11
4		120	289.42	689.89	242.21	19.49	184.83
5		110	306.28	692.1	259.07	18.22	167.15
6		100	327.18	701.35	251.13	30.28	179.27
7	Medium	90	361.24	708.84	247.06	46.22	186.91
8		80	410.27	729.92	251.41	63.19	190.33
9		70	512.49	765.8	263.60	94.42	190.52
10		60	675.37	791.78	301.40	124.08	162.70
11		50	886.82	861.51	306.76	189.09	180.84
12	High	40	1120.7	906.14	440.91	154.18	105.52
13		30	1378.74	972	473.32	191.29	105.36
14		20	1588.37	1031.54	649.89	144.41	58.73
15		10	1801.39	1143.38	808.14	122.91	41.48

These results confirm that assignment based on capability level and assignment based on workload levels both have importance on the performance of the work process. When both issues are taken into consideration together, by finding the right task-agent assignments, the performance of the work process increases. By using the developed ABOGA method, one can find the task-agent pairs that result in lower average duration of Air Interdiction Planning Mission process than ABOC and ABOA method.

5.4. Comparison of results from developed ABOGA to OptQuest

OptQuest uses a combination of scatter search, tabu search and neural networks algorithms in order to find a good, satisfactory or best solution. The developed ABOGA uses a special genetic algorithm that tuned for this problem. Since the search algorithms are different in these two methods, in order to compare them, the same rules are applied as closely as feasible. The following points summarize the similar rules used in both approaches:

- The same simulation model developed in ARENA simulation software is used.
- Same number of simulation replications (15) are used in order to evaluate a candidate solution.
- The same stopping criteria are used. Convergence rate is set to 100 in both algorithms. The maximum number of generation is set to 1000.

The results to the problem obtained from OptQuest can be found in Table 21 with agent-task assignments and number of solutions generated in order to find the lowest average work process durations. Note that all the OptQuest experiments are also converged without completing the maximum allowed generations. The highest generations that OptQuest has taken in order to converge was 600 (experiment 15). The consistency of some agent-task pairs such as Task 1-Agent 8 and Task 10- Agent 7 can be seen in OptQuest results as well.

Table 21. Results from OptQuest Under Varying Operational Tempo

Experiment No	Operational Tempo	Inter-arrival time of work orders (in seconds and randomly distributed)	Average Duration (sec.)/ Timeliness	Agent Assignments to (Task 1, Task 2, …., Task 10)	# of OptQuest generations to find the solution
1	Low	150	238.72	8,9,7,1,7,3,3,5,6,7	346
2		140	287.92	9,9,7,1,2,2,3,5,7,7	391
3		130	292.28	8,5,7,1,2,9,5,3,6,7	464
4		120	243.03	8,2,7,1,2,7,2,5,6,7	503
5		110	328.33	6,9,3,1,4,3,2,5,3,7	366
6		100	315.18	8,9,7,1,8,2,9,3,6,7	446
7	Medium	90	278.77	8,9,10,1,4,7,6,5,3,7	354
8		80	290.6	8,9,10,1,8,3,4,5,3,7	498
9		70	385.02	6,2,7,3,4,9,4,5,6,7	538
10		60	378.15	8,9,3,1,10,3,4,5,2,7	529
11		50	599.85	8,2,10,1,5,5,8,3,6,7	464
12	High	40	384.54	8,2,7,1,2,9,2,5,3,7	387
13		30	852.39	9,1,7,1,10,8,9,3,6,7	440
14		20	807.11	2,4,7,9,4,2,7,5,6,7	494
15		10	992.17	8,7,7,2,10,4,7,5,6,7	600

The percent difference between developed ABOGA to OptQuest algorithm is shown in Table 22. While the difference is as low as 0 and 0.003% in low operational tempo, this difference increases to 95% in higher operational tempo levels. In most cases (except experiment no 1 and 4) the GA based algorithm (ABOGA) performs better than OptQuest algorithm under the same rules (number simulation replications are 15, convergence rate is set to 100, and maximum number of generation is set to 1000). Moreover, the randomness (the fluctuation) in OptQuest output is higher than the ABOGA output, which makes ABOGA more reliable (Figure 15).

Table 22. Percent Difference Between Developed ABOGA Method Results to OptQuest Results

Experiment No	Operational Tempo	Inter arrival time of work orders (sec., randomly distributed)	Average duration from ABOGA assignments	Average duration from OptQuest assignments	% Difference
1	Low	150	238.73685	238.72	0
2		140	262.94016	287.92	9.5002
3		130	241.6398	292.28	20.9569
4		120	242.21153	243.03	0.337915
5		110	259.0673	328.33	26.73541
6		100	251.13481	315.18	25.50231
7	Medium	90	247.05888	278.77	12.83545
8		80	251.41364	290.6	15.58641
9		70	263.60068	385.02	46.06184
10		60	301.40204	378.15	25.46365
11		50	306.76413	599.85	95.54111
12	High	40	440.90564	500.54	13.52542
13		30	473.32428	852.39	80.08584
14		20	649.88586	807.11	24.19258
15		10	808.13715	992.17	22.77248

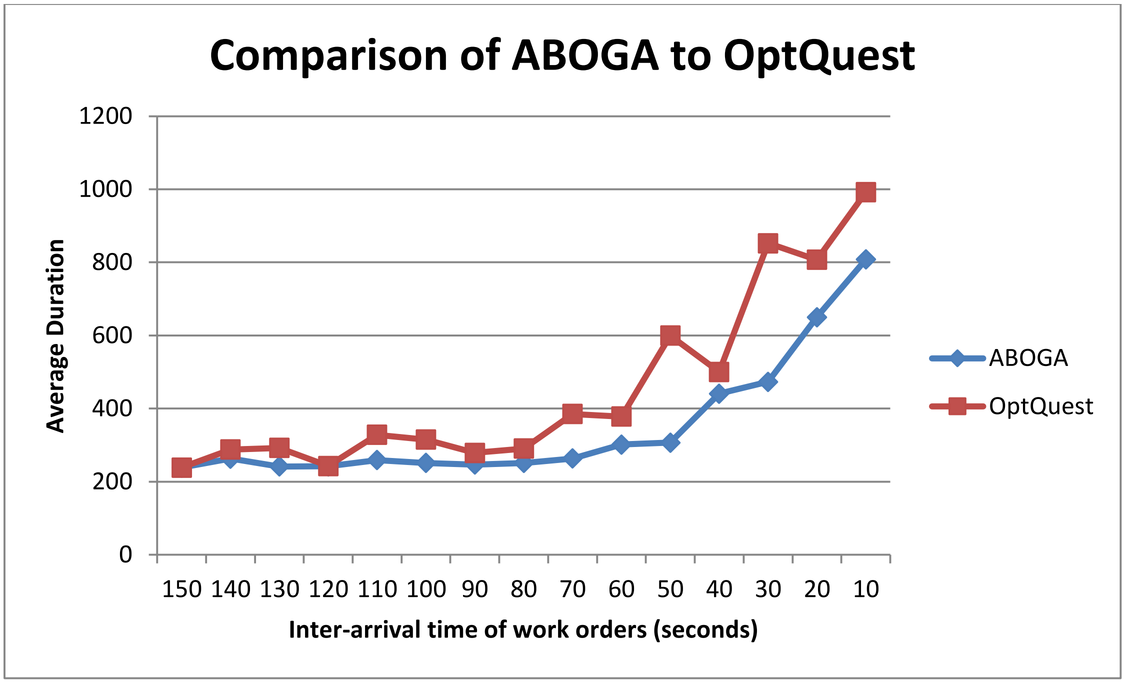


Figure 15. Comparison of developed ABOGA results to OptQuest results

Note that the OptQuest results are better than common practices' results during medium and high operational tempo. In low operational tempo, there are situations that ABOC method results are better than OptQuest results (Table 23 experiment no 2, 3, and 5). The percent difference between OptQuest and common practices results increases as the operational tempo increases. While OptQuest leads better results than common practices in general (except in low operational tempo); overall, it has been shown that the developed ABOGA method leads better results than all approaches (Figure16).

