A permutation flowshop model with time-lags and waiting time preferences of the patients

A Thesis Submitted to the College of

Graduate and Postdoctoral Studies

In Partial Fulfillment of the Requirements

For the Degree of Master of Science in Finance

In the Department of Finance and Management Science

Edwards School of Business

University of Saskatchewan

Saskatoon, Saskatchewan, Canada

by Iman Ramezani

Permission To Use

In presenting this thesis in partial fulfillment of the requirements for a Postgraduate degree from the University of Saskatchewan, I agree that the Libraries of this University may make it freely available for inspection. I further agree that permission for copying of this thesis in any manner, in whole or in part, for scholarly purposes may be granted by the professor or professors who supervised my thesis work or, in their absence, by the Head of the Department or the Dean of the College where thesis work was done. It is understood that any copying or publication or use of this thesis or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to the University of Saskatchewan in any scholarly use which may be made of any material in my thesis. Requests for permission to copy or to make other use of material in this thesis in whole or part should be addressed to:

Head of the Department of Finance and Management Science
Edwards School of Business
University of Saskatchewan
25 Campus Drive
Saskatoon, Saskatchewan S7N 5A7
OR
Dean
College of Graduate and Postdoctoral Studies
University of Saskatchewan
116 Thorvaldson Building, 110 Science Place
Saskatoon, Saskatchewan S7N 5C9 Canada

Abstract

The permutation flowshop is a widely applied scheduling model. In many real-world applications of this model, a minimum and maximum time-lag must be considered between consecutive operations. We can apply this model to healthcare systems in which the minimum time-lag could be the transfer times, while the maximum time-lag could refer to the number of hours patients must wait. We have modeled a MILP and a constraint programming model and solved them using CPLEX to find exact solutions. Solution times for both methods are presented. We proposed two metaheuristic algorithms based on genetic algorithm and solved and compared them with each other. A sensitivity of analysis of how a change in minimum and maximum time-lags can impact waiting time and C_{max} of the patients is performed. Results suggest that constraint programming is a more efficient method to find exact solutions and changes in the values of minimum and maximum time-lags can impact waiting times of the patients and C_{max} significantly.

JEL Classification: $C44$,$C61$

Keywords: Waiting times of the patients, Healthcare scheduling, Constraint programming, Genetic algorithm

Acknowledgements

First of all, I would like to express my gratitude to my thesis supervisors, Dr. Hamed Samarghandi and Dr. Keith Willoughby, for their support throughout my studies in University of Saskatchewan. I am also very grateful to Dr. Abdullah Mamun, current Graduate Chair of the Master of Science in Finance department and all the faculties and staff from the Edwards School of Business.

I would like to give my special thanks to my committee member Dr. Mehran Hojati and Dr. Eman Almehdawe for their insightful comments.

I take this opportunity to thank to Saskatchewan Centre for Patient-Oriented Research (SCPOR) for providing funding for this research.

Last but not the least, I like to express my special thanks to my parents and my brothers for all their kind helps and supports throughout my life.

Contents

List of Tables

List of Figures

1 Introduction

Everyone has availed a healthcare facility, whether for doing a check-up, performing a medical test, visiting a physician or accompanying a friend; however, a worrisome aspect of visiting healthcare systems is waiting in a line to receive the necessary care service. Specifically, if a person is suffering from a severe illness, the resulting pain, anxiety, and inconvenience of waiting in a line would increase substantially. Furthermore, spending a lot of time in a line has the risk of worsening the health condition of the patient (Prentice and Pizer, 2007).

Long waiting time increases exposure to pain, but it can increase the mortality rate (Prentice and Pizer, 2007). Barua, Esmail, and Jackson (2014) suggested in their study that only between the years 1993 and 2009, waiting has been the cause of death of an estimated range of $25,456$ to $63,090$ female Canadian patients.

Among healthcare stakeholders, patients are not the only group who would prefer to reduce waiting times. Healthcare decision makers are another group who have the challenge with waiting times and want to reduce waiting times in healthcare systems, and one of their aims is to provide better service by shortening waiting times that improve quality of healthcare systems (Nova Scotia Department of Health and Wellness, 2010). In many healthcare systems total or mean waiting times are estimated as one of the performance criteria of the system and a criterion of patient satisfaction. However, not only mean waiting time is crucial but variations in waiting time are important, and they can be explained by characteristics of patient, clinic and provider level (Dansky and Miles, 1997; Dimakou, Parkin, Devlin, and Appleby, 2009).

Although a considerable amount of effort has been put into decreasing waiting times, it is still a challenge for healthcare decision makers to lower waiting times and the waiting times are not small numbers; for example, statistics of Health Quality Ontario (2018) suggested that on average each patient has to wait for 1.5 hours to visit a doctor in the emergency departments after triage nurse has assessed them for the first time in April 2018 in Ontario. A study for publicly funded hospitals in Quebec suggested that outpatients who need to receive physiotherapy service have a waiting time of more than six months for more than 41% of those services (Deslauriers et al., 2017).

Statistics provided by Health Quality Ontario (2018) showed that on average patients who were admitted to emergency departments had 14.8 hours stay in emergency departments before going to hospitals. Low-urgency patients who sent home after the first visit by doctors spent 2.5 hours from the time they were assessed by the triage nurse until the time they left the emergency and went home. Also, high-urgency patients who sent home after the first visit by doctors spent four hours from the time triage nurses assessed them until the time they left the emergency and went home after their first assessment by a doctor; assuming that the 1.5 hours waiting time can be considered the waiting time for both groups of patients whom doctors sent home, one can conclude that 60%

and 37.5% of the time of patients after being assessed by triage nurses was wasted for waiting in line. These statistics and estimates propose that a reduction in waiting time before seeing the doctor can result in the reduction in the time of the stay in emergency departments significantly.

Negative impact on health and comfort of the patients is not the only detriment of waiting times. Waiting times can impose heavy costs on healthcare systems. A research that aimed to study how waiting times impact the Canadian economic system found that only for the first four priorities out of the five priorities, the cumulative cost of "excess wait" or waiting longer than the medically allowed waiting is $14.8 billion in the year 2007 (Canadian Medical Association, 2008). Ministry of Health and Long-Term Care (2008) indicates that the average cost of extra wait for each patient who needs " total joint replacement surgery" is $26,400. Besides the costs imposed by waiting times, there have been huge investments in Canada to reduce waiting times; for example in 2008, a $109 million investment launched in Ontario to reduce waiting times during the three years period of 2008 – 2011 (Ministry of Health and Long-Term Care , 2008). Despite the investment, Vermeulen, Stukel, Boozary, Guttmann, and Schull (2016) concluded that the program had "modest overall benefits for ED length of stay without adversely affecting quality of care".

There are several solutions that decision makers consider to reduce waiting times or to minimize the negative effects of waiting times on patients (Nova Scotia Department of Health and Wellness, 2010): One solution to manage the flow and reduce waiting times is to streamline patients to special purpose areas. If it is not possible to reduce waiting times then explain the services available to the patients, length of stay and the time that patients need to wait in lines to reduce their anxiety and increase the comfort of the patients (Nova Scotia Department of Health and Wellness, 2010). Another approach is to prioritize patients based on their need to receive healthcare services (Willcox et al., 2007). As a result, patients with lower priority are required to wait more to receive the care they need, and patients who urgently need to receive the care receive the care they require at the earliest possible time. This can be helpful in many situations; however, one drawback of this approach is that low priority patients may feel humiliated and powerless (Dahlen, Westin, and Adolfsson, 2012). One other approach is to control maximum waiting times (Babashov et al., 2017; Sinko, Nikolova, Sutton, et al., 2015).

Notwithstanding, there have been different approaches to manage waiting times, many of these methods have their potential drawbacks (Deslauriers et al., 2017; Sinko et al., 2015). Regardless of the managerial methods that can play an important role in reducing waiting times, the correct sequence of the patients in line and true assignment of patients can reduce waiting time; this can be achieved by developing mathematical models and use of operations research and scheduling methods and techniques.

One potential solution to minimize waiting time and length of stay is to use operations research techniques to find the best settings of a healthcare system. Operations research techniques can be combined with most other waiting time management systems to reduce waiting times. These techniques can help to find the best setting of the system. Operations research techniques can help to improve performance criteria of the system by suggesting the best order and timing of healthcare systems.

Scheduling has been a traditional field of interest in operations research and operation management (Akers, 1956; Bowman, 1956). Applications of scheduling arise in different industries and sectors such as manufacturing (Rahmani and Ramezanian, 2016) and healthcare (Hancock and Algozzine, 2016; Qu, Peng, Kong, and Shi, 2013). Many studies have used scheduling and operations research tools to improve the performance of healthcare system in areas such as optimizing appointment scheduling (Grant, Gurvich, Mutharasan, and Van Mieghem, 2017), optimizing scheduling in emergency departments (Daldoul, Nouaouri, Bouchriha, and Allaoui, 2018), minimizing waiting time, improving access of patient to care and improving patient experience (Denton, Viapiano, and Vogl, 2007; Huang, 2008; Huang and Bach, 2016). Despite a vast range of researches on scheduling in healthcare, there are still situations that have not been discussed enough in the literature. One situation is to consider waiting times in a model. Wait Time Alliance provided some wait times benchmarks for different patients in Canada in different categories such as chronic Pain services, Arthritis Care and another 14th categories; they provided acceptable wait times for different types of diagnosis (*Wait Time for Canada Wait Time Alliance*, 2014).

There exist two types of waiting times for patients in every healthcare system. One kind of waiting time is the one before showing up of patients in the healthcare system and while making an appointment to show up in the clinic. The second type of waiting time arises when patients show up in the healthcare system, and since all healthcare resources such as physicians, nurses and all types of equipment are seized by other patients, patients have to wait in a line before the medical services they need to access. In this study, we are studying the second type of waiting times that occur when the patient attends the healthcare system.

In a situation where a patient must go through consecutive operations there might be a necessity to consider a minimal time lag or a minimum waiting time between two consecutive operations. This requirement can be addressed by including a minimal time lag constraint in the model.

Minimal time lags can be a transfer time from one department to another department or the minimum rest time after a medical test or a time to become ready to perform a surgery. On the other hand, patients do not like to wait for more than a maximum amount of time between two consecutive operations. For a patient, waiting for more than a maximum amount of time can cause discomfort or have some life threats in some cases (Prentice and Pizer, 2007). This requirement can be explained by adding a maximum time lag constraint to the model.

In addition, if the sum of these waiting times or cumulative waiting time in the healthcare system exceeds a certain amount of time, the patients may feel more inconvenience and being humiliated.

In addition, if there are several consecutive operations, the sum of individual waiting times between operations can become a big number and increase the access time to the necessary last operations and have some health risks for the patients; therefore, to tackle this noisome experience a constraint of the maximum total wait time of each patient must be considered to model the problem.

By examining all of these constraints, we developed a mixed integer programming model and corresponding constraint programming permutation flowshop scheduling model to minimize makespan. In our model, we considered minimal and maximal time lags between consecutive operations and the maximum waiting time for each patient.

This report is organized as follows: In chapter two, the healthcare optimization literature is reviewed and research gap is discussed. In chapter three, the developed mixed integer programming and constraint programming models, parameters and variables are explained. Chapter four is concentrated on presenting the developed algorithm to solve the model. After that, in chapter five, calculation results and sensitivity analysis are described. Finally, conclusions and directions for future research are presented in chapter six.

2 Literature Review

This chapter addresses different research works in the area of healthcare scheduling in five sections:

- In the first section, a review of healthcare service is presented.
- In the second section, general literature related to healthcare scheduling has been provided. Different challenges and opportunities that are discussed in the healthcare scheduling are addressed in this section.
- In the third section, literature related to single and multi-objectives is described.
- In the fourth section, the literature related to optimizing time, cost and utilization of resources is reviewed.
- In the fifth section, the literature related to stochastic and deterministic approach is reviewed.
- In the sixth section, literature related to inpatient and outpatient is reviewed.
- In the seventh section, the literature related to exact and approximate solution is studied.
- in the eights section, literature related to permuation flowshop is studied.
- In the last section, research gap and the contribution of this research is described.

The papers in this literature review include the ones that intend to model and solve the healthcare scheduling problem; the focus is on the research works that used integer programming techniques.

2.1 Service Quality in Healthcare

In this section, service quality within the concept of scheduling will be discussed. Service quality in the healthcare has been measured from different perspectives. Büyüközkan, Çifçi, and Güleryüz (2011) developed a fuzzy AHP methodology for measuring healthcare service quality. They investigated various aspects of service quality in healthcare systems, and one of the aspects of system performance was responsiveness that referred to the ability to provide operations and service on timeliness. Timeliness can be waiting time or total service time.

Singh and Prasher (2017), used fuzzy AHP to measure the service quality from patients' perspective; they concluded that reliability and trustworthiness are the most important factors from the patient's perspective in measuring healthcare quality.

2.2 General Healthcare Scheduling literature review

In this section, the literature related to research works that studied different aspects of scheduling in healthcare is discussed.

Gupta and Denton (2008) examined various aspects, challenges, and opportunities in the appointment scheduling. They considered three types of environments: First primary care, second specialty clinics and third surgeries and hospital stays; they argued that required services for patients in the primary care usually could be provided in the fixed amount of time. In such environments, service providers would break available time into time slots with the same size. Therefore, appointment scheduling problem becomes only the straightforward process of assigning appropriate time among available spots to the patients.

Gupta and Denton (2008) argued that in specialty care, service time has more fluctuations and making appointments is more complicated than primary care clinics. Because there is not any standard service time in primary care clinics and scheduler must consider an extra capacity to manage urgent appointment requests. They suggest that because of the variability of the procedure times, scheduling of surgical appointments can be more complicated. The requirement for making several appointments before surgery and the need to have some of the service providers at the same time, make surgical appointment scheduling more complicated.

Hall (2012) reviewed various aspects of healthcare systems scheduling concerning healthcare resources such as healthcare providers, rooms, facilities, supplies and organs to patients. Besides, he discussed capacity planning, nurse scheduling, patients appointments in ambulatory care, operating theatre planning and scheduling, appointment planning, scheduling in the outpatient procedure centers, the human and artificial scheduling system for operating rooms, bed assignment and bed management, queuing networks in healthcare systems, medical supply logistics.

Marynissen and Demeulemeester (2016) did a literature review for the problems of integrated hospital scheduling. They considered their review only for the integrated hospital scheduling problem (IHPS) which is the problem when a patient must visit multiple resources in hospital sequentially; however, their study and their methodology can be applied to other studies in the field of healthcare scheduling; they considered literature of integrated healthcare, patient flow, resource scheduling and the appointment scheduling. Marynissen and Demeulemeester (2016) wanted to find the relation of previous areas with IHPS and proposed the following steps for researchers who are researching in this area:

1. In the first step, researchers need to select the best setting for their research. The setting refers to the departments of hospitals that researchers want to study, the decision level at which decisions are being made and the patient mix that is being investigated; different departments may need different models such as flowshop, jobshop. Decision levels are divided into three categories of

strategic, tactical and operational levels; patient mix is the type of patient since decision making can be done based on inpatient, outpatient, mix of them or without mentioning these types of patients and emergency departments (Marynissen and Demeulemeester, 2016).

2. In the second step, researchers select the scope of their research. Hence, researchers define the assumptions, the departments of the hospital that they are going to consider, whether they want to include nurses or not or whether all patients have care time in each step (Marynissen and Demeulemeester, 2016).

3. In the third step, the model and the performance metrics to be optimized must be developed (Marynissen and Demeulemeester, 2016).

4. In the fourth step, researchers need to select how to optimize the problem which includes the strategy (online or offline) and scheduling methodology, patient classification method and patient preferences. Since in offline scheduling, the researchers have less time limitation compared to online scheduling, and there are more chance to use methods such as branch and bound which solution needs a fair amount of time while in online scheduling the optimization maybe limited to meta-heuristic methods. In addition, decision about methodology such as IP, MILP or metaheuristics is made at this stage (Marynissen and Demeulemeester, 2016).

5. In the fifth and the last step, researchers need to consider the appropriate approach for validating the model, and they need to decide for the model to be tested with fictional data or real data (Marynissen and Demeulemeester, 2016).

Figure 2.1 presents healthcare scheduling literature categories that are considered in this literature review:

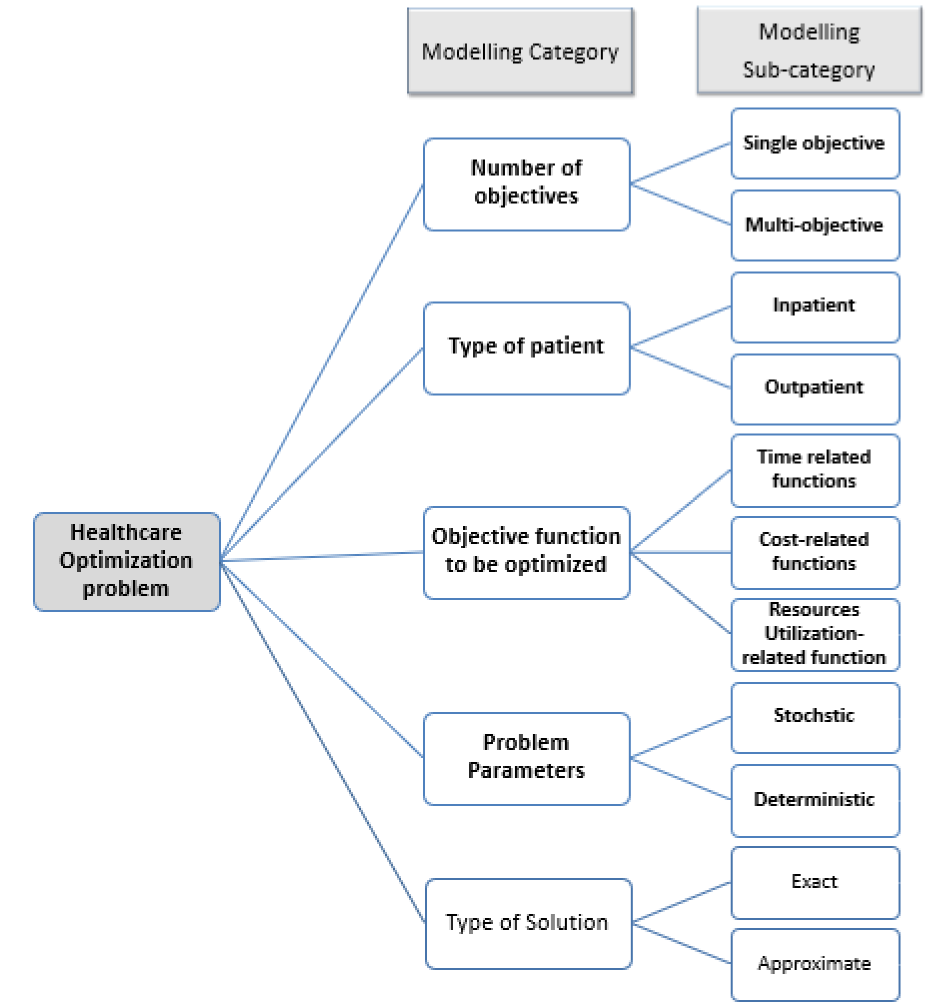

Figure 2.1: Healthcare Scheduling Literature Categories

2.3 Number of Objectives

Problems in healthcare scheduling can be categorized based on the number of objectives into single objective and multi-objective problems. Summary of the papers is presented in table 2.1.

Table 2.1: Number of Objectives

Category	Sub-category	Authors, Year	Objectives	Methodology
Number of Objective functions	Single Objective	Riise, Mannino, and Lamorgese, 2016	Minimize the penalty of unscheduled patients and waiting times of patient	Recursive logic-based decomposition
	Multi-objective optimization	Othman, Hammadi, and Quilliot (2015)	Minimize delays and costs	Fuzzy logic-based assignment procedure and the medical staff scheduling problem
		Karamyar, Sadeghi, and Yazdi (2016)	Minimize the total cost of the treatment of the patients and the completion time of the total patient services	Weighted sum for multi-objective optimization and solved their model using simulated annealing and bender-decomposition

2.4 Objective function to be optimized

Research studies in healthcare scheduling can be categorized based on the problem parameters; some studies include one or more time related functions, some include one or more cost-related functions and other studies include one or more function related to resource utilization. Summary of the papers is presented in table 2.2.

Table 2.2: Objective function to be optimized research works

Category	Sub-category	Authors, Year	Objectives	Methodology
Objective function to be optimized	Optimizing time	Augusto, Xie, and Perdomo (2010)	minimize the scheduled time of patients	Lagrangian relaxation-based algorithm
		Doulabi, Rousseau, and Pesant (2014)	maximize the sum of scheduled surgery time	constraint programming-based column generation approach for operating room planning and scheduling
		Hahn-Goldberg et al. (2014)	Maximizing the sum of scheduled surgery time	Constraint programming
		Latorre-Núñez et al. (2016),	Minimize makespan	Constraint programming and a genetic algorithm
	Optimizing cost	Fügener, Hans, Kolisch, Kortbeek, and Vanberkel (2014)	minimizing downstream costs including fixed costs, overcapacity costs, staffing costs and weekend staffing costs	Branch and bound algorithm, incremental improvement incremental improvement for optimizing scheduling costs
		Bard et al (2014)	Minimize the travelling cost among all patients plus the cost of providing treatment and minimize reimbursement and overtime payments costs respectively.	branch-price and cut and rolling horizon
		Laesanklang, Landa-Silva, and Salazar (2015)	Minimizing monetary costs, penalty cost of not satisfying workers and costs of violation of job assignment, work availability and region constraints	Decomposition for their mixed integer programming model
	Utilization of resources	Baesler and Sepúlveda (2001)	Minimize waiting time of patients, maximize utilization of chair, minimize closing time and maximize utilization of nurses	Goal programming and GA algorithm

2.5 Problem parameters

Research studies can have stochastic or deterministic problem parameters such as constraints, variables, and objective functions. Stochastic parameters can have a statistical distribution and deterministic parameters have a fixed value before solving the problem. Summary of research works that used deterministic approach is presented in the tables 2.3, 2.4, 2.5, 2.6.

Table 2.3: Problem Parameters selected research works between 1996-2000

Category	Sub-category	Authors, Year	Objectives	Methodology
Problem parameters	Deterministic approach	Luh and Hoitomt (1993)	Optimizing on-time delivery by minimizing tardiness	Lagrangian method
		Berrada, Ferland, and Michelon (1996)	They considered minimization objectives to satisfy the following soft constraints: 1. Stopping long stretch by restricting the number of consecutive working days 2. No working day between two days off 3. There must be a minimum number of nurses supervising 4. Allow non-working days together and working days together 5. Excess or lack of regular personnel must be assigned during the weekdays uniformly	Tabu search for multi-objective optimization
		Jaumard, Semet, and Vovor (1998)	1. Minimizing salary costs 2. Minimizing nurse preferences 3. Minimizing experienced, and less experienced staff balance	They proposed a binary column generation model and solved their model by applying a resource constraint shortest path auxiliary problem
		Beaulieu, Ferland, Gendron, and Michelon (2000)	They considered the following goals for their model: 1. There are predetermined hours of working for each physician 2. All physicians must work certain types of shifts such as night shifts or evening shifts evening and follow-up shifts	branch and bound algorithm

Table 2.4: Problem Parameters selected research works between 2000-2010

Category	Sub-category	Authors, Year	Objectives	Methodology
Problem parameters	Deterministic approach	Topaloglu (2006)	Minimizing deviations of the following soft constraints: 1. Residents should not be in night shifts in two following nights 2. Residents should not have three consecutive night shifts in their schedule 3. Residents should not have connected three-day shifts in their schedule 4. Residents should not have more than three successive day shifts in their schedule 5. Residents should be off at minimum one weekend at minimum one weekend 6. Supervising resident positions should be allocated reasonably	Goal programming was used to solve soft and hard constraints. She determined the weight of deviations of soft constraints using AHP
		Chern, Chien, and Chen (2008)	1. Minimize waiting time of examinees 2. Minimize waiting time of doctors	In their heuristic algorithm, first, they minimized the waiting time of examinee, then the waiting time of doctors is minimized.
		Cardoen, Demeulemeester, and Beliën, 2009	1. Minimizing the sum of the start time of children surgeries 2. Minimizing the sum of the start time of prioritized patients 3. Minimizing the number of travels 4. Minimizing the number of required recovery periods after the surgical daycare center finish time 5. Minimizing the maximum number of beds that are engaged in the recovery phase I 6. Minimizing the maximum number of beds that are engaged in the second recovery phase	Branch and bound
		Beliën, Demeulemeester, and Cardoen, 2009	1. Level the resulting bed occupancy at the hospitalization units. 2. Assign an operating room only to a group of surgeons who have the same expertise 3. Provide the most possible repetitive and manageable master surgery schedule	Applied a simulated annealing metaheuristic with quadratic optimization techniques

Table 2.5: Problem Parameters selected research works between 2011-2015

Category	Sub-category	Authors, Year	Objectives	Methodology
Problem parameters	Deterministic approach	Sadki, Xie, and Chauvin (2011)	1. Minimize the waiting time of the patient 2. Minimize the makespan	They developed one heuristic based on the Lagrangian Relaxation method and a local optimization heuristic. They used weighted sum to solve the multi-objective optimization part.
		Petrovic, Morshed, and Petrovic (2011)	Minimizing the average waiting time of the patient and the average breach's waiting time	multi-objective genetic algorithms
		Lim, Mobasher, and Côté (2012)	Minimize the following objectives: 1. The costs of nurse labor 2. Dissatisfaction of patient 3. The idle time of nurses And maximize the satisfaction of nurses	Used goal programming to model and solve their multi-objective optimization problem
		Meskens, Duvivier, and Hanset (2013)	Minimize makespan and overtime hours and maximize affinities among staffs who want to work at the same shift	The used constraint programming
		Zhao, Chien, and Gen (2015)	Minimize the entire waiting time of each inpatient and makespan.	They used NSGA2 at the rst stage and adaptive weight Genetic Algorithm
		Liang and Turkcan (2015)	Minimizing the waiting time of patients and the waiting times of the staffs	Three algorithms: Pareto optimal set, weighted sum method and € -constraint method
		Elomri, Elthlatiny, and Sidi Mohamed (2015)	Minimize the following objectives at each department at weekends: 1. The difference between average target and assignments for each resident at each duty at each department 2. Minimize the difference between average target and assignments for each resident at each duty	Used CPLEX to solve their model

Table 2.6: Problem Parameters selected research works in recent years

Category	Sub-category	Authors, Year	Objectives	Methodology
Problem parameters	Deterministic approach	Smalley and Keskinocak (2016)	Minimizing the following objectives: 1. Service demand deviations 2. Violation from identical assignments for residents in the same group 3. Denied resident requests for assignment of the services in special months And maximizing the number of residents that are allocated to their services of interest	Integer programming using CPLEX
		Lim, Mobasher, Bard, and Najjarbashi (2016)	Minimizing shortage of demand, overtime and idle time.	AHP and column generation for determining the weights and oen heuristic based on column generation
		Bard, Shu, Morrice, Leykum, and Poursani (2016)	1. Maximizing the balance of patients that are seen in each working session of the clinic and each week. 2. Minimizing the number of changes of the current assignments for each rotation	Greedy-based algorithm

Summary of selected papers is presented in the table 2.7.

Table 2.7: Selected research works that used stochastic parameters

Category	Sub-category	Authors, Year	Objectives	Stochastic optimization method
Problem parameters	Stochastic approach	Azcárate, Mallor, and Gafaro (2008)	1. Minimizing cost 2. Minimizing the rate of turned away patients (in percentage) 3. Maximizing the quality factor related to the quality of the service of doctor that can increase	metaheuristic and simulation optimization as solution method with the patient arrivals as random variables with Poisson distributions and each patient may go to another path with a specific probability.
		Liang, Turkcan, Ceyhan, and Stuart (2015)	1. Minimizing the gap between the minimum and the maximum number of engaged chairs to balance the utilization of the chairs. 2. Minimizing the gap between the minimum and the maximum number of occupied exam rooms at each timeslot	Service time and treatment duration are stochastic variables and solved their model for two objectives. At the first step they solved the model for the first objective and then used the result for the probability matrix and simulation.
		Bikker, Kortbeek, van Os, and Boucherie (2015)	1. Minimizing the access time lower bound and determining the lower bound for each patient based on the type of the patient, location of treatment and referral day 2. Minimizing the gap between daily supply and daily demand	ILP and discrete event simulation.
		Saremi, Jula, ElMekkawy, and Wang (2015)	1. Minimizing patients waiting time 2. Minimizing the facility completion time of (makespan)	They used a bi-criteria simulation-based optimization solution multi-objective Tabu search.

2.6 Type of the patient

Research studies in the healthcare scheduling can be categorized based on the type of the patient under study; some studies focus on inpatient and some on outpatient and each one may need different criteria. Summary of the papers is presented in the tables 2.8.

Table 2.8: Type of the patient

Category	Sub-category	Authors, Year	Objectives	Methodology
Type of patient	In-patient	Matis, Farris, McAllister, Dunavan, and Snider (2015)	minimize the waiting of inpatients who wanted to discharge from the hospital	Relaxed four of their objectives
	Outpatient	Srinivas and Ravindran (2018)	Maximizing net revenue, minimizing waiting time of the patient and using the resource as efficient as possible	predictive model using machine learning techniques

2.7 Type of the solution

One category that can be considered for the healthcare scheduling research studies is the type of solution the researchers found. In this section research works that used exact and the research works that used approximate method are investigated. Summary of the papers is presented in the table 2.9.

Table 2.9: Type of the Solution

Category	Sub-category	Authors, Year	Objectives	Methodology
Type of Solution	Exact	Chandra, Liu, He, and Ruohonen (2014)	minimizing maximum scheduling cost	Binary branch and bound algorithm
	Approximate	Guido and Conforti (2016)	Maximizing the following objectives: The quantity of scheduled patient The total clinical priority of the scheduled patients The overall priority if a patient is those who scheduled in the planning period And minimizing the following objectives: The underutilization of the allocated OR blocks The total cost of overtime	Hybrid genetic algorithm to solve a planning and scheduling problem

2.8 Permutation Flowshop

In any scheduling problem that a set of n jobs are scheduled on m machines, and all have the same flow on all machines, the problem is a flowshop one; in a specific case of flowshop problems that all jobs have the same orders on all machines the problem is considered a permutation flowshop problem (Nagano, Ruiz, and Lorena, 2008). Johnson (1954) was one of the first researchers who studied the flowshop problem and proposed a heuristic algorithm for flowshop problems with two machines or for the flowshop problem with three machines, contingent on specific processing times.

The permutation flowshop scheduling problem (PFSP) with more than three machines is NP-hard (Garey, Johnson, and Sethi, 1976); therefore, it is impossible to find an approach for finding exact solutions for the permutation flowshop problem; however, many heuristics and meta-heuristics have been developed to find a quality solution (Nawaz, Enscore Jr, and Ham, 1983; Rahman, Sarker, and Essam, 2015; Rajendran, Rajendran, and Leisten, 2017). Heuristics can be categorized into two categories: Constructive heuristics that build a feasible solution based on specific rules and improvement heuristics that improves the feasible solution (Govindan, Balasundaram, Baskar, and Asokan, 2017). Nawaz et al. (1983) proposed Nawaz-Enscore-Ham (NEH) heuristic algorithm algorithm that was one of the most efficient heuristics to solve permutation flowshop problems. Most of heuristics are adjusted types of NEH algorithm (Fernandez-Viagas, Ruiz, and Framinan, 2017). For PFSP many meta-heuristics have been developed and algorithms such as Ruiz and Stützle (2007) could improve previous meta-heuristics (Fernandez-Viagas et al., 2017).

A more specific type of PFSP that has many applications in different industries, is the problem with minimum and maximum time lags. One example can be food industry that there must be a maximal time lag after finishing of cooking and before the start of chilling down (Hodson, Muhlemann, and Price, 1985).

In the healthcare setting, maximal time lags can be preference of patients that do not want to wait more than specific time. Minimal time-lags can be transport time in both manufacturing and healthcare setting. Fondrevelle, Oulamara, and Portmann (2006) proved that the problem with minimal and maximal time lags is strongly NP-hard; they developed a branch and bound algorithm to solve m machine PFSP with minimal and maximal time lags with the objective of minimizing makespan and proposed several lower and upper bounds for the problem. Hamdi and Loukil (2011) proposed a genetic algorithm to solve m machine PFSP with time lags. They adopted the algorithm to calculate the minimum of makespan for a given permutation π from Fondrevelle et al. (2006) paper. Dhouib, Teghem, and Loukil (2013) proposed several simulated annealing algorithms for minimizing hierarchically the number of tardy jobs and makespan. Wang, Huang, and Li (2018) proposed a two-stage constructive heuristic to solve permutation flowshop with minimal and maximal time lags and tested their algorithm for small-scale problems.

2.9 Research Gap

In this section, research gaps have been highlighted. In the literature, different aspects of scheduling in healthcare systems have been discussed.

One area that needs more research is optimizing cost in healthcare systems. The cost is critical for healthcare system providers and minimizing cost can improve the services for patients. Minimization cost of healthcare systems because of minimizing the cost of nurses and/or physicians is an area that has not been studied. Cost factor can be minimized along with minimizing of makespan or minimizing waiting time of patients.

Another area that has not been examined in the literature is minimizing of waiting time and/or makespan with a mix of outpatient and inpatient that use the same resource in healthcare systems as a multi-objective problem with the Pareto front. This can be a multi-objective optimization function with one objective for each type of patients and one objective for waiting time of staffs. One other area that has not been considered in the literature is considering the whole problem as a bi-objective problem. One objective is optimizing the cost of staffs such as nurses, residents and another objective is minimizing waiting time of patients.

Based on the healthcare literature, there is a lack of enough research that consider minimization of C_{max} with considering waiting time of patients. The problem under consideration in this thesis, is minimizing waiting times before each job. Throughout this document, the word job refers to each patient and machines refers to the provider's resources such as physicians, nurses and staffs.

The general assumptions of the problem are as follows: *a*) the problem is assumed to be a permutation flow shop problem in which preemption is not allowed; *b*) All jobs have a predefined order of execution; *c*) The processing times are assumed to be deterministic, and it is assumed that the jobs arrive on time; *d*) each job must be processed by a maximum of one machine at a time. Also machines can process at most one operation at any moment in time; *e*) all of the processing times are nonzero and positive; *f*) it is assumed that in each day there is a fixed number of hours and intermediate storage is unlimited.

This problem can be denoted as $F_m||C_{max}$; note that $F_m||C_{max}$ does not consider any limitations between the waiting times of the operations of the jobs. Although this is how the healthcare providers treat the patients, long waiting times between the consecutive operations of the care result in reduced satisfaction of the care receivers.

Figure 2.2 presents a healthcare system in which there are three resources. Once the first patient has completed his/her work at machine one, he/she needs to wait for $(t_2 - t_1) + (t_4 - t_3)$. The proposal of this thesis is that this waiting time should not exceed a pre-specified amount. To consider this constraint, the problem of $F_m||C_{max}$ may be converted to $F_m|T_{ij}|C_{max}$ (Graham, Lawler, Lenstra, and Kan, 1979).