Table 23. Percent Difference Between OptQuest Results to Common Practice Heuristics Results

Experiment No	Operational Tempo	Inter arrival time of work orders (sec., randomly distributed)	Average duration from ABOC assignments	Average duration from ABOA assignments	Average duration from OptQuest assignments	% difference between ABOC to OptQuest	% difference ABOA to OptQuest
1	Low	150	259.42	678.06	238.72	8.67	184.04
2		140	266.71	678.03	287.92	-7.37	135.49
3		130	278.36	681.7	292.28	-4.76	133.24
4		120	289.42	689.89	243.03	19.09	183.87
5		110	306.28	692.1	328.33	-6.72	110.79
6		100	327.18	701.35	315.18	3.81	122.52
7	Medium	90	361.24	708.84	278.77	29.58	154.27
8		80	410.27	729.92	290.6	41.18	151.18
9		70	512.49	765.8	385.02	33.11	98.90
10		60	675.37	791.78	378.15	78.60	109.38
11		50	886.82	861.51	599.85	47.84	43.62
12	High	40	1120.7	906.14	500.54	123.90	81.03
13		30	1378.74	972	852.39	61.75	14.03
14		20	1588.37	1031.54	807.11	96.80	27.81
15		10	1801.39	1143.38	992.17	81.56	15.24

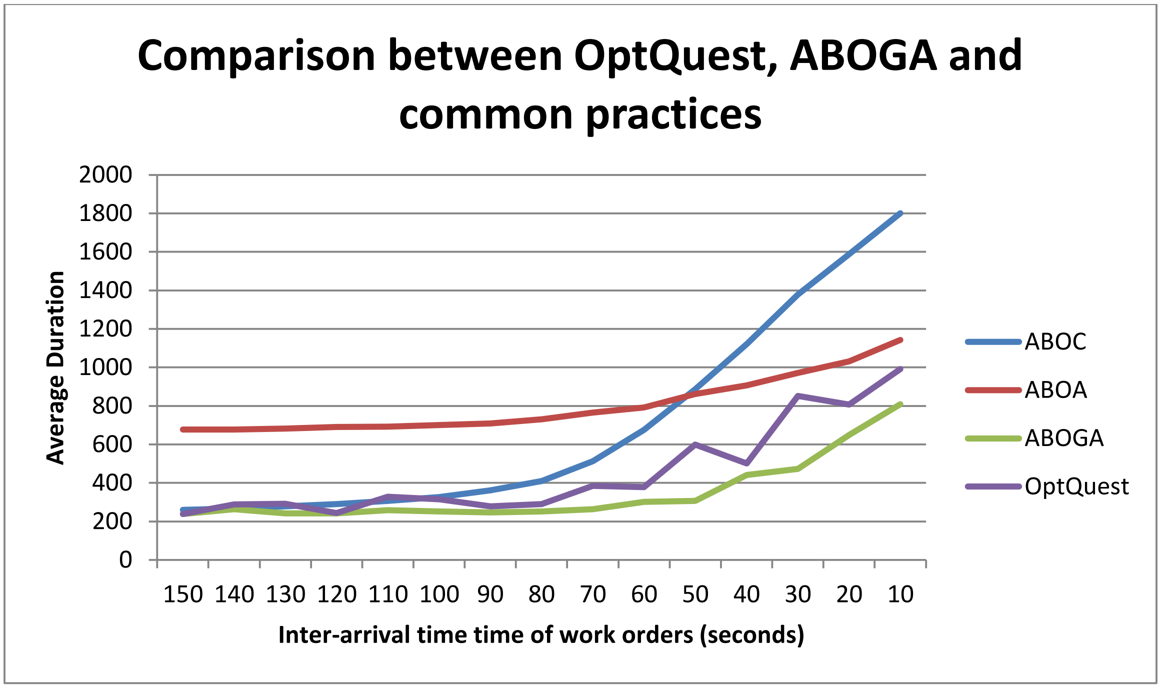


Figure 16. Comparison Between Developed ABOGA Approach, OptQuest and Common Practices

5.5. Influences of other parameters

The effect of capability level difference coefficient

Capability level difference coefficient emphasizes the importance of the difference between capability level of agents and required capability level from tasks on the task durations.

For the ABOA method, as the capability level difference coefficient increases the average duration of the work process increases (Figure 17). On the other hand, as seen in Figure 18, the average duration of the Air Interdiction Planning Mission results from ABOC methods does stay the same as the capability level coefficient changes. ABOGA method is sensitive to capability level difference coefficient only on high operational tempo (Figure 19). In ABOGA results, one can see the increase in the average duration during high operational tempo (Inter-arrival time of work orders 10 and 30) as the capability level coefficient increases.