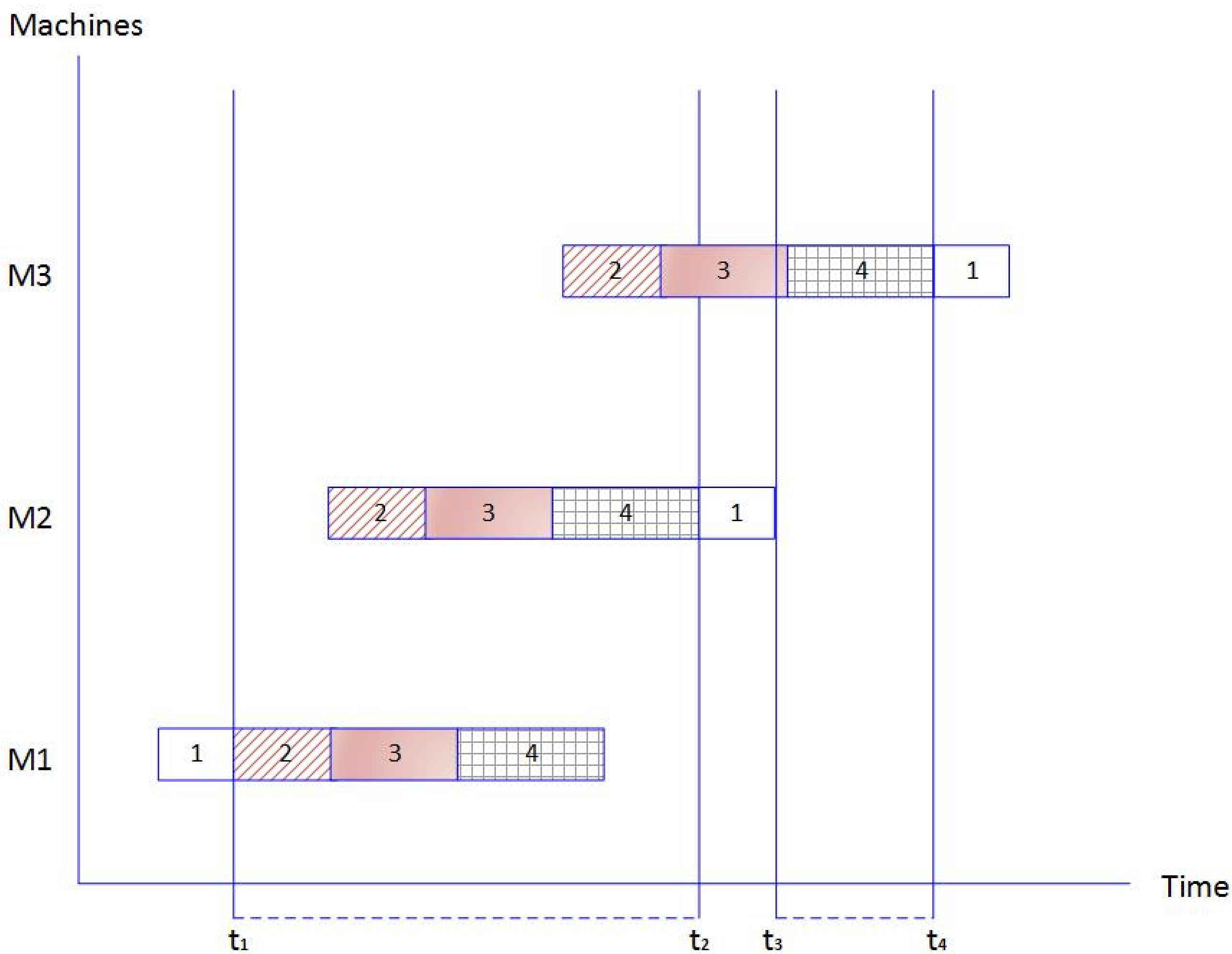

Figure 2.2: Waiting time of J_1, i.e., patient one

$F_m|T_{ij}|C_{max}$ limits the waiting times between the operations i and $i+1$ of the job j to a minimum of T^1_{ij} and a maximum of T^2_{ij}. In other words, $F_m|T_{ij}|C_{max}$ considers a minimum and a maximum bound for the waiting time of patient j between the operations i and $i+1$ planned for his/her visit; the minimum bound is shown by T_{ij}^1, and the maximum bound is represented by T_{ij}^2. T_{ij}^1 can be obtained from hospital and T_{ij}^2 can represent patient preference and can be obtained from the patients. A real world example can be a person who goes to meet a dentist. Many patients in the dental clinic go through the same process. There are typically four steps while visiting a dentist. At the first step, the patient meets reception; then dental hygienist performs initial checking. An x-ray is taken after that. Then cleaning is done and finally, the dentist examines the patient. Before each step, usually there some delays or waiting times for the patients. This is an example of permutation flowshop in healthcare. The job represents each patient, and in each step, every patient deals with a resource such as a hygienist or receptionist in which each resource is a machine. In the dental clinic, before each step, there might be delays. These delays are waiting times of the patients that are modeled in this research work.

3 Problem description

In this section, the problem, variables and parameters are presented. The problem formulation is presented based on mixed integer linear programming and Constraint Programming formulations. First, mixed integer linear programming formulation is presented and then constraint programming is presented.

3.1 Mixed Integer Linear Programming Model

It is assumed that there are m resources or machines to process n jobs or patients. The variables and parameters of the model are defined below:

m	Number of machines
n	Number of jobs
J_j	Job j
p_{ij}	Processing time of ith operation of J_j.
C_{max}	Completion time of the last job before leaving the system.
x_{ijk}	A binary variable; its value is equal to one if J_k is scheduled immediately after J_j on machine i, and zero otherwise.
s_{ij}	Starting time of operation i of J_j.
T_{ij}^1	Minimum time lag between the completion time of operation i and starting time of operation $(i+1)$ of J_j.
T_{ij}^2	Maximum time lag or maximum allowed waiting time between the completion time of operation i and starting time of operation $(i+1)$ of J_j.
$F_j = s_{mj} + p_{mj}$	Finish time of the last operation of J_j.
w_{ij}	The waiting time between the completion time of the ith operation and the starting time of the $(i+1)$th operation of J_j.
$w_j = \sum_{i=1}^{m-1} w_{ij}$	Sum of waiting times between all of the operations of J_j.
W	Maximum acceptable waiting time for all of the operations of J_j.

Equations (3.1)-(3.17) denote the model formulation:

$$\min C_{max} \tag{3.1}$$

$$C_{max} \geq s_{mj} + p_{mj} \qquad j = 1, \dots, n \tag{3.2}$$

$$\sum_{j=1}^{n} x_{ijk} \leq 1 \qquad i = 1, \dots, m \quad k = 1, \dots, n \tag{3.3}$$

$$\sum_{k=1}^{n} x_{ijk} \leq 1 \qquad i = 1, \dots, m \quad j = 1, \dots, n \tag{3.4}$$

$$\sum_{j=1}^{n} \sum_{k=1}^{n} x_{ijk} = n - 1 \qquad i = 1, \dots, m \tag{3.5}$$

$$x_{ijk} + x_{ikj} \leq 1 \qquad i = 1, \dots, m \quad j, k = 1, \dots, n \tag{3.6}$$

$$x_{ijj} = 0 \qquad i = 1, \dots, m \quad j = 1, \dots, n \tag{3.7}$$

$$s_{ik} + M(1 - x_{ijk}) \geq s_{ij} + p_{ij} \qquad i = 1, \dots, m \quad j, k = 1, \dots, n \tag{3.8}$$

$$s_{(i+1)j} - \left(s_{ij} + p_{ij}\right) = w_{ij} \qquad i = 1, \dots, m-1 \quad j = 1, \dots, n \tag{3.9}$$

$$T_{ij}^{1} \leq w_{ij} \qquad i = 1, \dots, m-1 \quad j = 1, \dots, n \tag{3.10}$$

$$T_{ij}^{2} \geq w_{ij} \qquad i = 1, \dots, m-1 \quad j = 1, \dots, n \tag{3.11}$$

$$\sum_{i=1}^{m-1} w_{ij} = w_j \qquad j = 1, \dots, n \tag{3.12}$$

$$w_j \leq W \qquad j = 1, \dots, n \tag{3.13}$$

$$x_{ijk} = x_{(i+1)jk} \qquad i = 1, \dots, m-1 \quad j, k = 1, \dots, n \tag{3.14}$$

$$w_{ij} \geq 0 \qquad i = 1, \dots, m-1 \quad j = 1, \dots, n \tag{3.15}$$

$$s_{ij} \geq 0 \qquad i = 1, \dots, m \quad j = 1, \dots, n \tag{3.16}$$

$$x_{ijk} \in \{0, 1\} \qquad i = 1, \dots, m \quad j, k = 1, \dots, n; \tag{3.17}$$

Constraint (3.1) describes objective function. Constraint (3.2) ensures that C_{max} is feasible. Constraints (3.3) to (3.7) ensure that all jobs appear only once in the sequence. Constraints (3.8) ensures that the jobs do not overlap. Constraint (3.9) calculates the waiting time between the operations i and $(i + 1)$ of J_j. Constraints (3.10) and (3.11) guarantee that the waiting time between any two consecutive operations of J_j falls within the required minimum and maximum time lags. Constraints (3.12) calculates the total waiting time of any job j. Constraint (3.13) ensures that the waiting time of each job is not more than the maximum acceptable waiting time threshold. Finally constraint (3.14) ensures that the problem is permutation problem.

Fondrevelle et al. (2006) have proved that the PFSP with minimal and maximal time-lags is NP-hard, and in the current problem if W is considered a very large number, then the constraint number 13 become non-binding and the problem can be converted to a PFSP with minimal and maximal time-lags.

3.2 Constraint Programming Model

It is possible to develop a constraint programming model for the problem to compare the performance of the constraint programming model with that of $MILP$. The constraint programming model has been developed and the model and constraints will be presented. The variables and parameters are described as follows:

m	Number of machines
n	Number of jobs
J_j	Job j

p_{ij}	Processing time of ith operation of J_j.
C_{max}	Completion time of the last job before leaving the system.
S_{ij}	Interval variable that denotes the start time and end time of operation i of J_j. Each interval variable can be used to represent an activity in scheduling problems. Interval variables can be used to describe jobs in scheduling problems. These interval variables have three characteristics: start time, end time and duration of the Job; therefore, we can use this variable in our CP model
T^1_{ij}	Minimum time lag between the completion time of operation i and starting time of operation $(i+1)$ of J_j.
T^2_{ij}	Maximum time lag or maximum allowed waiting time between the completion time of operation i and starting time of operation $(i+1)$ of J_j.
w_{ij}	The waiting time between the completion time of the ith operation and the starting time of the $(i+1)$th operation of J_j.
$w_j = \sum_{i=1}^{m-1} w_{ij}$	Sum of waiting times between all of the operations of J_j.
$SequenceVar_i$	Sequence variable that denotes the order of jobs in all machines. The variable is used to ensure the order of jobs in all machines are the same and there is a permutation flowshop problem
W	Maximum acceptable waiting time for all of the operations of J_j.

Equations (3.18)-(3.29) denote the model formulation:

$$\min C_{max} \tag{3.18}$$

$$C_{max} \geq End\ \ of\ \ S_{mj} \qquad j = 1, \ldots, n \tag{3.19}$$

$$No\ \ Overlap\ \ (S_{ij}) \qquad i = 1, \ldots, m \quad j = 1, \ldots, n \tag{3.20}$$

$$End\ \ of\ \ (S_{ij}) + T^1_{ij} \leq Start\ \ of\ \ (S_{(i+1)j}) \qquad i = 1, \ldots, m-1 \quad j = 1, \ldots, n \tag{3.21}$$

$$Start\ \ of\ \ (S(i+1)j) - End\ \ of\ \ (S_{ij}) \leq T^2_{ij} \qquad i = 1, \ldots, m-1 \quad j = 1, \ldots, n \tag{3.22}$$

$$Size\ \ of\ \ (Sij) = p_{ij} \qquad i = 1, \ldots, m \quad j = 1, \ldots, n \tag{3.23}$$

$$No\ \ Overlap\ \ (SequenceVar_i) \qquad i = 1, \ldots, m \tag{3.24}$$

$$Same\ \ Squence(SequenceVar_1, SequenceVar_i) \qquad i = 1, \ldots, m \tag{3.25}$$

$$Start\ \ of\ \ (S(i+1)j) - End\ \ of\ \ (S_{ij}) = w_{ij} \qquad i = 1, \ldots, m-1 \quad j = 1, \ldots, n \tag{3.26}$$

$$\sum_{i=1}^{m-1} w_{ij} = w_j \qquad j = 1, \ldots, n \tag{3.27}$$

$$w_j \leq W \qquad j = 1, \ldots, n \tag{3.28}$$

$$w_{ij} \geq 0 \qquad i = 1, \ldots, m-1 \quad j = 1, \ldots, n; \tag{3.29}$$

The objective function is described using the constraint (3.18). Constraint (3.19) ensures that C_{max} is feasible. Constraint (3.20) ensures that there is no overlap among processing of jobs in machines and each machine processes one job at each time. Constraint (3.21) ensures that minimal time lag constraint is feasible. Feasibility of the maximal time-lag constraint is ensured using the constraint (3.22). Processing time of the (i)th operation of J_j is calculated using constraint (3.23). Constraint (3.24) ensures that sequence variables do not overlap. The constraint (3.25) ensures that all jobs in all machines have the same permutation and the model is a permutation flowshop model. The constraint 3.27 calculates waiting time of each job. The waiting time of each job is equal to the sum of the waiting times of a job in all machines. The constraint (3.28) ensures that the waiting time of each job is less than or equal to the maximum acceptable waiting time of each job. Finally, the constraint (3.29) ensures that the waiting time of each operation is non-negative.

4 Proposed Solution Algorithm

In this chapter, proposed algorithm is presented. The algorithms that are used include genetic algorithm, Lagrangian relaxation and the intensification algorithm.

4.1 Pseudo-code of the algorithm

Procedures required before beginning the genetic algorithm are explained as follows: Define Class Chrome for keeping population properties and class problem for keeping problem parameters.

Getting required number of iterations.

Reading problem parameters and setting size of vectors.

Step 01: Giving initial values to the problem parameters.

While (number of generations<iterations)

Step02: Assign the order of jobs in the population and compute fitness value $Cmax$.

Step 03: If crossover probability is greater than the generated random number then do crossover.

Step 04: If mutation probability is greater than the generated random number then do mutation.

Step 05: Selecting top 50% of chromosomes from new population and top 50% of chromosomes from the old population.

Step 06: Go to Step 03

End

Lagrangian relaxation is performed for the model. The objective function would become to the form of $min\ C_{max}+ \lambda \sum_{j=1}^{n} (w_j - W)$. In the objective function, the penalty term $(w_j - W)$ is equal to zero if for each j $w_j < W$. The coefficient λ can be considered as the patient dissatisfaction of waiting. In the calculations, this coefficient is considered to be equal to one; however, in real world patients may give a bigger value to this coefficient.

4.2 Permutation Flow shop Scheduling based on GA

4.2.1 Encoding of Flowshop Problem

Each solution is represented by a two-dimensional string. If there exist m machines and n jobs in the problem, first dimension represents the machines and second dimension represents the jobs. Therefore, if there are five jobs in the sequence, a sample chromosome for the machine i can be presented in the form of $\{54321\}$. This representation suggests that job five gets processed first in all machines and then fourth job is processed and the order continues till job one. Each job is called a gene and a string of genes or a sequence is called a chromosome.

4.2.2 Initialization

The first step would be initialization. At this step, a number of populations or randomly generated sequences are generated. In each population, a number between one to J is assigned to each job, and the number is the order of job at initialization. Therefore, the job that is assigned randomly number one would be the first job and so on. The number of generated populations is equal to population size.

4.2.3 Calculating values

For updating values, first step is to update values for the first job at the first machine. Then the values for the other jobs at the first machine are assigned based on the predetermined orders of the jobs. Start time of the second job is equal to the finish time of the first job at the first machine plus minimum waiting time. After that, the finish time of each job is equal to the start time of the previous job plus corresponding minimum waiting time.

At the second step, values of waiting time, start time and completion time related to all machines after the first machine would be calculated. Therefore, start time of the first job in machine i where $i > 1$ is equal to finish time of the job on previous machine plus minimum waiting time. For each job j after the first job in i where $i > 1$, start time is equal to finish time of previous job (job $_{j-1}$) based on the predefined order. If finish time of previous job job$_{j-1}$ in the machine i where $i > 1$ is less than finish time of job j in machine$_{i-1}$+minimum required waiting time of the operation, then start time of job j on machine i is equal to the finish time of job j in machine$_{i-1}$+minimum required waiting time of the operation.

In case job j in machine i has a start time greater than the finish time of the job j in machine$_{i-1}$+minimum waiting time of the operation, then the waiting time of corresponding operation would be updated with the new value which is greater than minimum waiting time. In case waiting time of a job is more than the maximum waiting time T_{ij}^2 then the start time of first job in the sequence will increase until the waiting time gets equal to the T_{ij}^2.

4.2.4 Selection

At the next step, selection would be performed. To do the selection, at the first, chromosomes are sorted based on the fitness values of their chromosomes which is their $Cmax$. We use a selection method based on the Truncation selection method (Blickle and Thiele, 1995). These chromosomes are the ones that were modified by crossover and mutation. New population if made up of top 50% of previous population plus top 50% of population after crossover and mutation. At the first iteration, these two populations are the same.

4.2.5 Cross over

The next step would be crossover. Crossover would be implemented if generated random number is less than crossover probability. To do crossover a number of parents should be selected and to do that, each chromosome with the index i, does crossover with chromosome with the index $i + 1$ and these two chromosomes are called parents.

A crossover point would be randomly selected. It is a number between one to n or number of jobs. After selecting the point, all jobs between one to crossover-point would be copied into one new chromosome in the next population. This chromosome is called offspring. The remaining vacant positions for the jobs would be filled by copying jobs after crossover-point to the end of chromosome from the other parent.

Therefore, if each parent one has eight genes and it has a sequence of jobs equal to $\{12345678\}$, and the parent two has job sequence equal to $\{43218765\}$ and randomly generated crossover-point is equal to four, then the selected genes from the first parent of the first offspring would be $\{1234|\}$ and the second part of chromosome from the second parent would be $\{8765|\}$. After assigning the values related to the second part, the offspring is equal to $\{1234|8765\}$ and the same procedure but with the jobs after cross-over point in the parent one and jobs before cross-over point in the parent two gives the second offspring would be $\{4321|5678\}$. In case there are some common jobs between the selected parts for an offspring, for each common job, crossover point would shift one point to the right.

4.2.6 Mutation

After crossover, mutation would be performed. In each iteration, for mutation, a random number would be generated which is called the random job. It suggests the position of the job that the corresponding order must be swiped with another job in the machine. The next job would be found using the deduction of random job from number of jobs. Therefore, if there are eight jobs and random job is equal to three, then the third gene would be selected and it is swiped with the gene $= 8 - 3 = 5$. This process continues until the iterations finish.

4.3 Intensification

As size of problem increases solving scheduling problem with m machines and n jobs become more difficult and complex; however, the output of a genetic algorithm a problem with n jobs and m machines where $n = n_1 + n_2 + .. + n_k$ can be improved by replacing optimal order of jobs for the problem of (m, n_i) where $i = 1, 2, .., k$. An algorithm called intensification is designed and implemented to take advantage of this property. In the algorithm, the number of jobs for the small problem is defined and then for all jobs, starting from job one to number of jobs in the small problem is optimized. This process continues until the last job is considered. The flowchart of the algorithm is presented at Figure 4.1.