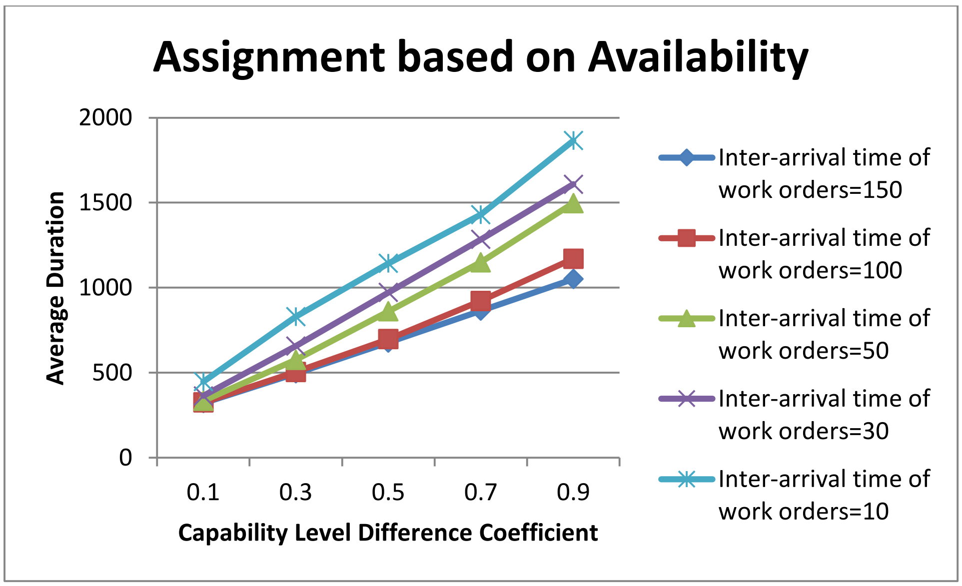


Figure 17. Results of ABOA with Varying Capability Difference Coefficient

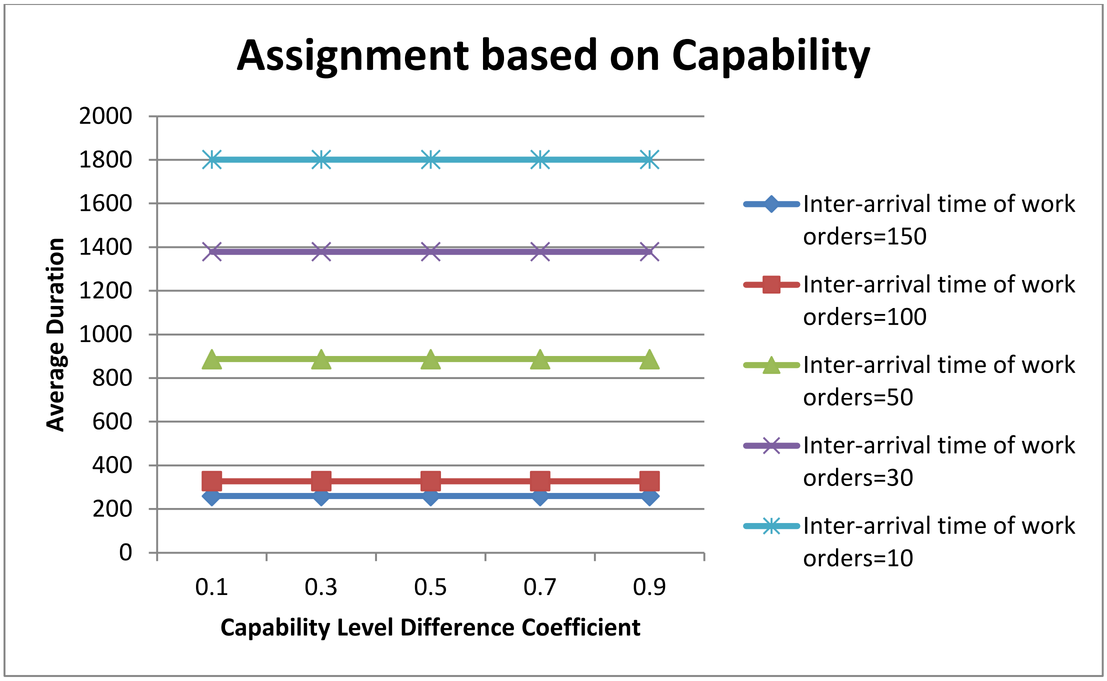


Figure 18. Results of ABOC with Varying Capability Difference Coefficient

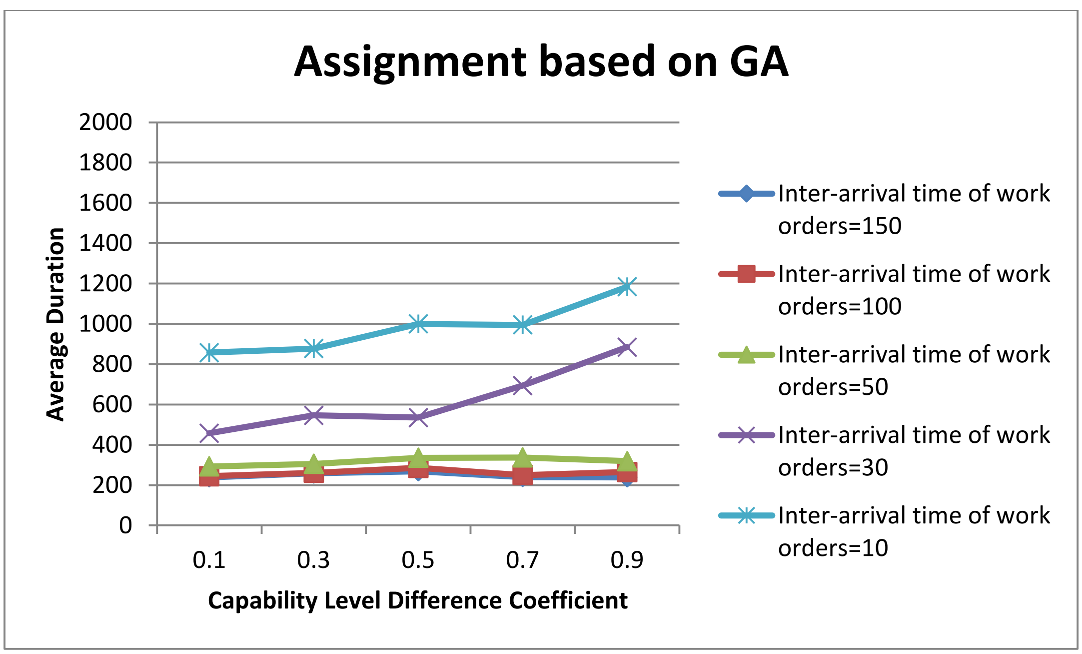


Figure 19. Results of ABOGA with Varying Capability Difference Coefficient

ABOGA method has led to lower average duration under low and medium operational tempo than ABOA method under changing capability level difference coefficient. Moreover, it has led to lower average duration under all operational tempos than ABOC method no matter the capability level difference coefficient is (see Table 24). In some cases, ABOA leads by three times the higher average durations than ABOGA method, while ABOC leads by 2 times the higher average durations. However, under high operational tempo, with low capability level difference coefficient (0.1 and 0.3), ABOA methods performs better than ABOC and ABOGA. With inter-arrival time of work orders of 30 seconds (randomly distributed) and capability level difference coefficient 0.1; ABOA found 20% lower average duration than ABOGA assignments. With inter-arrival time of work orders of 10 seconds (randomly distributed) and capability level difference coefficient 0.1; ABOA found 48% lower average duration than ABOGA assignments. It shows that, in the case of capability levels have very low effect on average duration, assigning agents to tasks based on their availability is necessary in order to minimize the average duration. It is important to mention again that low capability level difference coefficient means that the capability levels have low effect on average duration of the work process.

Table 24. Comparison on Outputs from ABOA, ABOC and ABOGA with Varying Capability Difference Coefficient

Operational Tempo	Inter arrival time of work orders (sec., randomly distributed)	Capability level difference coefficient	Average duration from ABOA assignments	Average duration from ABOC assignments	Average duration from ABOGA assignments	% difference between ABOA and ABOGA	% difference between ABOC and ABOGA
low	150	0.1	321.77	259.42	238.76	34.76	8.65
		0.3	496.48	259.42	258.39	92.15	0.40
		0.5	678.06	259.42	248.68	172.67	4.43
		0.7	865.23	259.42	240.24	260.15	7.98
		0.9	1050.89	259.42	238.62	340.40	8.72
low	100	0.1	325.19	327.18	244.32	33.10	33.91
		0.3	504.69	327.18	260.82	93.50	25.44
		0.5	697.99	327.18	285.88	144.16	14.45
		0.7	922.56	327.18	249.85	269.25	30.95
		0.9	1170.3	327.18	264.75	342.03	23.58
medium	50	0.1	335.32	886.82	292.71	14.56	202.96
		0.3	576.38	886.82	305.63	88.59	190.16
		0.5	861.51	886.82	335.61	156.70	164.24
		0.7	1150.12	886.82	337.23	241.05	162.98
		0.9	1497.81	886.82	319.22	369.21	177.81
high	30	0.1	364.03	1378.74	457.01	-20.35	201.69
		0.3	657.6	1378.74	545.81	20.48	152.61
		0.5	972	1378.74	534.73	81.77	157.84

Table 24. Continued

		0.7	1284.67	1378.74	694.03	85.10	98.66
		0.9	1608.29	1378.74	884.22	81.89	55.93
high	10	0.1	448.02	1801.39	857.14	-47.73	110.16
		0.3	830.71	1801.39	877.43	-5.32	105.30
		0.5	1143.38	1801.39	999.98	14.34	80.14
		0.7	1429.62	1801.39	994.80	43.71	81.08
		0.9	1864.93	1801.39	1183.91	57.52	52.16

These experiments show the effect of capability level differences between agents and tasks on Air Interdiction Planning Mission work process duration under different assignment rules namely; ABOC, ABOA, and ABOGA. The ABOGA method is the most robust with consistently delivering lower average durations under varying capability levels.

The Effect of Number of Work Orders

Number of work orders affects the average duration because it increases the maximum duration of work orders, i.e., as the number of work orders increases, the work orders that are waiting in the queue for parallel tasking increases.

As well as the original number of work orders, a lower number of work orders (10) and a higher number of work orders (50) were chosen to be tested under various operational tempos (randomly distributed inter-arrival times of work orders of 10, 30, 50, 100 and 150 seconds).

The results from ABOA and ABOC methods show that the number of work orders does not have a significant effect on average duration for low levels of operational tempo. However, during medium and high operational tempo, the number of work orders starts to show its effect on average duration of the work processes. For medium and high operational tempo, as the number of work orders increase, the average duration increases under ABOA and ABOC methods (Figure 20 and 21, respectively). On the other hand, the developed ABOGA method is less sensitive to increasing number work orders except during very high operational tempo (inter-arrival time of work orders of 10 seconds). During inter-arrival times of work orders of 150, 100, 50 and 30, the average duration does not fluctuate more than 10% percent. However, for inter-arrival times of work orders of 10 seconds, one can see the increase in average duration easily as the number of work orders increases under ABOGA method (Figure 22).

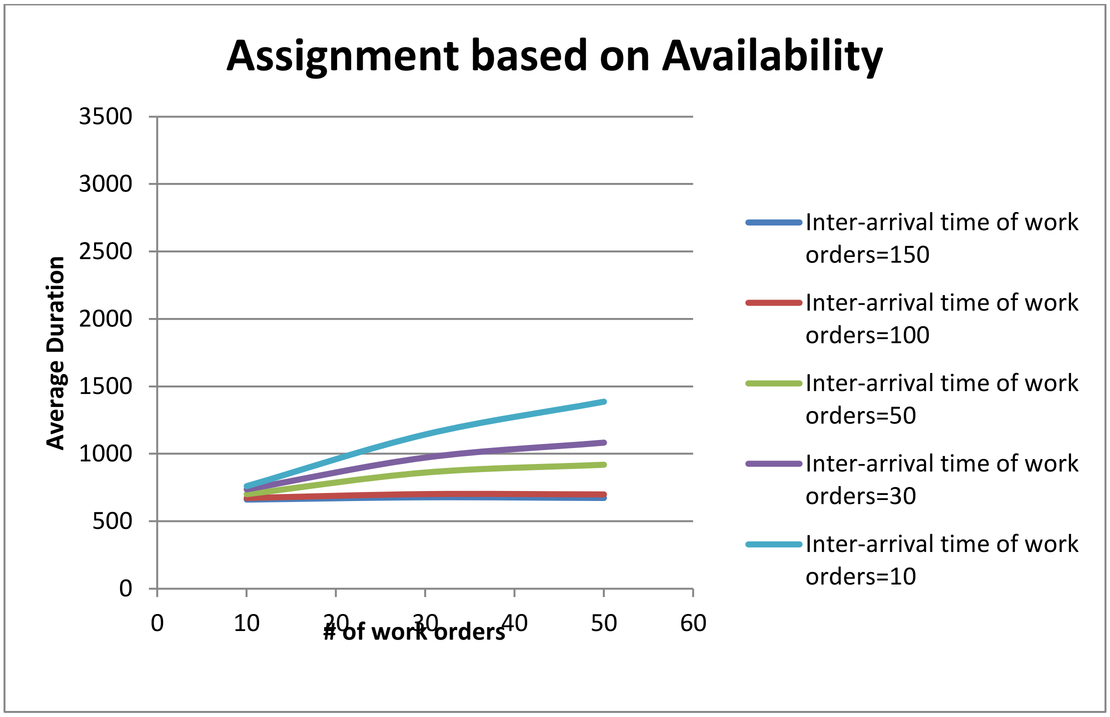


Figure 20. Results of ABOA with Varying Number of Work Orders

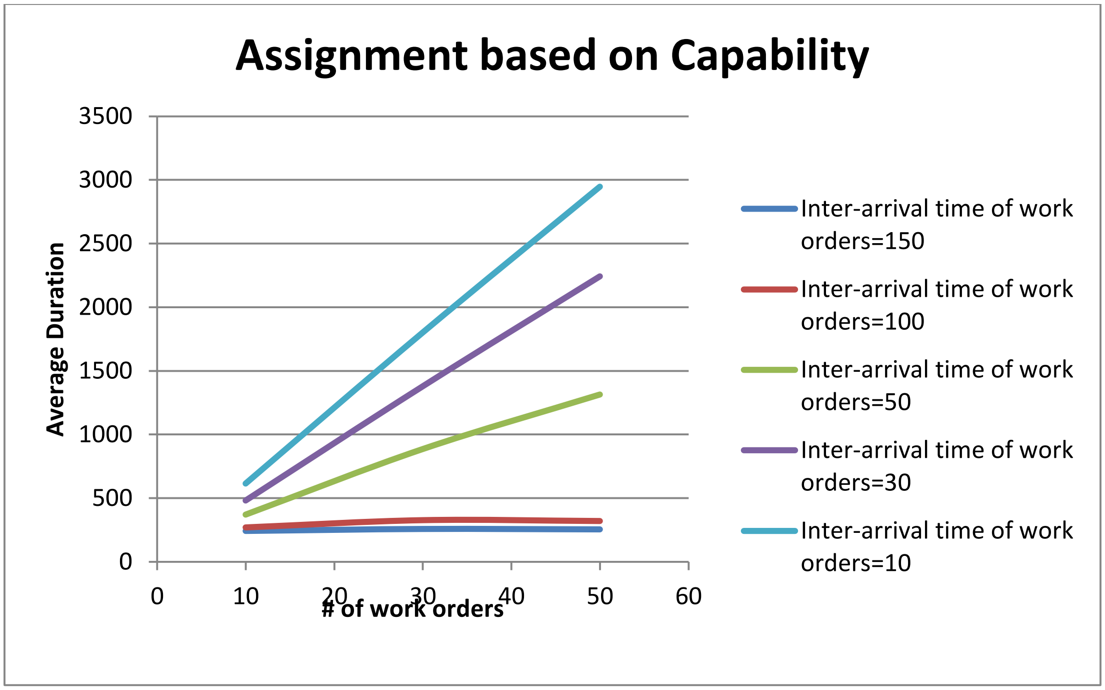


Figure 21. Results of ABOC with Varying Number of Work Orders

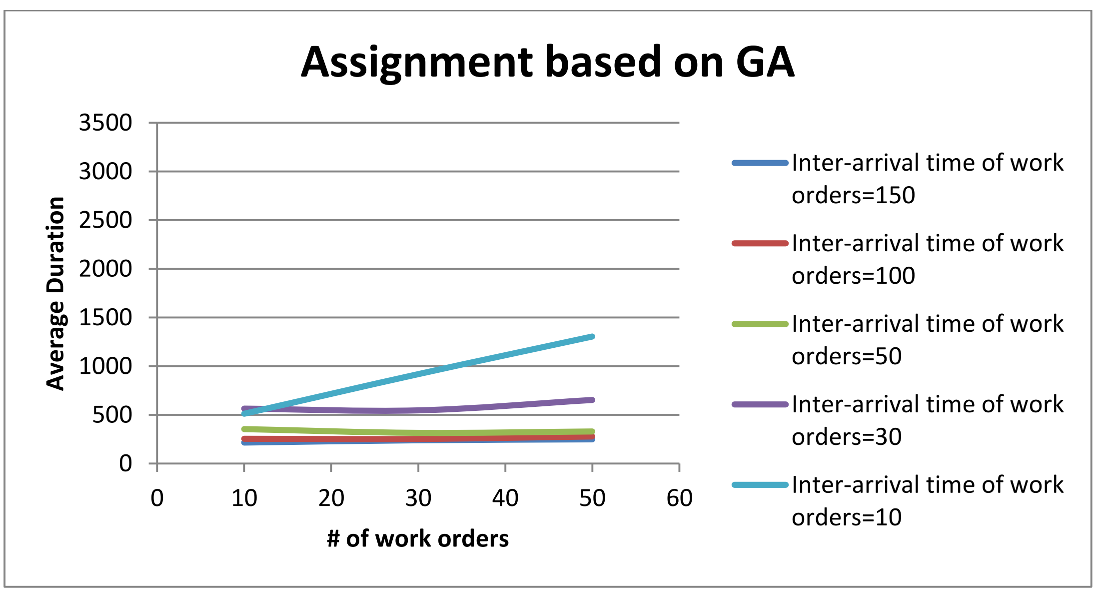


Figure 22. Results of ABOGA with Varying Number of Work Orders

Moreover, ABOC leads to higher average duration for high operational tempo, while ABOA lower and ABOGA is the lowest. For the percent differences in those common practices to ABOGA method, see Table 25.

Table 25. Comparison on Outputs from ABOA, ABOC and ABOGA with Varying Number of Work Orders

Operational Tempo	Inter arrival time of work orders (sec., randomly distributed)	# of work orders	Average duration from ABOA assignments	Average duration from ABOC assignments	Average duration from ABOGA assignments	% difference between ABOA and ABOGA	% difference between ABOC and ABOGA
low	150	10	661.16	242.56	216.55	205.32	12.01
		30	678.06	259.42	239.74	182.83	8.21
		50	673.11	255.98	248.93	170.4	2.83
low	100	10	672.56	271.01	252.85	166	7.18
		30	701.35	327.18	251.13	179.27	30.28
		50	699.08	320.52	277.21	152.19	15.62
medium	50	10	701.48	371.44	354.67	97.78	4.73
		30	861.51	886.82	316.39	172.29	180.29
		50	919.43	1315.47	330.66	178.06	297.83
high	30	10	735.05	481.3	414.82	77.20	16.03
		30	972	1378.74	547.34	77.59	151.9
		50	1083.58	2242.83	652.05	66.18	243.97
high	10	10	759.59	614.42	511.71	48.44	20.07
		30	1143.38	1801.39	919.11	24.4	95.99
		50	1386.65	2947.2	1305.99	6.18	125.67

To summarize, in order to initiate the experiments, the system behavior of Air Interdiction Planning Mission under varying operational tempo is analyzed through applying commonly used task-agent assignment methods namely *assignment based on capability (ABOC)* and *assignment based on availability (ABOA)*. ABOC leads to lower average duration during low and medium operational tempo and ABOA leads to lower average duration during high operational tempo showing that variables of capability and mental workload of the agents are important for the duration of the work process. Once the appropriate area of concern was identified, the developed *assignment based on GA* simulation optimization method (ABOGA) was employed. First, the results obtained from ABOGA method were compared to the results obtained from applying the common practices to the problem. In low operational tempo, the solutions found by ABOGA were better (around 10%), but very close to solutions found by ABOC. However, during medium and high operational tempo, the average duration resulted from ABOC started to increase gradually. The reason is ABOC over assigns the most capable agent while some of the agents stay idle. The ABOGA method addresses this issue and finds the right assignments without over assigning an agent. Even though, ABOA takes the workload of the agent into consideration, it doesn't consider the capability levels and result in assigning agents to tasks with unmatched capabilities. As a result, it ends up with higher average duration than the ABOGA. From the graph in Figure 14, one can see the considerable difference in the average duration results from common practices and ABOGA method. Especially during medium and high operational tempo, this difference gap increases. It can be concluded that ABOGA method finds the task-agent assignments that lead to lower average duration of work process over the ABOC and ABOA methods. Second, the developed ABOGA is compared to another simulation optimization tool, OptQuest. In most cases (except experiment no 1 and 4, in these experiments the results from OptQuest and ABOGA are equal) GA based algorithm performs better than OptQuest algorithm under the stated rules (convergence rate is 100, maximum number of generation is 1000). While the difference between ABOGA and OptQuest is as low as 0 and 0.003% (nearly the same) in low operational tempo, this difference increases to 95% in higher operational tempo levels where ABOGA performs better. Moreover, the randomness in OptQuest output is higher than the ABOGA output, which makes ABOGA more reliable (Figure 15). While OptQuest leads to a better solution than common practices in general; overall, it has been proved that the developed ABOGA approach leads better solutions under any operational tempo setting than all the alternative approaches stated in this study.

Additionally, the effects of parameters namely "*Capability Level Difference Coefficient*" and "*Number of Work Orders*" on the average work process duration (output) is tested under varying operational tempo in order to have further understanding of the impact of these parameters on the assignment methods.