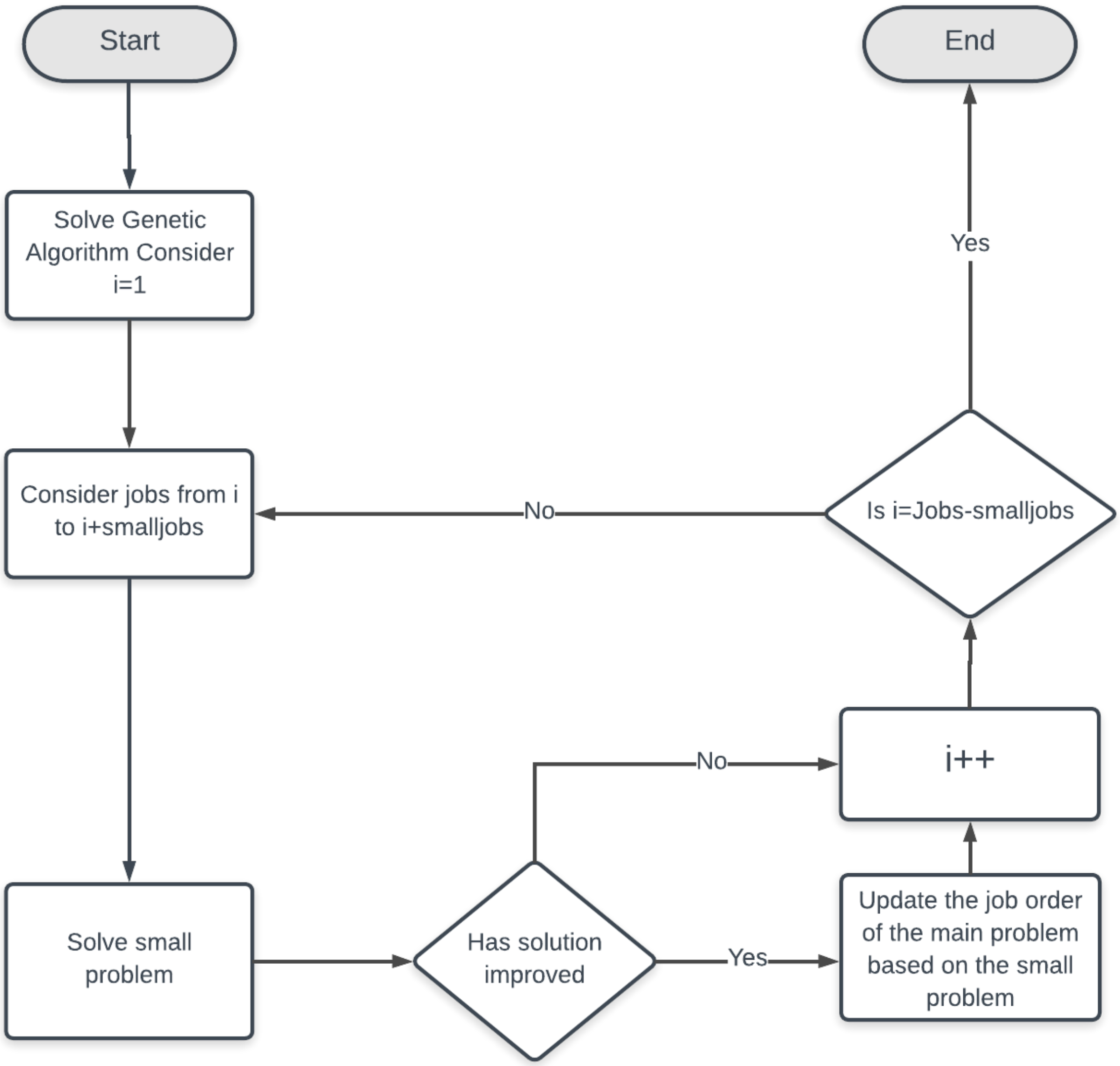

Figure 4.1: Intensification Algorithm

5 Computational Results

5.1 Exact solutions

In this part exact solutions of the problem are presented. Table 5.1 presents the output for exact solution. The results are obtained after running both MILP and CP models in CPLEX 12.7.1. The outputs suggest that CP outperforms MILP in terms of solution time and the solution time is quicker in CP. Processing times are generated from a uniform integer distribution of $[20, 50]$. Minimal times are generated randomly from uniform integer distribution of $[0, 7]$ and maximal time lags are generated from the the uniform integer distribution of $[T^1_{ij}, 14]$. Results are the average of five runs that are rounded to the nearest integer number. In each run, if solutions take more than one hour of solving time then the letter L is put in the table.

Table 5.1: CP and MILP results of the outputs of exact solutions of MILP and CP models

	MILP		CP		
Problem	Objective function	Time (in seconds)	Objective function	Time (in seconds)	W
(5, 12)	L		602	77.205	29
(3, 15)	L		613	8.850	30
(5, 15)	L		702	13.764	40
(5, 10)	519	809.901	519	3.954	36
(5, 5)	338	1.506	338	1.714	18

5.1.1 Comparison of solutions for the permutation flowshop problem

The permutation flowshop model can be obtained after relaxing the constraints related to minimum and maximum time-lags. The MILP model for the resulting permutation flowshop with the time-lags model can be obtained after dropping the constraints (3.9), (3.10), (3.11), (3.12), (3.13) and (3.15) from the proposed MILP model.

The resulting constraint programming permutation flowshop with time lags model can be obtained after dropping the constraints (3.21), (3.22), (3.27), (3.28) and (3.29) from the proposed constraint programming model.

The solutions of the MILP and CP permutation flowshop models are compared with 13 instances of the test problems provided by Vallada, Ruiz, and Framinan (2015). The test problems have a lower and upper bound. Vallada et al. (2015) ran the test problems with the maximum run-time of machines*jobs*60/1000 seconds. In this section, the solving time is set to the machines*jobs*60/1000 seconds. The results are presented in the table 5.2. The columns under VRF present the upper bound and lower bound reported by Vallada et al. (2015). The columns under MILP reports upper bound and lower bound found by MILP model after running the model in CPLEX. The columns under CP report upper bound and lower bound found by the constraint programming model.

Table 5.2: Comparison of results for the model without time-lags

Problem (Machines, Jobs)	MILP		CP		VRF		Run time (in seconds)
	Upper Bound	Lower Bound (non-integer)	Upper Bound	Lower Bound	Upper Bound	Lower Bound	
(5,10)	767	387	695	573	695	523	3
(5,20)	1246	353	1192	1134	1192	1095	6
(5,30)	1938	933	1805	1805	1805	1727	2.42
(5,40)	2535	444	2396	2365	2396	2292	12
(5,50)	3241	435	3055	3035	3055	2970	15
(5,60)	3370	405	3121	3121	3121	3074	36
(10,10)	1118	699	1097	926	1097	797	6
(10,20)	1672	706	1541	1372	1532	1290	12
(10,30)	2172	631	1987	1821	1944	1721	18
(10,40)	2941	754	2582	2341	2480	2258	24
(10,50)	3731	677	3097	2825	2926	2746	30
(10,60)	4106	642	3517	3359	3435	3256	36
(20,60)	5285	1279	4573	3914	4221	3764	72

Upper bounds are presented in figure 5.1. The outputs suggest that CP outperforms MILP for all problems. For problems with the small number of jobs and machines both VRF and CP provide the same upper bound; however, for the problems with ten machines and more than thirty jobs VRF provides lower upper bound. For the minimization problem, this is desired.

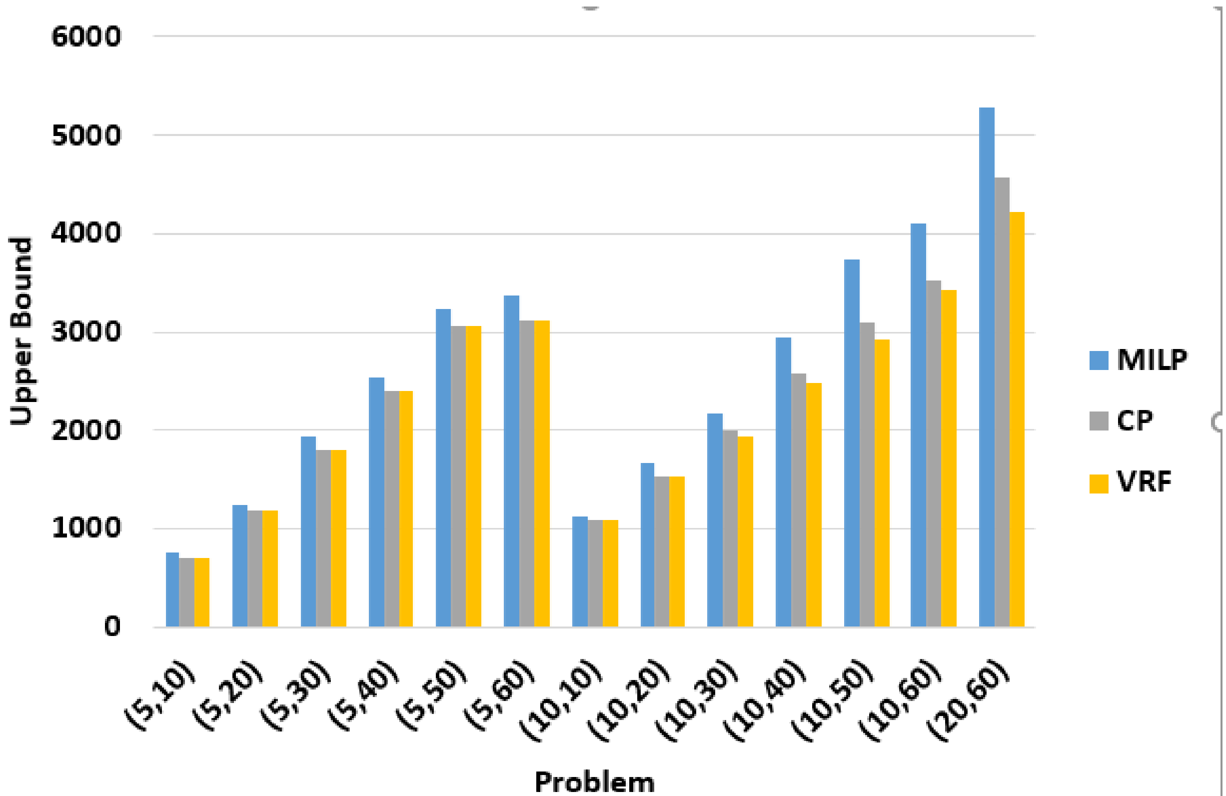

Figure 5.1: Comparison of upper bounds of the CP, MILP and VRF

On the other hand, comparison of lower bounds can be of interest. As figure 5.2 suggests, CP provides a higher value of upper bound compared to both VRF and MILP. Lower bounds of VRF provided by Vallada et al. (2015) are very close to the CP, but all of the upper bounds of the CP are higher than VRF.

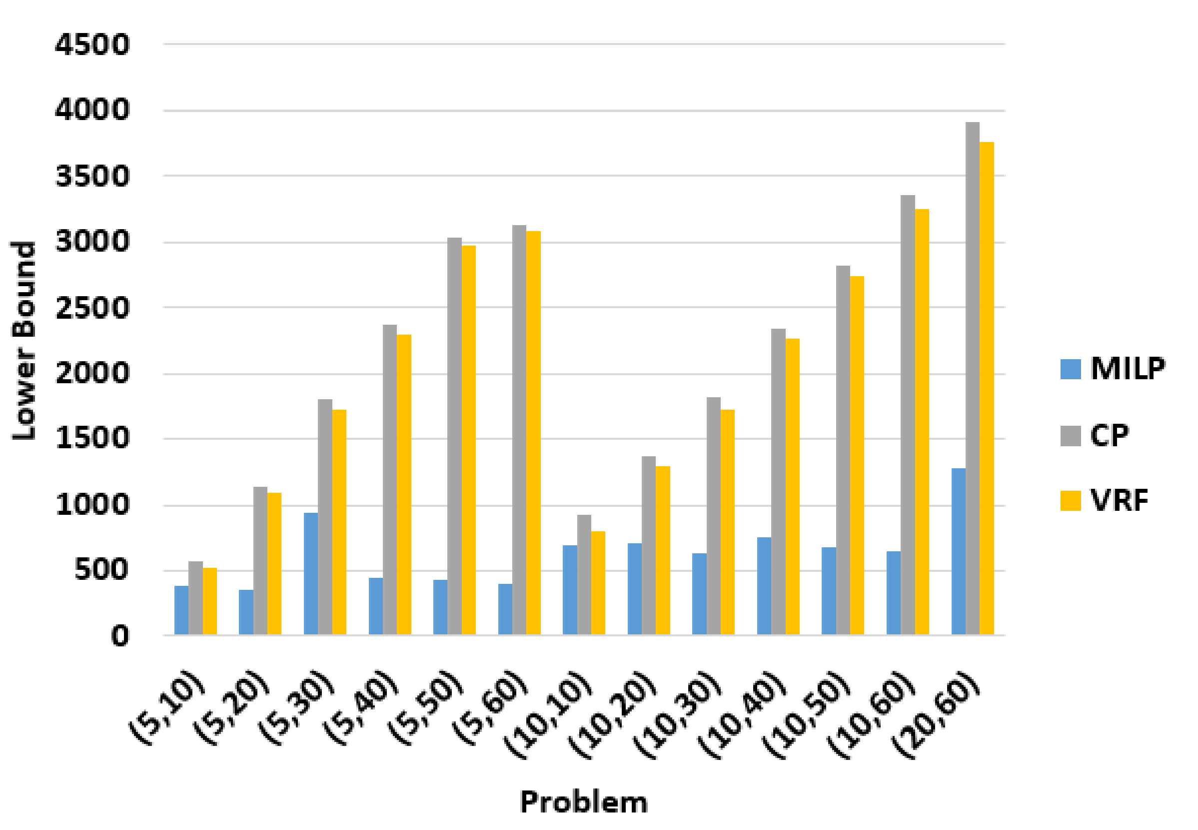

Figure 5.2: Comparison of lower bounds of the CP, MILP and VRF

5.1.2 Comparison of solutions for the permutation flowshop with time lags model

Another case for the analysis of the problem would be considering the problem without W. The practical implication would be when the total waiting time of each patient in the system is not considered. This model would be in the form of a permutation flowshop with time-lags. The MILP model for the resulting permutation flowshop with the time-lags model can be obtained after dropping the constraints (3.12) and (3.13) from the proposed MILP model. The resulting constraint programming permutation flowshop with time lags model can be obtained after dropping the constraints (3.27), (3.28) and (3.29) from the proposed constraint programming model.

MILP and CP models are compared using CPLEX. Four different combinations of machines and jobs are selected for the test. Problems with:

- Five Machines and fifteen jobs.
- Three machines and fifteen jobs.
- Five machines and twelve jobs.
- Ten machines and fifteen jobs.

For each problem, four different combinations of T_{ij}^1 and T_{ij}^2 are randomly generated using uniform distribution in the interval of [0, T_{ij}^1] and [0, T_{ij}^2]. The following combinations of T_{ij}^1 and T_{ij}^2 are used:

- (7, 14)
- (7, 200)
- (40, 60)
- (100, 200)

Processing times are generated randomly using uniform distribution of [20, 50]. The results are presented as follows:

Table 5.3: Comparison of MILP and CP for the model without W

Problem Parameters		CP		MILP	
(T1ij,T2ij)	Problem	Cmax	Time	Cmax	Time
(7, 14)	5 Machines 15 Jobs	701	149.459	709	300
	3 Machines 15 Jobs	611	10.839	614	300
	5 Machines 12 Jobs	594	19.5132	596	300
	10 Machines 15 Jobs	953	300	964	300
(7, 200)	5 Machines 15 Jobs	679	1.9478	685	300
	3 Machines 15 Jobs	628	0.1698	628	300
	5 Machines 12 Jobs	597	0.677	598	300
	10 Machines 15 Jobs	904	71.2868	921	300
(40, 60)	5 Machines 15 Jobs	780	81.8626	792	300
	3 Machines 15 Jobs	628	43.3788	631	300
	5 Machines 12 Jobs	671	15.2238	674	300
	10 Machines 15 Jobs	1082	300	1516	300
(100, 200)	5 Machines 15 Jobs	932	300	937	300
	3 Machines 15 Jobs	706	21.7835	712	300
	5 Machines 12 Jobs	847	47.003	847	300
	10 Machines 15 Jobs	1495	300	1513	300

Data are run in the CPLEX. The maximum run-time is 300 seconds. After 300 seconds the best objective value is reported. The results suggest that the CP significantly outperforms the MILP. The MILP almost did not solve any problem to optimality.

5.2 Sensitivity Analysis for the values of T_{ij}^1 and T_{ij}^2 for the exact solution

In this section, a sensitivity analysis is performed on how the changes in the values of T_{ij}^1 and T_{ij}^2 can impact the values of C_{max}, the average waiting times of the patients and the variance of the waiting times of the patients for the exact problem. One observation after multiple solving and running of the model in the CPLEX studio is that the solution time of the model is vulnerable to a small change in the values of W, and solution time may increase from less than two seconds to hours by decreasing the value of W by one or two units. Another observation is that although random numbers were generated to do sensitivity analysis, the solution times for the randomly generated numbers within one range of T_{ij}^1 or one range of T_{ij}^2 were not homogeneous; for example for the randomly generated numbers for T_{ij}^1 in the range of $[0, 3]$, while holding values of all other parameters constant, one observation was the vast range of the changes in the solution times of two different randomly generated instances of one range; to tackle this problem, a value for W was selected such that CPLEX can find the exact solution in a small amount of time for all problems. For each of the considered intervals of T_{ij}^1 and T_{ij}^2, five sets of random numbers are generated and solved in CPLEX 12.7.1.

In the sensitivity analysis section, the impact of changes in the values of T_{ij}^1 and T_{ij}^2 on the values of C_{max} and the waiting time of each patient is discussed. Five problem sets with different sizes are examined. The problems include the following:

- Seven machines and seven jobs.
- Three machines and twelve jobs.
- Five machines and twelve jobs.
- Five machines and nine jobs.
- Six machines and ten jobs.

5.2.1 Changes in the values of T_{ij}^1

For testing the change in the values of the T_{ij}^1, the interval of $[0, 14]$ is broken down into five equally sized intervals including the intervals of $[0, 2]$, $[3, 5]$, $[6, 8]$, $[9, 11]$ and $[12, 14]$, and for each interval, uniform random numbers are generated from each of the uniform distributions. For the values of processing times, uniform random numbers are generated from the uniform distribution of $[20, 50]$, for the values of the T_{ij}^2, uniform random numbers are generated from the uniform distribution of [14, 16], and W is considered based on the table 5.4. W is considered after doing a number of trials and errors for each problem. The W is developed such that the problem in different intervals can be solved in a reasonable amount of time. For each uniform distribution of T_{ij}^1, five sets of uniform random numbers for T_{ij}^1 are generated, and finally, the average values for each generated number are considered.

Table 5.4: Values of W for the problem of sensitivity analysis of the change of T_{ij}^1

Problem	W
7 Machines 7 Jobs	86.0
3 Machines 12 Job	40.0
5 Machines 12 Jobs	58.0
5 Machines 9 Jobs	54.0
6 Machines 10 Jobs	75.0

One system performance criterion that enhances with the increase in the values of T_{ij}^1 is the C_{max}. Figure 5.3 suggests how changes in the values of T_{ij}^1 influence the C_{max} for different intervals of T_{ij}^1 in different problems. As figure 5.3 suggests, for all problems, the C_{max} grows as T_{ij}^1 increases. It was already discussed that T_{ij}^1 is a necessary time lag; therefore, this increase in the C_{max} is a result of the increase in the waiting times of the jobs in each machine and consequently the increase in the finish time of each job J. The C_{max} does not increase with the same slope at all intervals. It is observed that for the last interval which is the closest interval to the values of T_{ij}^2, the slope increases. This rise in the slope can be a result of the decrease in the feasible space. For the problem of five machines and nine jobs, it is observed that there is not any change. One other observation is that for the problem with three machines and twelve jobs, C_{max} increases with smaller slope; this slow increase can be a result of the smaller number of machines.