ABOC method has found to be insensitive to capability level difference coefficient, while as this coefficient increases the average duration result from ABOA method increases. ABOGA method has found to be sensitive to this coefficient only on very high operational tempo. Under high operational tempo, with low capability level difference coefficient (0.1 and 0.3), ABOA methods performs better than ABOC and ABOGA. Note that low capability level difference coefficient means that the capability level of agents has low effect on task duration. For the managerial implications, if the work process under consideration is a process with high operational tempo and the agents' capabilities are not important then there is no need to develop a simulation-optimization model. In that case using the common approach of assigning agents to tasks based on their workload availability is the way to achieve lowest average duration. On the other hand, using ABOGA methods is beneficial under medium operational tempo with any capability level difference coefficient. ABOGA method finds assignments that lead to an average of 200% lower average durations than the common approaches. Under low operational tempo, ABOC method performs better than ABOA. However, ABOGA method's performance is better than both common approaches. Note that the results from ABOC and ABOGA under very low operational tempo are very close. This is also true for the cases that the number of work orders changing. Both, ABOC and ABOA methods are found to be sensitive to number of work orders during medium and high operational tempo. During medium and high operational tempo, as the number of work orders increases the average duration increased under ABOC and ABOA rules. Once again, ABOGA was only sensitive to number of work orders under very high operational tempo.

Overall, ABOGA method has proved to find the right agent-task assignments that lead to highest performance (lower average duration) for work processes (except the extreme case of a work process under high operational tempo, with low capability level difference coefficient) than the other methods: ABOC, ABOA and OptQuest.

6. CONCLUSIONS AND FUTURE WORK

This study proposes that optimizing task-employee assignment according to employees' capabilities while keeping them under their mental workload threshold to prevent them from making errors, results in better performance for work processes, especially during critical time junctions (i.e. work processes in an emergency room or military command center). For these work processes, timeliness is the key of the good performance. Accordingly, the goal is to select the best employee-task assignment in order to minimize average duration of a work process. Due to uncertainties inherent in the problem related with inter-arrival time of work orders, task durations, and employees' instantaneous workload, a simulation-optimization approach is utilized to solve the problem. More specifically, a discrete event human performance simulation model is used to evaluate the objective function of the problem together with a genetic algorithm based meta-heuristic optimization approach to search the solution space. In order to measure mental workload, a subjective mental workload measurement method (VACP) is employed by integrating it in the simulation model. The genetic algorithm used in the optimization engine is enhanced with additional rules (i.e. similarity avoid rate, mutation increase rate) and tuned for the right GA parameters (population size, crossover rate, mutation rate). The integration of the simulation model and optimization engine is achieved through exchange of text files. Moreover, throughout the dissertation, software requirements in order to achieve the integration are discussed in detail.

The proposed methodology has been shown to be advantageous in determining the right task-agent assignments by taking in to consideration employees' qualifications and mental workload in order to minimize average duration of a work process. The *Air Interdiction Planning Mission* work process is used as an example to show the effectiveness of the proposed approach.

In order to establish the baseline conditions for the example work process, first the simulation model combined with common practices namely, *assignment based on capability* and *assignment based availability,* are ran through various operational tempo (frequency of inter-arrival time of work orders). In doing so, low, medium and high operational tempo ranges are defined. These common practices represent the two extreme assignment rules that are used regularly in organizations. *Assignment based on capability* method represents the extreme case where only capability of employees is used as the assignment criteria. *Assignment based availability* method represents the other extreme case where only the availability of employees is used as the assignment criteria. According to the results, it is found that assignment based on capability method leads to better performance during lower operational tempo and assignment based on availability method leads to better performance during high operational tempo. However, the developed GA based simulation optimization method, *assignment based on genetic algorithm* found the task-employee assignments that lead to lower average duration for the work process than the common practices under any operation tempo range (low, medium, and high). In low operation tempo, the results from the developed method were better but close to assignment based on capability method results. As the operational tempo increases, the gap in the results from common practices and the developed method increased as well. It is shown that by using the developed method, one can find the task- employee pairs that result in lower average duration of the work process than common practices while keeping the employees under their workload threshold to prevent them from making errors.

Moreover, the developed method is compared to a commercial simulation optimization tool OptQuest. During medium and high operational tempo, the assignment based on genetic algorithm found the employee-task pairs that result in substantially lower average durations than OptQuest.

From a practitioner's point of view, in a work process assignment problem, there is a high number of possible agent-task assignments and this number increases exponentially as the number of tasks and employees increases (for instance, under the example scenario the possible agent-task assignments were 10,000,000,000). It is not possible for a decision maker to evaluate all possibilities in order to decide which pair would lead to the lowest duration. As a result, employing the developed simulation optimization method is paramount when finding the agent-task pairs that lead to a lower average duration under varying operational tempo. This is done while also insuring the employees are not overloaded. Especially if the work process is under medium or high operational tempo (such as work processes that intelligence analysts conduct during a war effort), the benefit of using this simulation optimization method is immense. The example work process showed the developed method could find solutions, which are up to 190% better in terms of duration. Moreover, if the work process is under very low operational tempo, the developed simulation optimization still outperforms commonly used assignment methods.

The effects of *Capability Level Difference Coefficient* and *Number of Work Orders* on the average work process duration is tested under varying operational tempo in order to have a further understanding of these parameters on the need for the simulation-optimization method. The assignment based on capability method has found to be insensitive to capability level difference coefficient. The duration does not change, regardless of the capability level difference coefficient. On the other hand, when this coefficient increases, the average duration result from assignment based on availability method also increases. The developed method has found to be sensitive to this coefficient only on very high operational tempo. Using the developed simulation optimization method is found to be most beneficial under medium operational tempo with any capability level difference coefficient. The developed method finds assignments that lead to around 200% lower average durations than the common approaches under medium operational tempo with varying capability level difference coefficient.

The common assignment approaches are found to be sensitive to the number of work orders during medium and high operational tempo. During medium and high operational tempo, as the number of work orders increases, the average duration increased under these methods. The developed approach was only sensitive to number of work orders under very high operational tempo.

For a decision maker, employing the developed approach would give the most benefit if the work process were under medium and high operational tempo. Moreover, if the capability levels of the employees have dramatic effect on the tasks durations, this benefit can increase up to 200%. Furthermore, the benefit of using the developed approach increases in case of the number of work orders for the process increases.

The numerical tests show the developed approach finds better solutions over common practices and other simulation-optimization methods (such as a commercial simulation optimization tool, OptQuest). By combining the benefits of optimization and simulation, the overall approach provides increased ability to understand the impact of operational tempo, workload threshold levels, and employee capabilities in terms of the organizational work process. The areas where the developed method provide the most benefit is explained in detail. Computational results of this study not only provide managerial visions and measure the significance of intangible factors in the employee assignment process, but also highlight the importance of computational tools such as simulation optimization of the assignment problem for work processes in critical time junctions.

6.1. Summary of Specific Findings for the Air Interdiction Planning Mission Case Study

- ABOC rule performs better than ABOA under low operational tempo. However, ABOA rule returns better (lower) average duration results in high operational tempo when timeliness is crucial (Figure 11). This result confirms the importance of both parameters “agent’s capability” and agent’s workload level” on the performance of a work process.
- ABOGA has better results than both ABOC and ABOA rules in low and high operational tempo. In low operational tempo, the results from ABOGA are closer to the results from ABOC. Yet, as the operational tempo increases, the benefit of using ABOGA starts to increase (Figure 14). In some cases, ABOGA finds assignments that return average duration up to 154% better than ABOC and 190% better than ABOA (Table 20). For work processes in critical time junctions, it is a very substantial difference.
- The commercial simulation optimization package “OptQuest” is used to solve the same problem with the same parameters. OptQuest found better solutions than ABOA method in every operational tempo and ABOC method in medium and high operational tempo (Table 22).
- However, ABOGA found assignments that return in better average duration than OptQuest (Figure 15). The difference between ABOGA and OptQuest is less in low operational tempo, and increases as the operational tempo increases up to 95% (Table 22).
- Overall, ABOGA is the approach that finds agent-task pairs that return the lowest (better) average duration (Figure 16).

- Capability level difference coefficient (the coefficient that emphasizes the importance of the difference between capability level of agents and required capability level from tasks on the task duration) affects the average duration negatively for ABOA method (Figure 17). As the coefficient increases, the average duration increases linearly. However, this coefficient does not have an effect on ABOC method (Figure 18). In ABOGA case, the coefficient affects the average duration results negatively in high operational tempo while there is no significant effect in medium and low operational tempo (Figure 19).
- Capability level difference coefficient (the coefficient that emphasizes the importance of the difference between capability level of agents and required capability level from tasks on the task duration) affects the average duration negatively for ABOA method (Figure 17). As the coefficient increases, the average duration increases linearly. However, this coefficient does not have an effect on ABOC method (Figure 18). In ABOGA case, the coefficient affects the average duration results negatively in high operational tempo while there is no significant effect in medium and low operational tempo (Figure 19).
- Overall, ABOGA leads to better results in finding the task-agent pairs that leads to lower average duration than ABOA and ABOC methods under changing capability level difference coefficient except the of the work process under high operational tempo, with low capability level difference coefficient. In the mentioned case, ABOA method performs better than ABOC and ABOGA. (Table 24).
- ABOA and ABOC methods are proved sensitive to changing number of work orders. As the number of work orders increase the average duration output from these methods increases in high operational tempo (inter-arrival times of work orders:10- 30 -50 sec.)