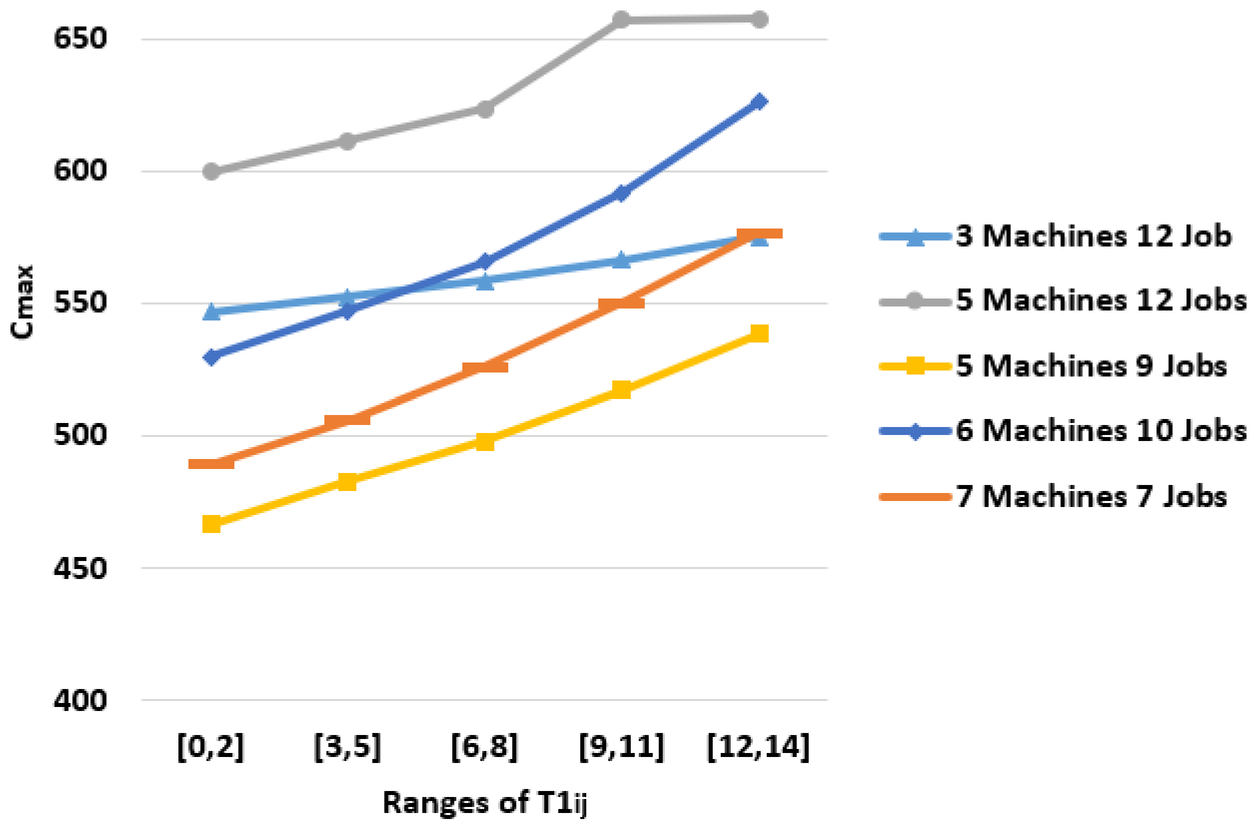

Figure 5.3: Sensitivity of C_{max} to the changes of T^1_{ij}

Table 5.5 presents the sensitivity of C_{max} with the change in the values of T^1_{ij} for different problems. As shown in figure 5.3, as the number of machines increase, the C_{max} increases. The last column of the table shows the difference between the average of C_{max} for the random generated T^1_{ij} with the smallest average, that is $[0, 2]$, and the random generated T^1_{ij} with the largest average, that is $[12, 14]$.

Table 5.5: Sensitivity of change of C_{max} to the change of T_{ij}^1

Problem	[0,2]	[3,5]	[6,8]	[9,11]	[12,14]	Increase in Cmax (%)
3 Machines 12 Job	546.8	552.6	558.4	566.4	575.0	5.16%
5 Machines 12 Jobs	599.6	611.4	623.6	657.0	657.4	9.64%
5 Machines 9 Jobs	466.8	483.0	497.8	517.2	538.6	15.38%
6 Machines 10 Jobs	529.8	547.2	565.8	591.8	626.0	18.16%
7 Machines 7 Jobs	489.4	505.8	525.8	550.2	576.6	17.82%

Figure 5.4, suggests the effect of changes in the values of T_{ij}^1 on the average waiting time of each patient for different problems. The figure indicates that as T_{ij}^1 or the minimum time-lag increases, as it is expected, the average waiting time of each patient increases. This implication is predictable since when T_{ij}^1 increases, this value is a compulsory waiting time per se, and it directly increases the average and the total waiting time of each patient. Although the average waiting time per patient increases for different problems, there is a difference between the slopes of different problems. Figure 5.4 suggests that as the number of machines rises the slope of the average waiting time of each patient increases. The lowest value of the increase is for the problem with three machines and twelve jobs. The problem with seven machines and seven jobs has the highest slope. In the healthcare systems this can be helpful in that when a patient needs to go through a longer sequence of medical operations such as more physicians, more medical tests, etc., by increase in each of T_{ij}^1 there would be further increase in the average waiting time of patients. Another observation is that we do not observe a specific trend of changes in the values of C_{max} based on the changed in the number of jobs or machines

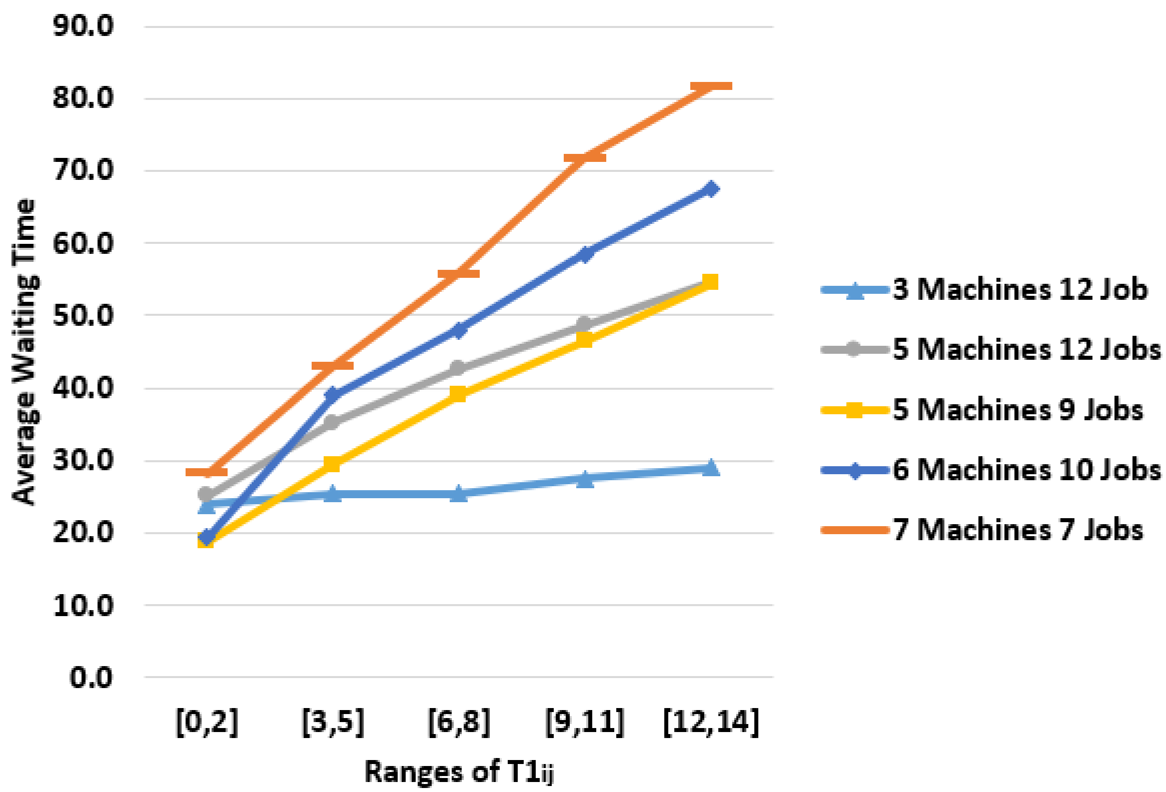

Figure 5.4: Sensitivity of average waiting times per patient on the changes of the values of T^1_{ij}

Table 5.6 provides the data for changes in the values of T^1_{ij} for different problems. As shown in figure 5.4, as the number of machines increases, the average waiting time per patient increases. The last column of the table implies that the highest increase in the average waiting time corresponds to the problem with seven machines and seven jobs and the problem with three machines and twelve jobs has the lowest value of the average waiting time.

Table 5.6: Sensitivity of change of average waiting time to the change of T^1_{ij}

Problem	[7,9]	[10,12]	[13,15]	[16,18]	[19,21]	Increase in Average Waiting Time (%)
3 Machines 12 Job	23.9	25.5	25.4	27.9	29.1	21.38%
5 Machines 12 Jobs	25.2	35.3	42.6	48.7	54.7	116.93%
5 Machines 9 Jobs	18.8	29.5	39.0	46.5	54.5	189.83%
6 Machines 10 Jobs	19.4	39.0	47.9	58.5	67.6	249.21%
7 Machines 7 Jobs	28.3	43.0	55.7	71.7	81.6	188.78%

It was discussed in chapter one that variations in waiting time can be explained by patients, clinic and provider level (Dansky and Miles, 1997; Dimakou et al., 2009). Therefore, it would be interesting to study the variances of the waiting times sensitivity in different problems. One observation after changing the values of T_{ij}^1 in the output of waiting times is the trend in the variance of waiting time of each interval. As figure 5.5 suggests, there is a decline in the variance of waiting times of each interval. With the increase in the values of T_{ij}^1, the values of the variance of waiting times decrease. Therefore, in healthcare systems and for small values of T_{ij}^1, it can be expected that there is more variation in the waiting times of the patients, and there are patients who wait for long hours while some patients receive their service very quickly. This phenomenon might not be favorable for healthcare decision makers and can increase dissatisfaction of those patients who have to go through long waiting times.

Another observation is that for smaller values of T_{ij}^1, the largest value of the variance of waiting time is for the problem with seven machines and seven jobs and the smallest value of variance is for the problem of three machines and twelve jobs. This suggests an increase in the machines can increase the variance of waiting time when there is enough feasible space. Therefore, if the number of resources that a patient wants to visit increases, decision makers can expect more variations in waiting times if there is a gap between the values of minimal and maximal time lags.

It can be observed that with the rise of T_{ij}^1, in all problems the variance tends to zero. This observation implies that regardless of the size of the problem, with the decrease in the gap between T_{ij}^1 and T_{ij}^2 and decrease in feasible space, there are fewer variations in waiting times of the patients.

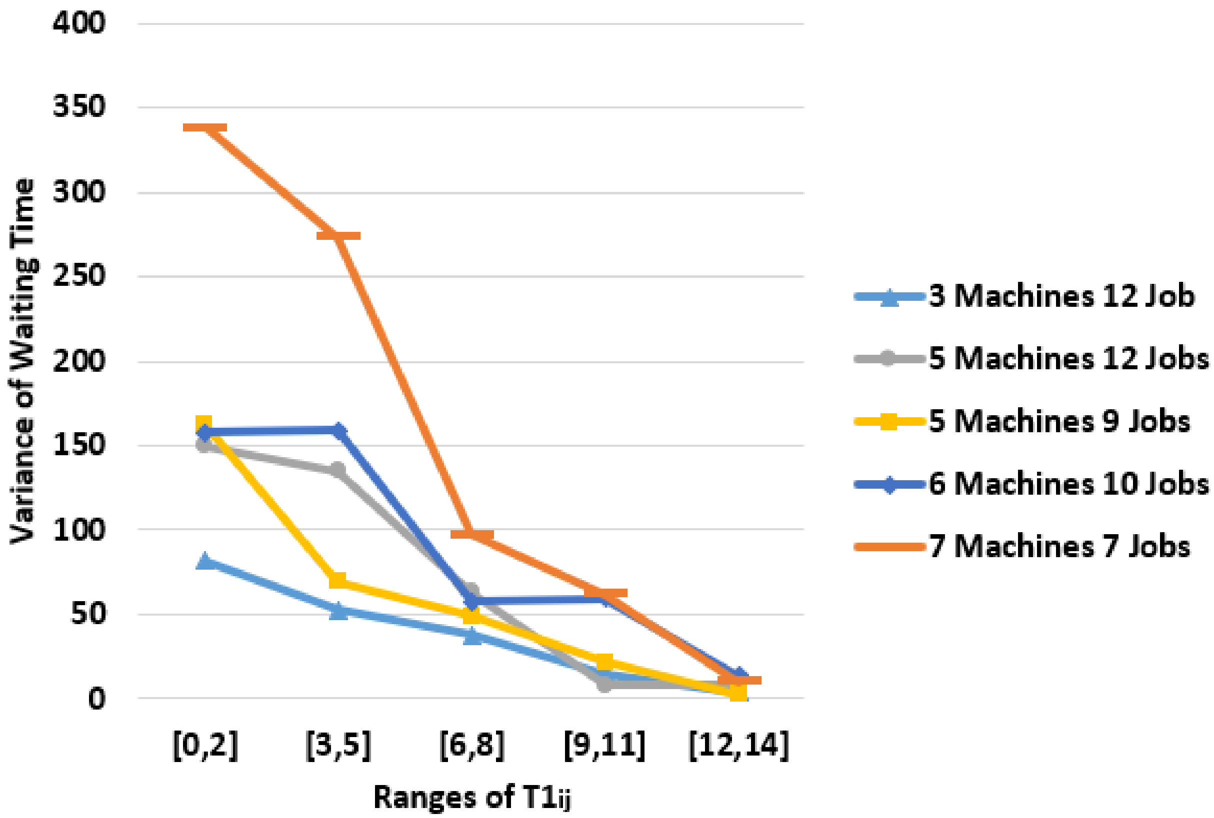

Figure 5.5: Sensitivity of variance of waiting times to the changes of the values of T^1_{ij}

Results of the sensitivity analysis of figure 5.4 and figure 5.3 suggest that any increase in the values of T^1_{ij} can result in the increase in both C_{max} and the average waiting time of each patient; therefore, if decision makers try to decrease this time they can improve both performance metrics of average waiting time of each patient and C_{max} while they can expect that there are more variations in the variance of waiting times of patients. One other implication of high values of variance is that high values of variance can make it difficult to forecast future waiting times. This fact suggests that in higher values of T^1_{ij} it is easier to forecast possible waiting times of the patients. One other important statistical property of variance of waiting times arises if we want to consider a distribution, such as normal distribution, for waiting times. Kurtosis is defined as $\frac{\mu_4}{\sigma_4}$, and smaller values of kurtosis makes forecasting of the values of a distribution less reliable (Joanes and Gill, 1998). If μ is fixed then higher values of variance (σ^2) decreases the kurtosis and makes it more difficult to forecast the values of the waiting times.

Table 5.7 presents numerical values of C_{max} versus total waiting time of the patients for different problems in different intervals of T^1_{ij}.

Table 5.7: Changes of Total waiting time V.S changes of C_{max} for different intervals of the T^1_{ij}

Problem	Problem	[7,9]	[10,12]	[13,15]	[16,18]	[19,21]
3 Machines 12 Job	Total Waiting Time	287.2	305.8	305.2	329.8	348.6
	Cmax	546.8	552.6	558.4	566.4	575.0
5 Machines 12 Jobs	Total Waiting Time	302.8	423.2	510.8	583.8	656.0
	Cmax	599.6	611.4	623.6	657.0	657.4
5 Machines 9 Jobs	Total Waiting Time	169.2	265.8	351.2	418.6	490.4
	Cmax	466.8	483.0	497.8	517.2	538.6
6 Machines 10 Jobs	Total Waiting Time	193.7	390.0	479.4	584.6	676.4
	Cmax	529.8	547.2	565.8	591.8	626.0
7 Machines 7 Jobs	Total Waiting Time	197.8	301.0	389.8	502.0	571.2
	Cmax	489.4	505.8	525.8	550.2	576.6

The table suggests that for all problems the trend of the change is very similar although for the problem with seven machines and seven jobs the total waiting time does not change similarly to the others. This can be due to the low number of jobs and a large number of machines compared to other problems.

Table 5.7 suggests that the problem with six machines and ten jobs has the highest total waiting time and the largest value of the change in total waiting time. Therefore, this problem is the most sensitive to the changes of T^1_{ij}.

5.2.2 Changes in the values of T^2_{ij}

A procedure similar to that of sensitivity analysis of T^1_{ij} is followed to study the effect of the changes in the values of T^2_{ij} on C_{max}, the average waiting time of each patient and the variance of waiting time of each patient. For processing times, uniform random numbers are generated from the uniform distribution of [20, 50]; for generating values for T^1_{ij}, uniform random numbers are generated from the uniform distribution of [0, 6], and to generate values for T^2_{ij}, the interval of [7, 21] was broken down into five intervals of [7, 9], [10, 12], [13, 15],[16, 18], [19, 21] and [22, 24]. Values for W are based on the table 5.4. Uniform random numbers are generated for each of the intervals of T^2_{ij}, and then the average of each of the five observations are calculated and compared.

Figure 5.6 shows the impact of the changes in the values of T^2_{ij} on C_{max} in different problems. It can be observed that an increase in the values of T^2_{ij} can lead to decrease in C_{max}. This is completely in contrast to the case of the changes in the values of T^1_{ij}. One implication is that an increase in the values of T^2_{ij} can increase the feasible space. This increase can improve and decrease the C_{max}. The largest improvement in the C_{max} can be obtained when T_{ij}^2 increases from the range of [7, 9] to the range of [10, 12]. The range of [7, 9] is the closest range to [0, 6] which is the range that is used for generating the random numbers for the values of T_{ij}^1. Therefore, with an increase in the values of T_{ij}^2, feasible space, and the solution increases.

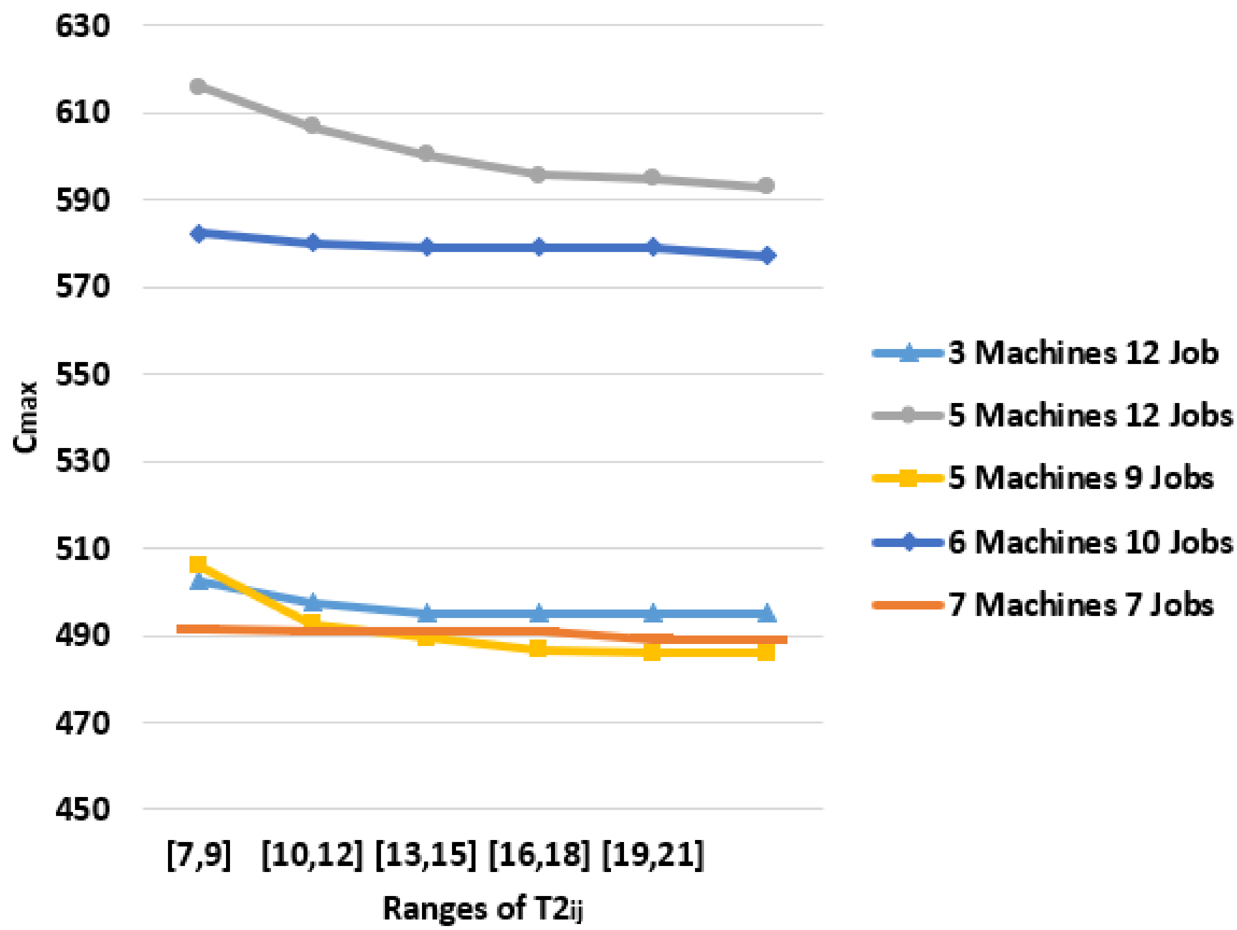

Figure 5.6: Sensitivity of C_{max} to the changes of T_{ij}^2

Values of the change are presented in table 5.8. The table suggests that as T_{ij}^2 increases, more decrease in C_{max} is achieved.

Table 5.8: Sensitivity of change of C_{max} to the change of T_{ij}^2

Problem	[7,9]	[10,12]	[13,15]	[16,18]	[19,21]	[22,24]	Decrease in Cmax (%)
3 Machines 12 Job	502.4	497.6	495.0	495.0	495.0	495.0	1.47%
5 Machines 12 Jobs	616.0	606.8	600.2	595.6	595.0	593.0	3.73%
5 Machines 9 Jobs	506.0	492.6	489.4	486.8	486.0	486.0	3.95%
6 Machines 10 Jobs	582.4	580.0	579.0	579.0	579.0	577.0	0.93%
7 Machines 7 Jobs	491.4	491.0	491.0	491.0	489.2	489.0	0.49%

Figure 5.7 presents how the changes in the values of T_{ij}^2 impact the average waiting time of each patient. The figure suggests an increase in the values of T_{ij}^2 increases the average waiting time of each patient. The increase in the waiting times holds true for all problems. One implication can be that any increase in the values of T_{ij}^2 (or maximum waiting times) would increase the potential waiting times. This increase in maximum waiting times before each machine can help the algorithm to find the best values of decision variables for minimizing makespan. It is observed that generally if patients increase the maximum waiting times between successive operations, the average time they wait would increase. Another observation is that the slope of the average waiting time of each patient changes in different intervals and in the values close to the T_{ij}^1, an increase of the values of T_{ij}^2 results in greater increase in the average waiting time.

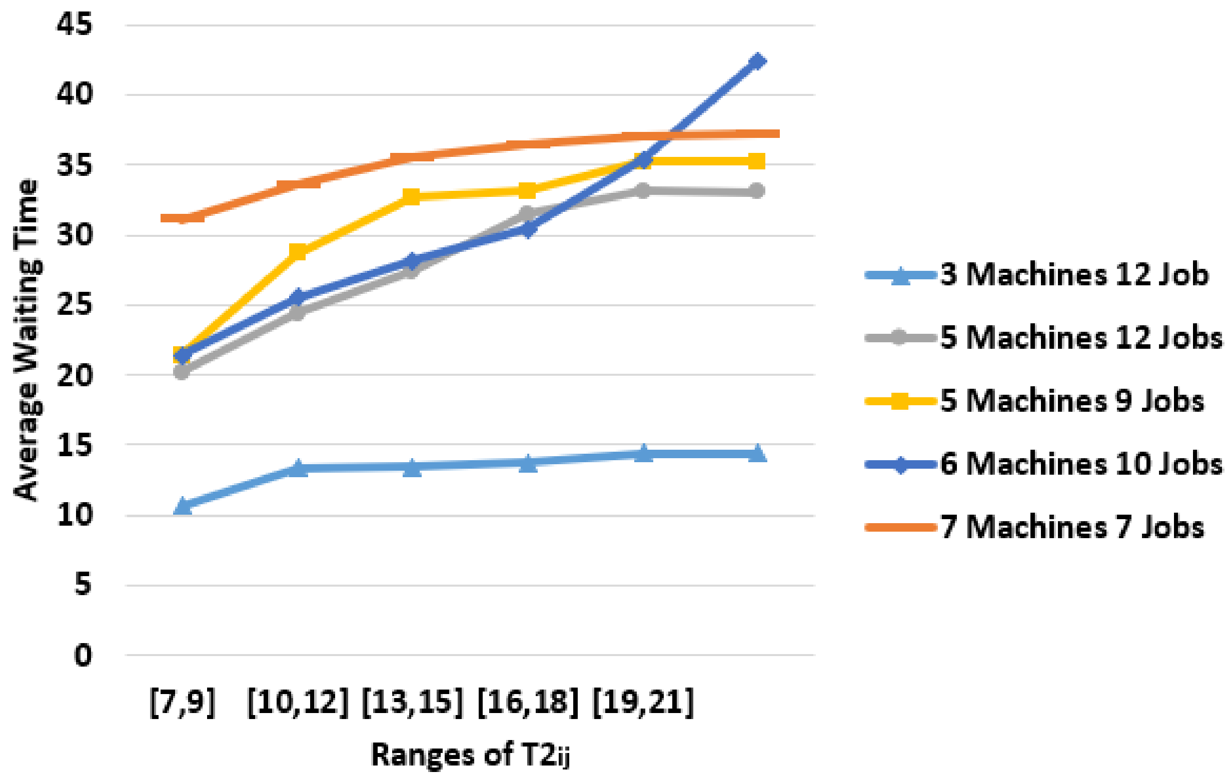

Figure 5.7: Sensitivity of average waiting time of each patient to the changes in the values T_{ij}^2

Numerical values of the average waiting times have been calculated and presented in table 5.9. The percentage of increase in average waiting time after increasing T_{ij}^2 is presented in the last column. The values suggest that the lowest value of the decrease is for the problem with seven machines and seven jobs and the highest value in the increase in average waiting time is for the problem with six machines and six jobs.

Table 5.9: Sensitivity of change of the Average Waiting Time to the change of T_{ij}^2

Problem	[7,9]	[10,12]	[13,15]	[16,18]	[19,21]	[22,24]	Increase in Average Waiting Time (%)
3 Machines 12 Job	10.7	13.4	13.4	13.8	14.4	14.4	34.95%
5 Machines 12 Jobs	20.2	24.4	27.4	31.5	33.1	33.0	63.45%
5 Machines 9 Jobs	21.4	28.7	32.7	33.1	35.2	35.2	64.42%
6 Machines 10 Jobs	21.4	25.5	28.2	30.4	35.4	42.4	98.13%
7 Machines 7 Jobs	31.1	33.6	35.5	36.5	37.0	37.1	19.38%

Figure 5.8 shows how the variance of waiting times for patients changes as T_{ij}^2 increases. As the figure suggests, generally the variance increases with the increase in the values of T_{ij}^2. One reason for the increase can be the rise in the values of T_{ij}^2 that increases the gap between T_{ij}^1 and T_{ij}^2. Increase of this gap can increase feasible space, and therefore it can cause more variations in the waiting times among different patients. There are some changes in the behaviour of variance; the most significant change is for the problem with seven jobs and seven machines. This change in the trend can be a result of some irregular waitings for some of the jobs when T_{ij}^2 is generated from uniform distribution of [16, 18].

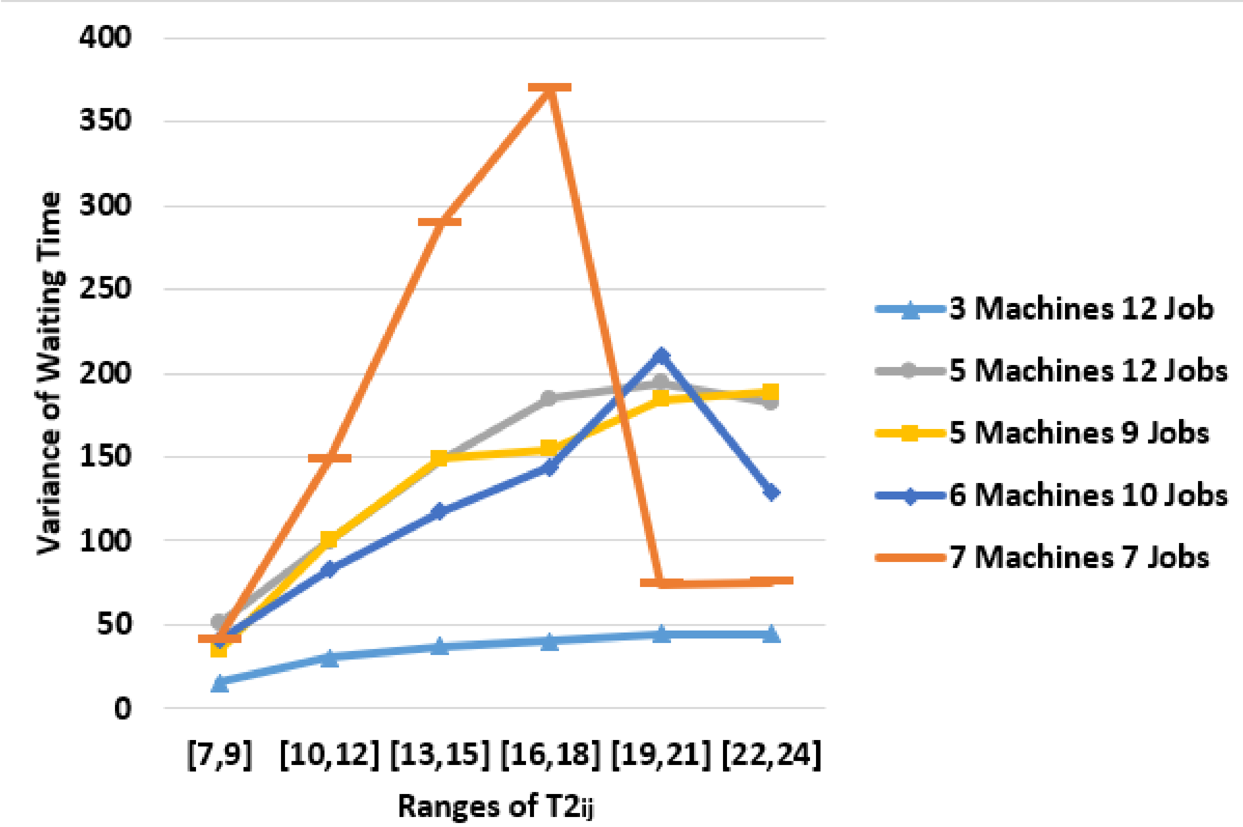

Figure 5.8: Sensitivity of variance of waiting times to the changes in the values of T_{ij}^2

Figures 5.6 and 5.7 suggest that an increase in the values of T_{ij}^2 would not improve both performance metrics of C_{max} and the average waiting time of each patient at the same time. It is shown that while an increase in the values of T_{ij}^2 can improve the C_{max}, it does not improve the average waiting time and the variance of waiting times.

Table 5.10 presents numerical values C_{max} versus total waiting time of the patients plot for different problems in different intervals of T_{ij}^2.

Table 5.10: Changes of total waiting time V.S changes of C_{max} for different intervals of the T_{ij}^2

Problem	Problem	[7,9]	[10,12]	[13,15]	[16,18]	[19,21]	[22,24]
3 Machines 12 Job	Total Waiting Time	128.2	160.6	161.2	165.0	173.0	173.0
	Cmax	502.4	497.6	495.0	495.0	495.0	495.0
5 Machines 12 Jobs	Total Waiting Time	242.4	292.8	329.2	378.0	397.2	396.2
	Cmax	616.0	606.8	600.2	595.6	595.0	593.0
5 Machines 9 Jobs	Total Waiting Time	192.8	258.2	294.2	298.2	317.0	317.0
	Cmax	506.0	492.6	489.4	486.8	486.0	486.0
6 Machines 10 Jobs	Total Waiting Time	214.0	255.4	281.8	303.8	353.8	424.0
	Cmax	582.4	580.0	579.0	579.0	579.0	577.0
7 Machines 7 Jobs	Total Waiting Time	217.8	235.0	248.6	255.2	259.2	260.0
	Cmax	491.4	491.0	491.0	491.0	489.2	489.0

In table 5.10, total waiting time is used to present the total waiting time of the system. For all problems, the table suggests that the trend of the change is very similar although for the problem with seven machines and seven jobs the total waiting time does not change similar to others. This can be due to the low number of jobs and a large number of machines compared to other problems.

5.3 Metaheuristic solutions

In this section, the solution to the metaheuristic model is presented. At the first step, parameters need to be tuned, then a genetic algorithm, intensification using MILP (GA_MILP) and intensification using CP (GA_CP) are compared to find the algorithm with the best performance for each problem size. Taguchi method is used to set the parameters (Montgomery, 2017). Population size, crossover probability, and mutation probability are tuned using the Taguchi method. The parameters used and their corresponding levels are presented in table 5.11.

Table 5.11: Parameter levels used in parameter setting

	Parameters Levels		
Parameters	1	2	3
Mutation Probability	0.25	0.5	0.75
Crossover Probability	0.25	0.5	0.75
PopulationSize	50	100	75

For each problem, three randomly generated datasets are used. Each dataset is run five times and the average value of the total 15 runs is used in experimental design. *L*9 orthogonal array is used for the purpose of experimental design. Table 5.12 shows the layout of the L9 Orthogonal array for the problem of 10 machines and 60 jobs. The response column presents average values of C_{max}.

Table 5.12: Layout of L9 orthogonal array for the problem of 10 machines and 60 jobs

Mutation Probability	Crossover Probability	PopulationSize	Response
1	1	1	3426.2
1	2	2	3421.2
1	3	3	3466.4
2	1	2	3420.6
2	2	3	3414.733
2	3	1	3433.067
3	1	3	3422.533
3	2	1	3419.667
3	3	2	3417.2

S/N ratio with the condition of smaller is better is used to test the model. S/N ratio measures the change of response variable relative to the noise variables and it is calculated using the equation (Haleh, Maghsoudlou, Hadipour, and Nabovati, 2017):

$$S/Nration = -10log(\frac{1}{N}\sum_{j=1}^{N} y_i^2) \tag{5.1}$$

Where i is the number of experiment, y_i is the response value and N is the total number of experiments. After analyzing the outcomes, the findings of the study are as follows:

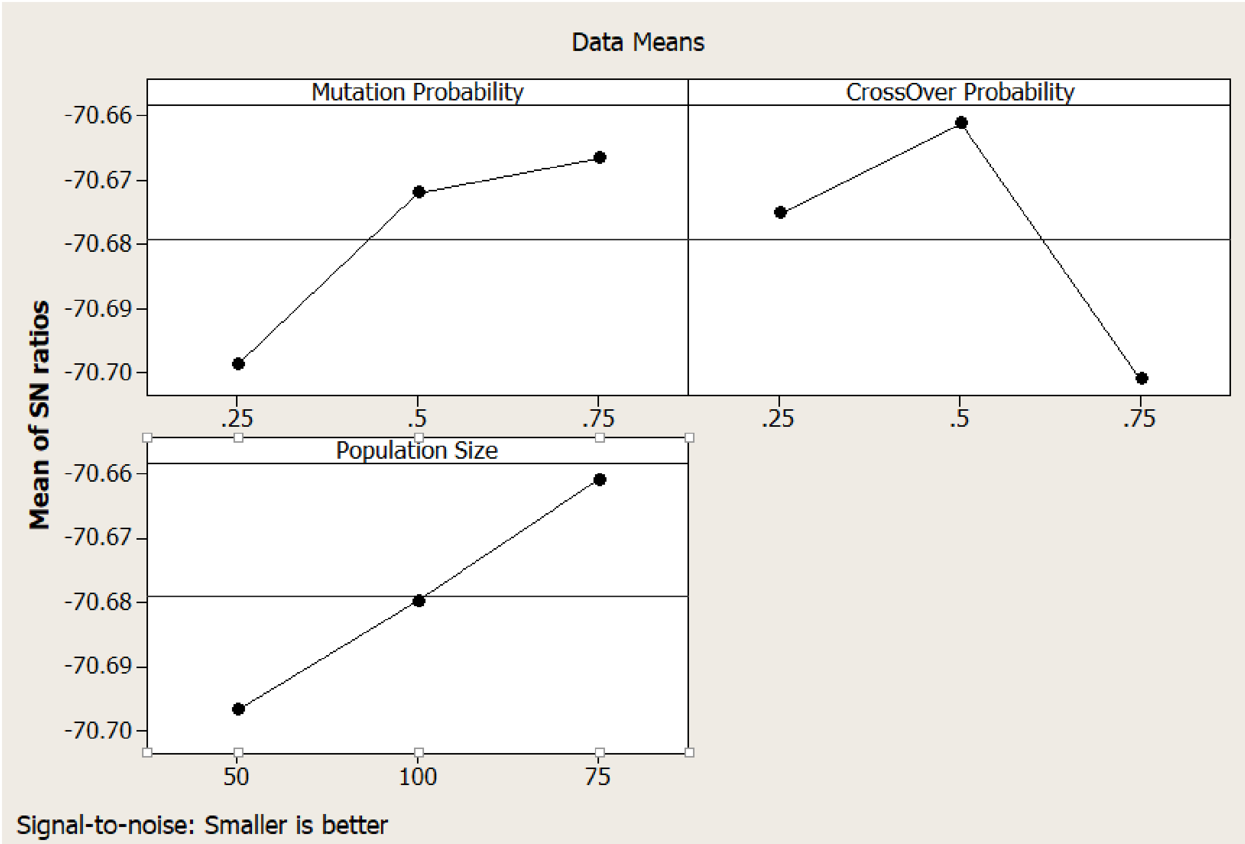

Figure 5.9: S/N ratios of the parameters for the problem of 10 Machines and 60 Jobs