(Figure 20 and Figure 21). On the other hand, ABOGA only sensitive to number of work orders in very high operational tempo (inter-arrival time work order 10 sec.) (Figure 22).

- It is important to mention that under very low operational tempo with changing number of work orders the results obtained from ABOGA and ABOC methods were very close, while ABOGA method perform better an average of 10% than ABOC.
- Overall, once again ABOGA leads to better results in finding the task-agent pairs that leads to lower average duration than ABOA and ABOC methods under changing number of work orders (Table 25).

6.2. Contributions to the Body of Knowledge

The major contributions of this study are as follows:

- A *genetic algorithm based human performance simulation optimization* approach that finds sufficiently good solutions of employee-task assignments in order to minimize average durations of work processes in critical time junctions.
- An employee-task assignment tool that can handle large solution spaces (high number of employee and tasks).
- A simulation modeling framework that embraces the stochastic nature of work processes (such as task durations, inter arrival time of work orders, employees' instantaneous workload).
- A human performance simulation-modeling tool, which is integrated seamlessly with other software. Current human performance simulation modeling tools (for commercial use) are only capable of getting input from the user (such as IMPRINT, C3TRACE,

IPME) which limit the analysis to only what-if analysis. The need of a human performance simulation-modeling tool that communicates with other software is satisfied.

- A flexible tool, which managers can use to evaluate different work processes with different task-employee sizes, capabilities, workloads and operating rules.

6.3. Future Work

Some future research directions after this dissertation will include the followings:

- The key limitation of the overall solution approach lies in the large computing times that are mainly due to simulation. As a future research direction, methods to reduce the time spent in simulation should be investigated.
- Another (more sophisticated) alternative assignment method that explores the ratio of capability and availability of the employee as a metric to get an acceptable solution should be developed.
- The GA based simulation optimization method should be applied to bigger size problems with work processes that comprised of 15 to 20 tasks.
- A military environment task process is used in order to test the developed approach. Another context, such as health care environment should be considered.
- Beside the VACP workload measurement method, NASA-TLX should be applied and compared to the results from VACP method.

REFERENCES

Acuna, S., & Juristo, N. (2004). Assigning people to roles in software projects. *Software Practice and Experience, 34*(7), 675-696. doi: 10.1002/spe.586

Acuna, S., Juristo, N., & Moreno, A. M. (2006). Emphasizing human capabilities in software development. *Software, IEEE, 23*(2), 94-101. doi: 10.1109/ms.2006.47

Al-Aomar, R. (2006). Incorporating robustness into genetic algorithm search of stochastic simulation outputs. *Simulation Modelling Practice and Theory, 14*(3), 201-223. doi: 10.1016/j.simpat.2005.05.001

Alidaee, B., Gao, H., & Wang, H. (2010). A note on task assignment of several problems. *Computers & Industrial Engineering, 59*(4), 1015-1018. doi: 10.1016/j.cie.2010.07.010

Amaran, S., Sahinidis, N. V., Sharda, B., & Bury, S. J. (2016). Simulation optimization: A review of algorithms and applications. *Annals of Operations Research, 240*(1), 351-380. doi: 10.1007/s10479-015-2019-x

Ammeri, A., Dammak, M. C., Chabchoub, H., Hachicha, W., & Masmoudi, F. (2013, 29-31 May 2013). *A simulation optimization approach-based genetic algorithm for lot sizing problem in a MTO sector*. Paper presented at the Advanced Logistics and Transport (ICALT).

Andradóttir, S. (2002). Simulation optimization: Integrating research and practice. *INFORMS Journal on Computing, 14*(3), 216.

Andradóttir, S. (2006). Chapter 20 An overview of simulation optimization via random search. In G. H. Shane & L. N. Barry (Eds.), *Handbooks in operations research and management science,* 13, 617-631. doi: 10.1016/S0927-0507(06)13020-0

Andradottir, S. (2015). A review of random search methods. In *Handbook of Simulation Optimization* (pp. 277–292). doi: 10.1007/978-1-4939-1384-8

April, J., Better, M., Glover, F., Kelly, J., & Laguna, M. (2006, 3-6 Dec. 2006). *Enhancing business process management with simulation optimization.* Paper presented at the Simulation Conference, 2006. WSC 06. Proceedings of the Winter.

Azadivar, F., & Tompkins, G. (1999). Simulation optimization with qualitative variables and structural model changes: A genetic algorithm approach. *European Journal of Operational Research, 113*(1), 169-182. doi: 10.1016/S0377-2217(97)00430-X

Baesler, F. F., Jahnsen, H. E., & DaCosta, M. (2003). *Emergency departments I: The use of simulation and design of experiments for estimating maximum capacity in an emergency room*. Paper presented at the Proceedings of the 35th conference on Winter simulation: driving innovation, New Orleans, Louisiana.

Barton, R. (2005). Meta-Model Based Simulation Optimization. In S. G. Henderson & B. L. Nelson (Eds.), *Handbooks in Operations Research and Management Science: Simulation*. (pp. 535-574) Amsterdam: Elsevier.

Bierbaum, C. R., Fulford, L. A., Hamilton, D. B., & Fort, R. A. L. (1990). *Task Analysis/Workload (TAWL) User's Guide. Version 3.0*. Ft. Belvoir: Defense Technical Information Center.

Boles, D. B., & Adair, L. P. (2001). *Validity of the multiple resources questionnaire.* Paper presented at the Proc Hum Factors rgonomics Soc 45th Annual meet Minneapolis,MN.

Cain, B. (2007). A review of the mental workload literature CANADA: Defence Research and Deelopment Canada Toronto.

Carayon, P., & Gürses, A. P. (2005). A human factors engineering conceptual framework of nursing workload and patient safety in intensive care units. *Intensive and Critical Care Nursing, 21*(5), 284-301. doi: 10.1016/j.iccn.2004.12.003

Carley, K. M. (2002). Computational organizational science and organizational engineering. *Simulation Modelling Practice and Theory, 10*(5–7), 253-269. doi: 10.1016/S1569-190X(02)00119-3

Carson, Y., & Maria, A. (1997). *Simulation optimization: Methods and applications*. Paper presented at the Proceedings of the 29th Conference on Winter Simulation, Atlanta, Georgia, USA.

Casali, J. G., & Wierwille, W. W. (1983). A comparison of rating scale, secondary-task, physiological, and primary-task workload estimation techniques in a simulated flight task emphasizing communications load. *Human Factors: The Journal of the Human Factors and Ergonomics Society, 25*(6), 623-641. doi: 10.1177/001872088302500602

Cavazzuti, M. (2013). Deterministic optimization. *Optimization Methods* (pp. 77-102): Springer Berlin Heidelberg.

Cheng, H., & Chu, X. (2012). Task assignment with multiskilled employees and multiple modes for product development projects. *International Journal of Advanced Manufacturing Technology, 61*(1-4), 391-403. doi: 10.1007/s00170-011-3686-7

Chu, P. C., & Beasley, J. E. (1997). A genetic algorithm for the generalised assignment problem. *Computers & Operations Research, 24*(1), 17-23. doi: 10.1016/S0305-0548(96)00032-9

Colombi, J. M., Miller, M. E., Schneider, M., McGrogan, M. J., Long, C. D. S., & Plaga, J. (2012). Predictive mental workload modeling for semiautonomous system design: Implications for systems of systems. *Systems Engineering, 15*(4), 448-460. doi: 10.1002/sys.21210

Cook, M. B., Heacox, N. J., Averett, M. G., & Handley, H. A. H. (2012). MASTS: Multi-Perspective Assessments and Tasking for Submarines. Arlington, CA: Office of Naval Research.

Cummings, M. L., & Guerlain, S. (2007). Developing operator capacity estimates for supervisory controls of autonomous vehicles. *Human Factors, 49*(1).

Dahn, D., & Laughery, K. R. (1997). *The integrated performance modeling environment; Simulating human-system performance*. Paper presented at the Proceedings of the 29th Conference on Winter simulation, Atlanta, Georgia, USA.

Daniel, J. S. R., & Rajendran, C. (2005). A simulation-based genetic algorithm for inventory optimization in a serial supply chain. *International Transactions in Operational Research, 12*(1), 101-127. doi: 10.1111/j.1475-3995.2005.00492.x.

Daskal, L. (2016, August 12). 7 Reasons the besteEmployees quit, even when they like their Job. *Inc.*

Dengiz, B., & Alabas, C. (2000). *Simulation optimization using tabu search*. Paper presented at the Proceedings of the 32nd Conference on Winter simulation, Orlando, Florida

Derringer, G., & Suich, R. (1980). Simultaneous optimization of several response variables. *Journal of Quality Technology, 12*, 214-219.