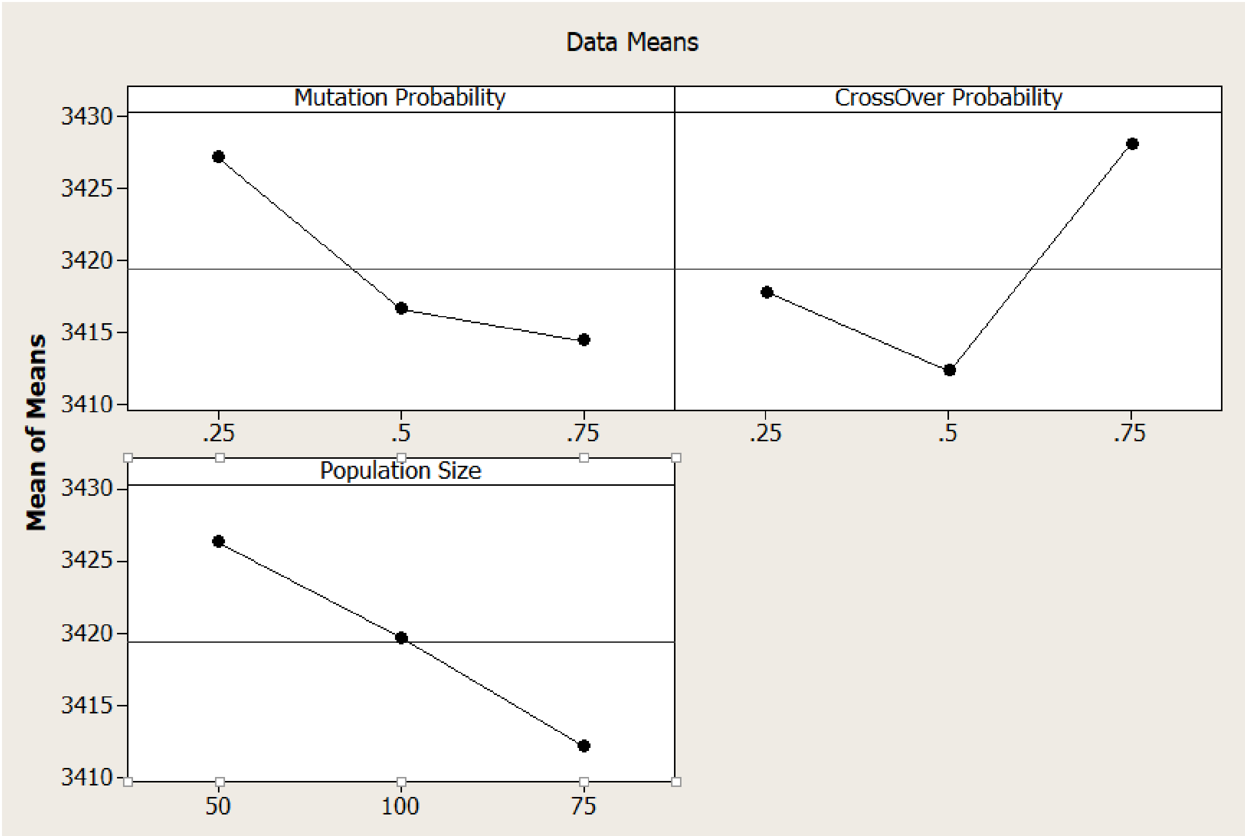

Figure 5.10: Mean of the response for the parameters for the problem of 10 Machines and 60 Jobs

Higher values of S/N ratio imply better response. These values suggest that the best value of crossover probability is 0.5, the best value of mutation probability is 0.75 and the best population size is 100. For each problem, a similar procedure is followed. Table 5.13 presents a summary of the outputs.

Table 5.13: Summary of the best parameters after DOE

Problem	Mutation Probability	Crossover Probability	PopulationSize
5 Machines and 10 Jobs	0.75	0.75	100
5 Machines and 40 Jobs	0.5	0.75	75
5 Machines and 60 Jobs	0.5	0.75	100
10 Machines and 20 Jobs	0.25	0.25	50
10 Machines and 40 Jobs	0.75	0.75	100
10 Machines and 60 Jobs	0.75	0.5	100
15 Machines and 40 Jobs	0.5	0.75	75
20 Machines and 50 Jobs	0.5	0.5	75
20 Machines and 200 Jobs	0.5	0.75	50

After setting the parameters, a comparison of result is performed to investigate the performance of algorithm. Table 5.14 presents a summary of the outputs. GA is run for 500 iterations. The small problem of the intensification for the GA_MILP and GA_CP is considered with three jobs. Maximum run-time is 100 seconds. For each problem, two runs, in five different datasets are run and the average values of five runs is recorded.

Table 5.14: Comparison of metaheuristics

(Machines, Jobs)	GA		GA_MILP		GA_CP	
	Objective Function	Time (in seconds)	Objective Function	Time (in seconds)	Objective Function	Time (in seconds)
(5, 10)	521.8	4.3	521.4	4.7	517.0	4.1
(10, 20)	1183.4	3.3	939.1	9.2	1154.4	6.7
(10, 40)	2126.8	16.1	2117.0	100.0	2069.0	17.8
(10, 60)	3117.6	18.3	3075.0	100.0	3018.0	18.7
(10, 100)	5016.4	11.0	4974.4	100.0	4869.4	15.1
(20, 50)	3778.4	15.7	3771.2	100.0	3656.6	18.1

The results suggest that GA_CP model outperforms GA_MILP. As problem size increases the competitiveness of GA_CP becomes more significant due to its speed solution advantage.

5.3.1 Comparison of the solutions for the permutation flowshop problem

In this section, the solutions of the GA_MILP and GA_CP permutation flowshop models are compared using four instances of the test problems provided by Vallada et al. (2015). The test upper bounds of problems are presented. The algorithms used here are the ones used after relaxing the relevant time-lag constraints.

Each dataset has been run three times and the number suggests the average of three runs. Each dataset is run using genetic algorithm. All problems of proposed algorithms had run-time less than the one considered by Vallada et al. (2015). For the small problem of five machines and ten jobs and the problem of five machines and forty jobs, the genetic algorithm improves the upper bounds; however, for other larger sizes the algorithms used by Vallada et al. (2015) have better performance.

Table 5.15: Comparison of results for the metaheuristic models without time-lags

Problem (Machines, Jobs)	GA	GA_MILP	GA_CP	VRF	Run time (in seconds)
	Average solution	Average solution	Average solution	Upper Bound	
(5,10)	623.33	623.33	623.33	695	3
(5,40)	2450.00	2446.00	2366.33	2396	12
(5,60)	3482.33	3479.33	3452.33	3350	18
(10,20)	1644.67	1624.67	1647.67	1532	12
(10,40)	2834.50	2818.33	2823.33	2480	24
(10,60)	3931.00	3849.67	3901.67	3435	36
(15,40)	3388.00	3314.33	3312.33	3011	36
(20,50)	4245.67	4299.33	4217.67	3693	60
(20,200)	12994.33	13034.67	12795.00	11305	240

In this section, a sensitivity analysis is performed on how the changes in the values of T_{ij}^1 and T_{ij}^2 can impact the values of C_{max} and the average waiting times of the patients.

Four problem sets with different sizes are examined. The problems include the following:

- Five machines and ten jobs.
- Ten machines and forty jobs.
- Ten machines and sixty jobs.
- Ten machines and 100 jobs.

- Six machines and ten jobs.

The sensitivity analysis is performed using GA_CP algorithm since it had better performance that GA_MILP algorithm.

5.3.2 Changes in the values of T_{ij}^1

For testing the change in the values of the T_{ij}^1, the intervals similar to the intervals and random numbers that were considered in the exact part are studied. Only W is considered equal to 100. For each uniform distribution of T_{ij}^1, five sets of uniform random numbers for T_{ij}^1 are generated, and finally, the average values for five generated numbers are considered.

Figure 5.3 suggests how changes in the values of T_{ij}^1 influence on the C_{max} for different intervals of T_{ij}^1 in different problems. As figure 5.3 suggests, for all problems, the C_{max} increases as T_{ij}^1 increases. As discussed earlier, this increase in the C_{max} is a result of the increase in the waiting times of the jobs in each machine and consequently the increase in the finish time of each job J. The C_{max} does not increase with the same slope at all intervals. It is observed that for the last interval which is the closest interval to the values of T_{ij}^2, the slope increases. This rise in the slope can be a result of the decrease in the feasible space. For the problem of five machines and nine jobs, it is observed that there is not any change. One other observation is that for the problem with three machines and twelve jobs, C_{max} increases with smaller slope; this slow increase can be a result of the smaller number of machines.

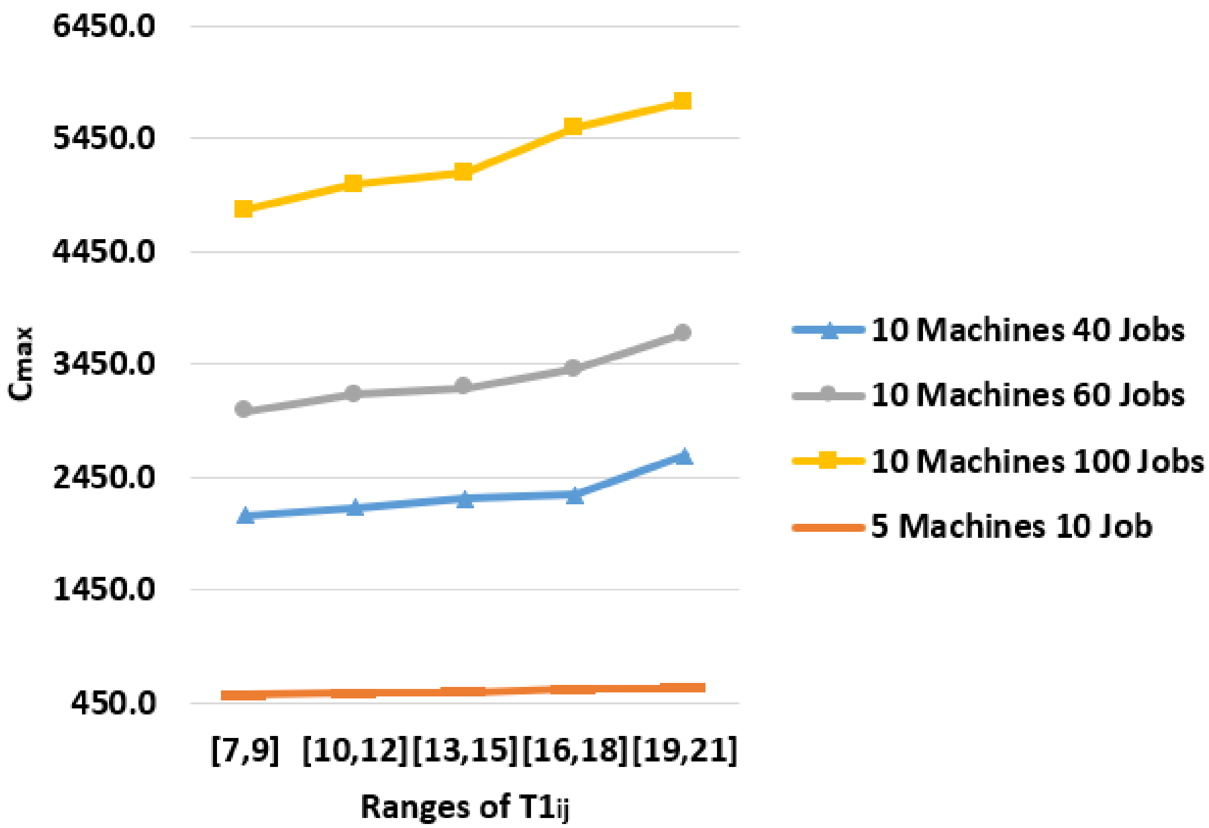

Figure 5.11: Sensitivity of C_{max} to the changes of T^1_{ij} in metaheuristic solutions

Table 5.16 presents the sensitivity of C_{max} on the change of the values of T^1_{ij} for different problems. It is observed that in figure 5.3, as the number of machines increase, the C_{max} increases that is similar to the exact solution. The last column of the table shows the difference of the average of C_{max} for the random generated T^1_{ij} in the range of $[0, 2]$, with the random generated of T^1_{ij} in the range of $[12, 14]$. These findings can have implications in healthcare systems how a change in the necessary waiting times between operations can increase the C_{max} of the patients. Increase in necessary waiting time, from $[0, 2]$ to $[12, 14]$ can increase the C_{max} in the ranges of 12 to 25 percent.

Table 5.16: Sensitivity of change of C_{max} to the change of T_{ij}^1 in metaheuristic solutions

Problem	[0,2]	[3,5]	[6,8]	[9,11]	[12,14]	Increase in Cmax (%)
5 Machines 10 Job	518.8	535.6	544.2	568.6	585.8	12.91%
10 Machines 40 Jobs	2115.0	2178.0	2259.6	2294.0	2643.8	25.00%
10 Machines 60 Jobs	3032.4	3184.6	3241.0	3406.2	3718.5	22.63%
10 Machines 100 Jobs	4812.2	5040.2	5149.4	5550.6	5768.6	19.87%

Figure 5.12 suggests the effect of changes in the values of T_{ij}^1 on the average waiting time of each patient for different problems. The figure indicates that as T_{ij}^1 or the minimum time-lag increases, as it is expected, the average waiting time of each patient increases. This implication is predictable since when T_{ij}^1 increases, this value is a compulsory waiting time per se, and it directly increases the average and the total waiting time of each patient. Although the average waiting time per patient increases for different problems, there is a difference between the slopes of different problems. Figure 5.4 suggests that as the number of machines increases the slope of the average waiting time of each patient rises. In the healthcare systems this can be helpful in that when a patient needs to go through a longer sequence of medical operations such as more physicians, more medical tests, etc., by increase in each of T_{ij}^1 there would be greater increase in the average waiting time of patients.

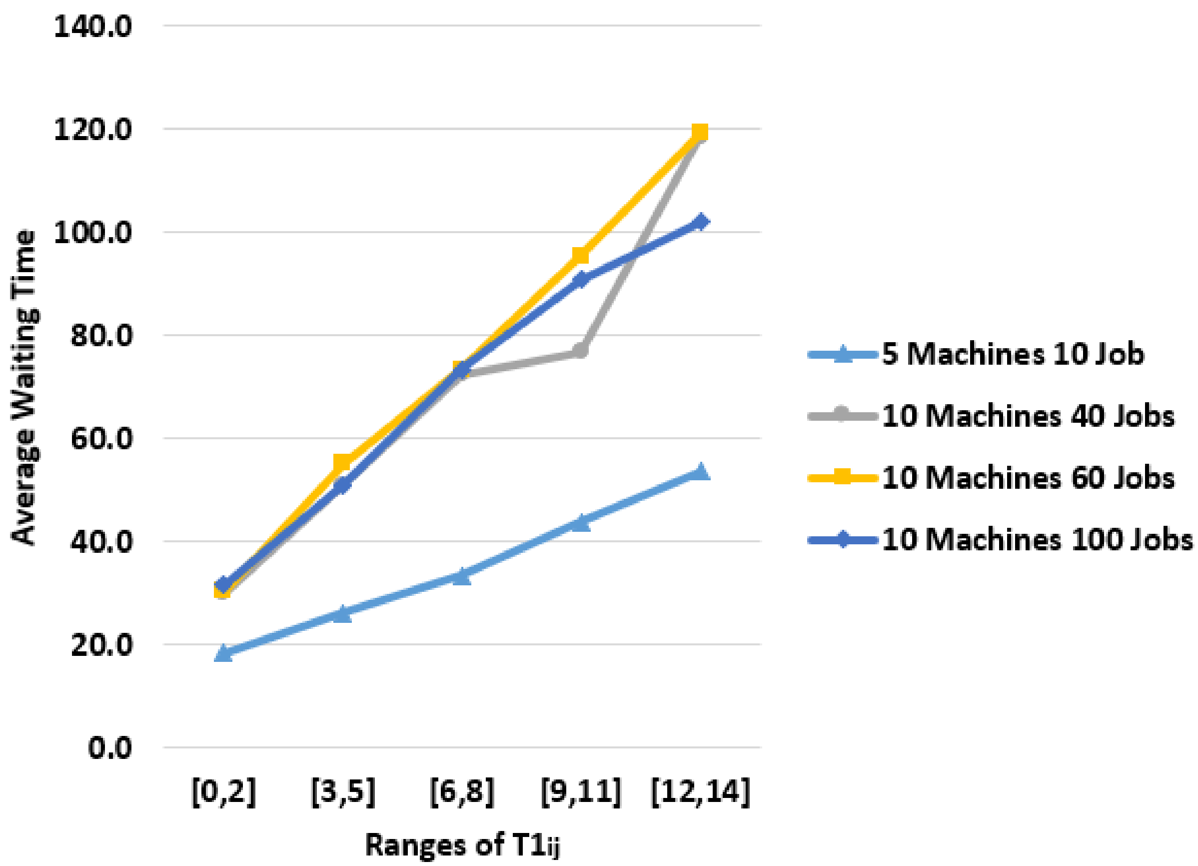

Figure 5.12: Sensitivity of average waiting times per patient on the changes of the values of T_{ij}^1 for metaheuristic solutions

Table 5.17 provides the data for changes in the values of T_{ij}^1 for different problems. As it was already indicated in table 5.12 as the number of machines increases, the average waiting time per patient increases. The last column of the table implies that the highest percentage increase in the average waiting time belongs to the problem with ten machines and forty jobs and the problem with five machines and ten jobs has the lowest value of the average waiting time. We can observe that an increase of T_{ij}^1 from $[0, 2]$ to $[12, 14]$ can result in increase of at minimum 190.45% in average waiting times. This is a large increase in average waiting times.

Table 5.17: Sensitivity of change of average waiting time to the change of T_{ij}^1 for metaheuristic solutions

Problem	[0,2]	[3,5]	[6,8]	[9,11]	[12,14]	Increase in Average Waiting Time (%)
5 Machines 10 Job	18.4	26.2	33.5	43.8	53.5	190.45%
10 Machines 40 Jobs	29.9	50.9	72.4	76.8	118.6	296.04%
10 Machines 60 Jobs	30.5	55.1	73.4	95.4	119.2	290.96%
10 Machines 100 Jobs	31.6	51.1	73.4	90.8	101.9	222.78%

5.3.3 Changes in the values of T_{ij}^2

A procedure similar to that of sensitivity analysis of T_{ij}^2 in exact solutions is followed to study the effect of the changes in the values of T_{ij}^2 on C_{max} and the average waiting time of each patient for metaheuristic solutions.