Dixon, S. R., Wickens, C. D., & Chang, D. (2005). Mission control of multiple unmanned aerial vehicles: A workload analysis. *Human Factors: The Journal of the Human Factors and Ergonomics Society, 47*(3), 479-487. doi: 10.1518/001872005774860005

Eiselt, H. A., & Marianov, V. (2008). Employee positioning and workload allocation. *Computers & Operations Research, 35*(2), 513-524. doi: 10.1016/j.cor.2006.03.014

Eskandari, H., Mahmoodi, E., Fallah, H., & Geiger, C. D. (2011). Performance analysis of comercial simulation-based optimization packages: OptQuest and witness optimizer. *Proceedings - Winter Simulation Conference*, (2005), 2358–2368. doi: 10.1109/WSC.2011.6147946

Fu, M. C. (2002). Optimization for simulation: Theory vs.practice. *INFORMS Journal on Computing, 14*(3), 192.

Fu, M. C. (2005). Gradient Estimation. In S. G. Henderson & B. L. Nelson (Eds.), *Handbooks in Operations Research and Management Science: Simulation*. Amsterdam: Elsevier.

Ghazavi, E., & Lotfi, M. M. (2016). Formulation of customers' shopping path in shelf space planning: A simulation-optimization approach. *Expert Systems with Applications, 55*, 243-254. doi: http://dx.doi.org/10.1016/j.eswa.2016.01.043

Goodfriend, J. H. (1995). Discrete event systems: Sensitivity analysis and stochastic optimization by the score function method. *Technometrics, 37*(1), 124-125. doi: 10.1080/00401706.1995.10485910

Gorbach, C., Blanton, L., Lukawski, B. A., Varkey, A. C., Pitman, E. P., & Garey, K. W. (2015). Frequency of and risk factors for medication errors by pharmacists during order verification in a tertiary care medical center. *American Journal of Health-System Pharmacy, 72*(17), 1471-1474. doi: 10.2146/ajhp140673

Grooms, B. E. (2009). Close Air Support: United States Armed Forces

Hart, S., & Wickens, C. (1990). Workload assessment and prediction. In H. Booher (Ed.), *Manprint* (pp. 257-296): Springer: Netherlands.

Hart, S. G., & Staveland, L. E. (1988). Development of NASA-TLX (Task Load Index): Results of empirical and theoretical research. *Human Mental Workload* (pp. 139-183). Amsterdam: North Holland

He, J., Huang, Y., & Chang, D. (2015). Simulation-based heuristic method for container supply chain network optimization. *Advanced Engineering Informatics, 29*(3), 339-354. doi: http://dx.doi.org/10.1016/j.aei.2014.08.001

Holden, R. J., Scanlon, M. C., Patel, N. R., Kaushal, R., Escoto, K. H., Brown, R. L., Karsh, B.-T. (2011). A human factors framework and study of the effect of nursing workload on patient safety and employee quality of working life. *BMJ Quality & Safety*, *20*(1), 15-24. doi: 10.1136/bmjqs.2008.028381

Holland, J.H. (1975). Adaptation in natural and artificial systems. The University of Michigan Press, Ann Arbor, MI.

Huey, B. M., & Wickens, C. D. (1993). *Workload transition: Implications for individual and team performance*: The National Academies Press.

Hunn, B. P., Schweitzer, K. M., Cahir, J. A., & Finch, M. M. (2008). IMPRINT analysis of an unmanned air system geospatial information process: Army Research Laboratory.

Jackson, E. (2014). The top 8 reasons your best people are about to quit -- And how you can keep them. *Forbes*.

Kamrani, F., Ayani, R., & Moradi, F. (2012). A framework for simulation-based optimization of business process models. *Simulation*, *88*(7), 852-869. doi: 10.1177/0037549711417880

Kamrani, F., Ayani, R., Moradi, F., & Holm, G. (2009). *Estimating performance of a business process model*. Paper presented at the Winter Simulation Conference, Austin, Texas.

Kamrani, F., Rassul, A., & Karimson, A. (2010, 17-19 May 2010). *Optimizing a business process model by using simulation.* Paper presented at the Principles of Advanced and Distributed Simulation (PADS), 2010 IEEE Workshop on.

Keller, J. (2002, 8-11 Dec. 2002). *Human performance modeling for discrete-event simulation: workload.* Paper presented at the Simulation Conference, 2002. Proceedings of the Winter.

Kelly, J. P. (2002). Simulation optimization is evolving. *INFORMS Journal on Computing, 14*(3), 223.

Kelton, W. D. (2000). *Design of experiments: Experimental design for simulation*. Paper presented at the Proceedings of the 32nd Conference on Winter Simulation, Orlando, Florida.

Keskin, B. B., Melouk, S. H., & Meyer, I. L. (2010). A simulation-optimization approach for integrated sourcing and inventory decisions. *Computers & Operations Research*, *37*(9), 1648-1661. doi: 10.1016/j.cor.2009.12.012

Kilduff, P. W., Swoboda, J. C., & Barnette, B. D. (2002). Command, control, and communications: Techniques for the reliable assessment of concept execution (C3TRACE) modeling environment: The tool. Aberdeen Proving Ground, MD 21005-5425 Human Research and Engineering Directorate, ARL.

Kim, S. H., & Nelson, B. L. (2005). Selecting the best system. In S. G. Henderson & B. L. Nelson (Eds.), *Handbooks in Operations Research and Management Sciense: Simulation* Amsterdam: Elsevier.

Klassen, K. J., & Yoogalingam, R. (2009). Improving performance in outpatient appointment services with a simulation optimization approach. *Production and Operations Management, 18*(4), 447-458. doi: 10.1111/j.1937-5956.2009.01021.x

Kleijnen, J. P. C. (1995). *Sensitivity analysis and optimization in simulation: Design of experiments and case studies*. Paper presented at the Proceedings of the 27th conference on Winter simulation, Arlington, Virginia, USA.

Lamoureux, T. (1999). The influence of aircraft proximity data on the subjective mental workload of controllers in the air traffic control task. *Ergonomics, 42*(11), 1482-1491. doi: 10.1080/001401399184839

Laughery, K. R., Archer, S., & Corker, K. (2007). Modeling human performance in complex systems. *Handbook of Industrial Engineering* (pp. 2407-2444): John Wiley & Sons, Inc.

Law, A. M., & Kelton, D. M. (1999). *Simulation modeling and analysis*: McGraw-Hill Higher Education.

Levin, S., Aronsky, D., Hemphill, R., Han, J., Slagle, J., & France, D. J. (2007). Shifting toward balance: Measuring the distribution of workload among emergency physician teams. *Annals of Emergency Medicine, 50*(4), 419-423. doi: 10.1016/j.annemergmed.2007.04.007

Lipski, J. (2013). *Optimization of production process* (A. Świć & J. Lipski Eds.). Lublin, Poland: Lublin University of Technology.

McCormack, R., & Coates, G. (2015). A simulation model to enable the optimization of ambulance fleet allocation and base station location for increased patient survival. *European Journal of Operational Research, 247*(1), 294-309. doi: 10.1016/j.ejor.2015.05.040

McCracken, J. H., & Aldrich, T. B. (1984). Analyses of selected LHX mission functions: Implications for operator workload and system automation goals. Fort Rucker, AL: U.S. Army Research Institute Aviation Research and Development Activity.

Metzger, N. L., Chesson, M. M., & Momary, K. M. (2015). Simulated order verification and medication reconciliation during an introductory pharmacy practice experience. *American Journal of Pharmaceutical Education, 79*(7).

Miller, S. (2001). Literature review: Workload measures (N. A. D. Simulator, Trans.). Iowa City: The University of Iowa.

Minxin, S., Gwo-Hshiung, T., & Liu, D.-R. (2003, 6-9 Jan. 2003). *Multi-criteria task assignment in workflow management systems.* Paper presented at the System Sciences, 2003. Proceedings of the 36th Annual Hawaii International Conference on.

Mitchell, D. (2000). Mental workload and ARL workload modeling tools. Army Research Laboratory.

Mitchell, D. K. (2009). Workload analysis of the crew of the Abrams V2 SEP: Phase I baseline IMPRINT Model (U. S. A. R. Laboratory, Trans.): Human Research and Engineering Directorate, ARL.

Mitchell, D. K., Samms, C. L., Henthorn, T., & Wojciechowski, J. Q. (2003). Trade study: A two- versus three-soldier crew for the mounted combat system (MCS) and other future combat system platforms (U. S. A. R. Laboratory, Trans.): Human Research & Engineering Directorate.

Mitchell, D. (2009). *Successfully changing conceptual system designs using human performance modeling.* Paper presented at the Human System Interation Symposium Annapolis, MD.

Mitchell, D., & Brennan, G. (2009). Infantry squad using the common controller for small unmanned ground vehicle control: Soldier workload analysis: Army Research Laboratory.

Mitchell, D. K., & Samms, C. (2009). Workload warriors: Lessons learned from a decade of mental workload prediction using human performance modeling. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting, 53*(12), 819-823. doi: 10.1177/154193120905301212

O'Donnell, R. D., & Eggemeier, F. T. (1986). Workload assessment methodology. In K. R. Boff, L. Kaufman, & J. P. Thomas (Eds.), *Handbook of Perception and Human Performance: Cognitive Processes and Performance* (Vol. 2). Canada: John Wiley and Sons,.

Olaffson, S. (2005). Metaheuristics. In S. G. Henderson & B. L. Nelson (Eds.), *Handbooks in Operations Research and Management Science: Simulation*. Amsterdam: Elsevier.

Otero, Centeno, G., Ruiz-Torres, A. J., & Otero, C. E. (2009). A systematic approach for resource allocation in software projects. *Computers & Industrial Engineering, 56*(4), 1333-1339. doi: http://dx.doi.org/10.1016/j.cie.2008.08.002

Otero, Otero, L. D., Weissberger, I., & Qureshi, A. (2010, 24-26 March 2010). *A multi-criteria decision making approach for resource allocation in software engineering.* Paper presented at the Computer Modelling and Simulation (UKSim), 2010 12th International Conference on.

Parker, D. (2011, March 19). Employee overload is bad for business. *The Australian Business Review*.

Perdu, D. (1997). *Distributed process coordination in adaptive command and control teams.* (Ph. D.), George Mason University, Fairfax, VA. (GMU/C3I-184-T)

Persson, A., Grimm, H., Ng, A., Lezama, T., Ekberg, J., Falk, S., & Stablum, P. (2006, 3-6 Dec. 2006). *Simulation-based multi-objective optimization of a real-world scheduling problem.* Paper presented at the Proceedings of the 2006 Winter Simulation Conference.

Plott, B., Quesada, S., Kilduff, P., Swoboda, J., & Allender, L. (2004). Using an information-driven decision_making human performance tool to assess U.S. Army command, control, and communication issues. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting, 48*(20), 2396-2400. doi: 10.1177/154193120404802005

Pomranky, R. a, & Wojciechowski, J. Q. (2007). *Determination of mental workload during operation of multiple unmanned systems*.

Rao, S. S. (2009). *Engineering optimization: Theory and practice*. *Wiley*. doi: 10.1002/9780470549124

Reilley, S., Grasha, A. F., & Schafer, J. (2002). Workload, error detection, and experienced stress in a simulated pharmacy verification task. *Perceptrial and Motor Skills, 95*, 27-46.

Reid, G. B., & Colle, H. A. (1988). Critical SWAT values for predicting operator overload. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting, 32*(19), 1414-1418. doi: 10.1177/154193128803201923

Reid, G. B., Potter, S. S., & Bressler, J. R. (1989). Subjective workload assessment technique (SWAT): A user's guide (A. A. M. R. Laboratory, Trans.). OH: Systems Research Laboratories, Inc.

Samms, C., & Mitchell, D. (2010). Predicting the consequences of workload management strategies with human performance modeling. Maryland: Army Research Laboratory.

Seppanen, M. S. (2016). Building flexible data-driven models. from http://www.arena-china.com/uploads/soft/120220/1_1924354721.pdf

Smallman, H. S., Cook, M. B., Beer, R. D., & Lacson, F. C. (2009). Concept of operations (CONOPS) for submarine-based monitor and control of unmanned aerial vehicles for over-the-horizon targeting applications. San Diego, CA: Pacific Science & Engineering Group.

Stanton, N. A., Salmon, P. M., Walker, G. H., Baber, C., & Jenkins, D. P. (2006). *Human factors methods: A practical guide for engineering and design*: Ashgate Publishing Company.

Sheridan, T. (2012). Human supervisory control in G. Salvendy (Ed.), *Handbook of Human Factors and Ergonomics* John Wiley and Sons, New Jersey.

Schwartz, J., & Erikson, R. (2009, 8/12). Prepare now or lose your best employees soon. *Forbes*.

Subulan, K., & Cakmakci, M. (2012). A feasibility study using simulation-based optimization and Taguchi experimental design method for material handling—transfer system in the automobile industry. *The International Journal of Advanced Manufacturing Technology, 59*(5-8), 433-443. doi: 10.1007/s00170-011-3514-0

Tsai, H.-T., Moskowitz, H., & Lee, L.-H. (2003). Human resource selection for software development projects using Taguchi's parameter design. *European Journal of Operational Research, 151*(1), 167-180. doi: 10.1016/S0377-2217(02)00600-8

Wan, J., & Kleijnen, J. P. C. (2006). Simulation for the optimization of (s , S) inventory system with random lead times and a service level constraint by using arena and OptQuest (Date : 18 June , 2006), 1–35.

Wickens, C. D. (1992). *Engineering psychology and human performance.* New York: HarperCollins Publishers Inc.

Wickens, C. D. (2002). Multiple resources and performance prediction. *Theoretical Issues in Ergonomics Science, 3*(2), 159-177

Wickens, C. D. (2008). Multiple resources and mental workload. *Human Factors, 50*(3), 449-455.

Wong, D., Walters, B., & Fairey, L. (2010). *Validating human performance models of the future Orion crew exploration vehicle.* Paper presented at the 53th Annual Meeting Human Factors Ergonomic Society, San Francisco, CA.

Xie, B., & Salvendy, G. (2000). Review and reappraisal of modelling and predicting mental workload in single- and multi-task environments. *Work & Stress, 14*(1), 74-99. doi: 10.1080/026783700417249

Yang, T., Kuo, Y., & Chang, I. (2004). Tabu-search simulation optimization approach for flow-shop scheduling with multiple processors — A case study. *International Journal of Production Research, 42*(19), 4015-4030. doi: 10.1080/00207540410001699381

Yeh, J.-Y., & Lin, W.-S. (2007). Using simulation technique and genetic algorithm to improve the quality care of a hospital emergency department. *Expert Systems with Applications, 32*(4), 1073-1083. doi: http://dx.doi.org/10.1016/j.eswa.2006.02.017

Yeh, Y.,Y., & Wickens, C. D. (1988). Dissociation of performance and subjective measures of workload. *Human Factors: The Journal of the Human Factors and Ergonomics Society, 30*(1), 111-120. doi: 10.1177/001872088803000110

Zeng, Q., & Yang, Z. (2009). Integrating simulation and optimization to schedule loading operations in container terminals. *Computers & Operations Research, 36*(6), 1935-1944. doi: 10.1016/j.cor.2008.06.010

Zhen, L., Wang, K., Hu, H., & Chang, D. (2014). A simulation optimization framework for ambulance deployment and relocation problems. *Computers & Industrial Engineering, 72*, 12-23. doi: 10.1016/j.cie.2014.03.008

VITA

CANSU KANDEMIR

Engineering Management and Systems Engineering
Old Dominion University, Norfolk, VA
e-mail: ckand002@odu.edu, cansuknd@gmail.com

ACADEMIC QUALIFICATIONS

Ph.D.	2012-2016	Department of Engineering Management and Systems Engineering, Old Dominion University
M.S.	2010-2012	Department of Logistics Management, Izmir University of Economics
B.S.	2006-2010	Department of Industrial Engineering, Izmir University of Economics

PUBLICATIONS

Publications in Peer Reviewed Journals

Kandemir, C., Handley, H.A.H., Thompson, D. (2015). A Workload Model to Evaluate Distracters and Driver's Aid. *International Journal of Industrial Ergonomics. Online Ready.* http://dx.doi.org/10.1016/j.ergon.2016.09.004

Ozpeynirci, O. and **Kandemir, C.** (2015). A Pseudo-Polynomial Time Algorithm for a Special Multi-Objective Order Picking Problem. *International Journal of Information Technology & Decision Making, 14 (5),* pp. 1111-1128 doi:10.1142/S0219622015500169

Biresselioglu, M.E., Hakan Demir, M. and **Kandemir, C.** (2012). Modeling Turkey's future LNG supply security strategy. *Energy Policy*, 46, pp.144-152. doi:10.1016/j.enpol.2012.03.045

Publications in the Proceedings of International Conferences

Handley, H.A.H., Amissah, M., & **Kandemir, C.** (2016). Levels of SysML Compatibility for Collaborative Human System Development. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting,* Washington, DC, USA. doi: 10.1177/1541931213601393

Hancerlioglu, G., Hancerlioglu, O., **Kandemir, C.**, Dario, E. (2015). A Design of Experiment and Simulation Experience in a Health Care System. *Proceedings of the 2nd Global Conference on Engineering and Technology Management 2015,* Chicago, IL, USA.

Kandemir, C. & Handley, H. (2014). Employee Task Assignments for Organization Modeling: A Review of Models and Applications. *Proceedings of the Annual Meeting of the American Society of Engineering Management,* Virginia Beach, VA.