Figure 5.6 shows the impact of the changes in the values of T_{ij}^2 on C_{max} in different problems. It can be observed that an increase in the values of T_{ij}^2 can lead to decrease in C_{max}. This is completely in contrast to the case of the changes in the values of T_{ij}^1. One implication is that an increase in the values of T_{ij}^2 can increase the feasible space. This increase can improve and decrease the C_{max}.

The largest improvement in the C_{max} can be obtained when T_{ij}^2 increases from the range of $[7, 9]$ to the range of $[10, 12]$. The range of $[7, 9]$ is the closest range to $[0, 6]$ which is the range that is used for generating the random numbers for the values of T_{ij}^1. Therefore, with an increase in the values of T_{ij}^2, the feasible space and the solution increases.

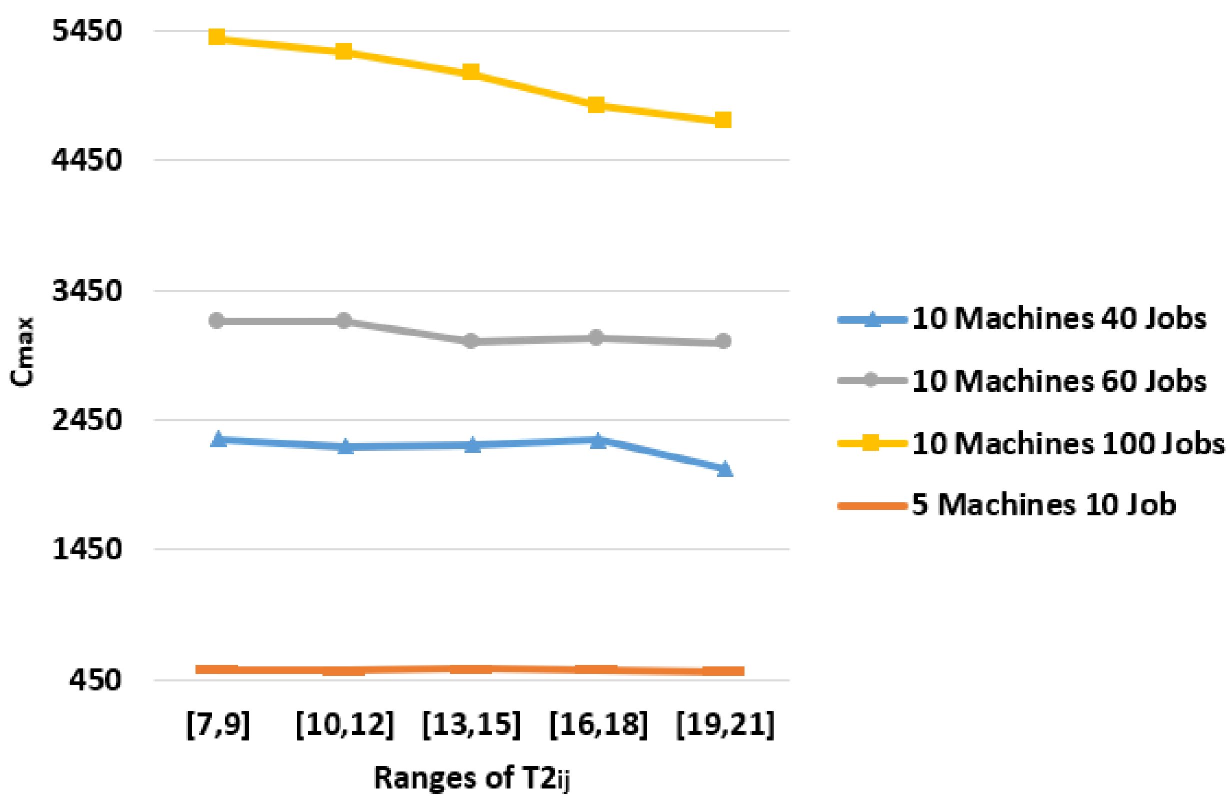

Figure 5.13: Sensitivity of C_{max} to the changes of T_{ij}^2

The values of the change are presented in table 5.18. The table suggests that as T_{ij}^2 increases, more decrease in C_{max} is achieved. Improvements are from 3.42 to almost 12 percent.

Table 5.18: Sensitivity of change of C_{max} to the change of T_{ij}^2

Problem	[7,9]	[10,12]	[13,15]	[16,18]	[19,21]	Decrease in Cmax (%)
5 Machines 10 Job	532.2	519.6	527.6	527.4	514.0	3.42%
10 Machines 40 Jobs	2303.0	2245.6	2197.4	2118.2	2079.2	9.72%
10 Machines 60 Jobs	3205.8	3206.6	3047.8	3078.2	3045.2	5.01%
10 Machines 100 Jobs	5383.4	5277.4	5115.8	4868.2	4747.8	11.81%

Figure 5.14 presents how the changes in the values of T_{ij}^2 impact the average waiting time of each patient for metaheuristic solutions. The figure suggests an increase in the values of T_{ij}^2 increases the average waiting time of each patient. The increase in the waiting times holds for all problems. One implication can be that any increase in the values of T_{ij}^2 (or maximum waiting times) would increase

the potential waiting times. This increase in maximum waiting times before each machine can help the algorithm to find the best values of decision variables for minimizing makespan. It is observed that generally if patients increase the maximum waiting times between successive operations (T^2_{ij}), the average time they wait would increase. Another observation is that the slope of the average waiting time of each patient changes in different intervals and in the values close to the T^1_{ij}, an increase in the values of T^2_{ij} results in greater increase in the average waiting time.

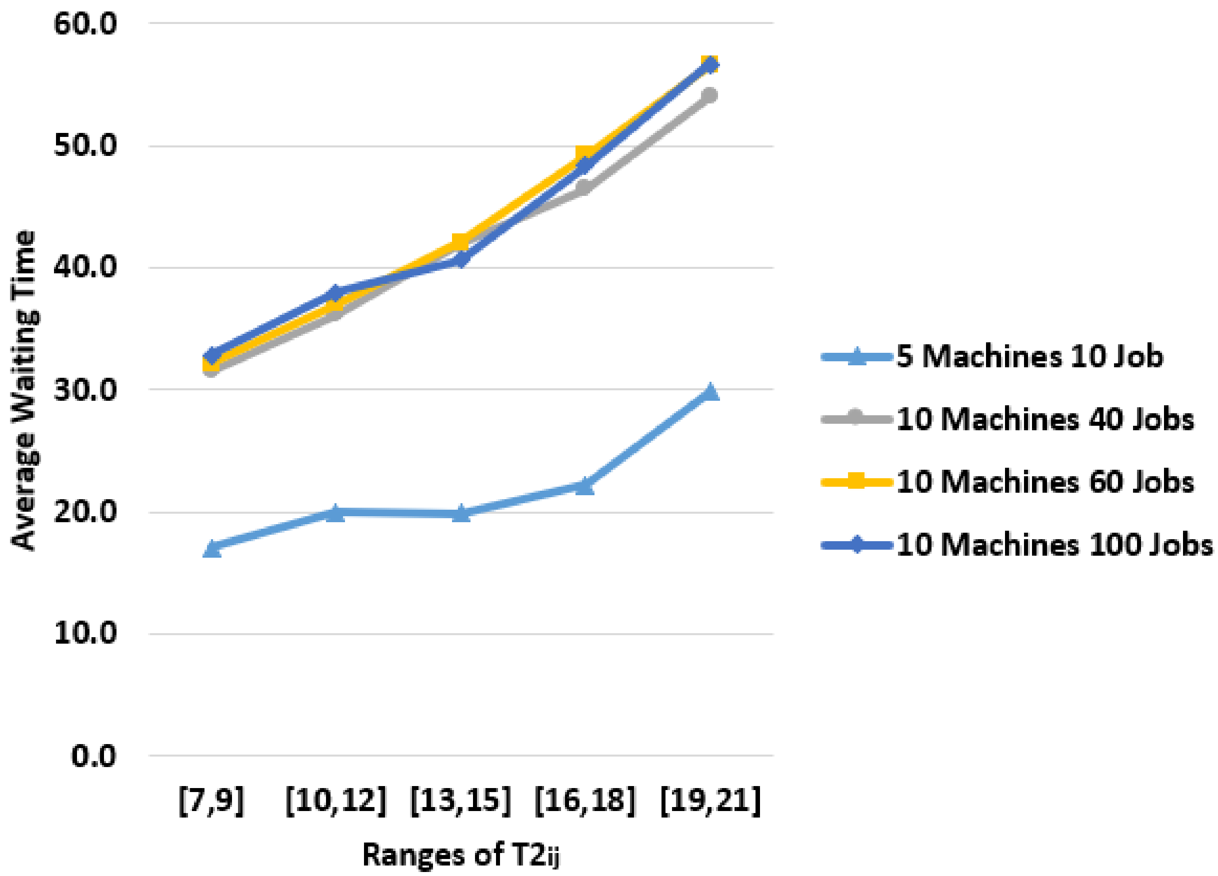

Figure 5.14: Sensitivity of average waiting time of each patient to the changes in the values T^2_{ij}

Numerical values of the average waiting times have been calculated and presented in table 5.19. The percentage of increase in the average waiting time after increasing T^2_{ij} is presented in the last column. The values suggest that an average increase in the range of 71 to 76 % is calculated as the increase in average waiting time.

Table 5.19: Sensitivity of change of the Average Waiting Time to the change of T_{ij}^2

Problem	[7,9]	[10,12]	[13,15]	[16,18]	[19,21]	Increase in Average Waiting Time (%)
5 Machines 10 Job	17.1	20.0	19.9	22.2	29.8	74.71%
10 Machines 40 Jobs	31.6	36.1	41.9	46.4	54.0	71.22%
10 Machines 60 Jobs	32.2	37.0	42.1	49.2	56.5	75.82%
10 Machines 100 Jobs	32.8	38.0	40.6	48.3	56.7	72.70%

6 Future Research and Conclusions

6.1 Simulation optimization

Simulation is an approach to address stochastic behavior in different systems. Any optimization problem can be described as follows (Jian and Henderson, 2015) :

$$Min\ f(x)$$

$$x \in \Theta$$

Where f is the objective function to optimize decision variable x subject to constraint $x \in \Theta$. If at least one of the objective function, decision variables and constraints are not deterministic then the problem is a simulation optimization problem. In the problem under study, some cases that can make the problem a simulation optimization are as follows:

1. By using the objective function of the current problem, if processing times and/or minimal and maximal time lags are stochastic variables, then simulation optimization can be used. Then one approach in solving the problem is to use the expected values of stochastic variables. In the current scenario, it can be assumed that arrival times are deterministic.

2. Since in the real world, arrival times are stochastic, one other potential scenario would be to consider stochastic arrival times for the patients. Then the problem consists of potential n types of the patients and m machines, and objective functions can be the maximizing utilization of each machine and/or the minimizing average $Cmax$ for each of patients, minimizing tardiness of each machine. In this case, the sequence of the machines or the ideal time-lags can be tested by the model. Utilization of each machine, in which in many cases are physicians, or high paying staffs can provide valuable information to the healthcare decision makers. Analysis of physician utilization can significantly decrease the system costs by balancing the number of resources or help decision makers to make better decisions by designing the system and assigning the resources in such a way that no resource is overused or have high idle times.

6.2 Other exact methods

One other approach can be applying more exact solutions to the problem. One desirable method can be logic-based Benders decomposition by combining MILP and CP to investigate other exact solutions (Hooker and Ottosson, 2003). Some other techniques that can be used are Benders Decomposition (Benders, 1962) and Column Generation method (Barnhart, Johnson, Nemhauser, Savelsbergh, and Vance, 1998). An improvement in the solutions may be achieved if these methods are used. It is possible to investigate heuristics that can solve the model under study.

6.3 Multi-objective models

In the current research work, the objective function was optimizing only one objective function; however, other objective functions may be considered with makespan to develop a multi-objective optimization function. Objective functions such as minimizing of maximum waiting time of job J, maximizing utilization of each machine or minimizing idle time of each machine. There might be a need to make some adjustments in the current model such as adding more constraints to develop a multi-objective model. The outcome of the multi-objective optimization can be a Pareto front of the best dominant solutions.

The analytic Hierarchy Process(AHP) method is a method that finds the weights of different criteria using multiple comparison (Saaty, 2008). AHP can be used to measure weights for a weighted sum multi-objective model. We can change the multi-objective problem into a single objective optimization model.

6.4 Other scheduling models

In this research, it is assumed that the model is a permutation flowshop model and some assumptions were proposed to support the permutation flowshop. Permutation flowshop needs the same flow of jobs in all machines, but in healthcare systems, there might be situations that patients do not follow up the same flow in different healthcare resources. A flowshop model that jobs can have different orders on different machines can be an interesting and highly applicable problem to investigate. The advantage of flowshop is that it can be applied to many systems that patients do not follow the same procedure. Also, a flowshop model can provide a better solution, but there are systems that need to consider permutation flowshop models. Other scheduling models such as openshop or job shop are other potential models that tackling them will need huge computational works, and solving them may require developing new meta-heuristic or heuristic algorithms.

6.5 Test the models with real data

One potential promising research is to test all of the models proposed in this research and the ones proposed in the future research with the real data. One of the limitations of this research is the lack of real data. In this research, because of the time limitation and the issues to access to real data, real data were not used; but testing the proposed model with real data, for example a sample of data collected in a clinic or a hospital is a really interesting area of research that can result in very practical and theoretical research outcomes. By applying the model in a real problem, specially, measuring different values of T_{ij}^1, a parameter mostly determined by decision makers and T_{ij}^2, a subjective parameter mostly determined by patients, the effect of changes of constraints of the system and preferences of decision makers and patients can be calculated in a real world problem.

6.6 Application of the model in other sectors

The model with minimal and maximal time lags and a maximum waiting time of each job is developed in this research. The model is developed mainly for healthcare systems, but there are other sectors that their decision makers try to minimize the makespan of the customers or other stakeholders of their systems with the constraints of minimal and maximal time lags and a maximum waiting time for each customer. Sectors such as transportation systems, banks, different organizations or manufacturing systems are other areas of the application of the proposed model in this research.

6.7 Conclusions

In this report, initially, healthcare scheduling problems were reviewed. Generally, research studies in the area of scheduling in healthcare can be divided into the following five categories"

- Number of objectives
- Type of patients
- Objective function to be optimized
- Problem parameters
- Type of solution

Then constraint programming and MILP models that consider minimum and maximum time-lags and constraint for the total waiting time of the patient were proposed. For solving the models, a genetic algorithm is developed and solved. The algorithm considered patient dissatisfaction coefficient in the objective function as a penalty to the C_{max}. In this study, the coefficient was considered as 1. Two intensification algorithms based on the genetic algorithm and using MILP and constraint programming were developed. These algorithms can improve the results of the genetic algorithm. The intensification algorithm was developed by breaking down the problem into the smaller problems, and after finding the optimal sequence in the small problems, the sequence of the large problem was updated. The findings in this report suggest that an improvement of the results of genetic algorithm can be achieved by using the intensification.

After solving the models, a sensitivity analysis is constructed on how the changes in the values of minimum and maximum time-lags impact the performance metrics of a healthcare system such as the effect of their change on the values of C_{max} and average waiting time and the variance of waiting times.

In the sensitivity analysis, for the exact solutions, five different small problems and for the metaheuristic solutions, four different problems of different sizes are considered. The findings suggest that as minimum time-lag increases, the C_{max} and the average waiting time of each patient increases. The minimum time-lag is the necessary waiting time of each patient between consecutive operations, and the maximum time-lag is the preference or the needs of the patient who do not desire to wait for more than specific amount of time. An increase in maximum time-lag can cause a decrease in the C_{max} and increase in the average waiting time. Another finding is generally, that the variance of waiting times of the patients increases as maximum time-lag increases and decreases as minimum time-lag increases. This can have practical implications for decision-makers of the healthcare systems.

References

Akers, S. B. (1956). Letter to the editor—a graphical approach to production scheduling problems. *Operations Research*, *4*(2), 244–245.

Augusto, V., Xie, X., and Perdomo, V. (2010). Operating theatre scheduling with patient recovery in both operating rooms and recovery beds. *Computers and Industrial Engineering*, *58*(2), 231–238.

Azcárate, C., Mallor, F., and Gafaro, A. (2008). Multiobjective optimization in health care management. a metaheuristic and simulation approach. *Algorithmic Operations Research*, *3*(2).

Babashov, V., Aivas, I., Begen, M., Cao, J., Rodrigues, G., D'Souza, D., ... Zaric, G. (2017). Reducing patient waiting times for radiation therapy and improving the treatment planning process: a discrete-event simulation model (radiation treatment planning). *Clinical Oncology*, *29*(6), 385–391.

Baesler, F. F., and Sepúlveda, J. A. (2001). Multi-objective simulation optimization for a cancer treatment center. In *Simulation conference, 2001. proceedings of the winter* (Vol. 2, pp. 1405–1411).

Bard, J. F., Shao, Y., Qi, X., and Jarrah, A. I. (2014). The traveling therapist scheduling problem. *IIE Transactions*, *46*(7), 683–706.

Bard, J. F., Shu, Z., Morrice, D. J., Leykum, L. K., and Poursani, R. (2016). Annual block scheduling for family medicine residency programs with continuity clinic considerations. *IIE Transactions*, *48*(9), 797–811.

Barnhart, C., Johnson, E. L., Nemhauser, G. L., Savelsbergh, M. W., and Vance, P. H. (1998). Branch-and-price: Column generation for solving huge integer programs. *Operations research*, *46*(3), 316–329.

Barua, B., Esmail, N., and Jackson, T. (2014). *The effect of wait times on mortality in Canada*. Fraser Institute Vancouver.

Beaulieu, H., Ferland, J. A., Gendron, B., and Michelon, P. (2000). A mathematical programming approach for scheduling physicians in the emergency room. *Health care Management Science*, *3*(3), 193–200.

Beliën, J., Demeulemeester, E., and Cardoen, B. (2009). A decision support system for cyclic master surgery scheduling with multiple objectives. *Journal of scheduling*, *12*(2), 147–161.

Benders, J. F. (1962). Partitioning procedures for solving mixed-variables programming problems. *Numerische Mathematik*, *4*(1), 238–252.

Berrada, I., Ferland, J. A., and Michelon, P. (1996). A multi-objective approach to nurse scheduling with both hard and soft constraints. *Socio-Economic Planning Sciences*, *30*(3), 183–193.

Bikker, I. A., Kortbeek, N., van Os, R. M., and Boucherie, R. J. (2015). Reducing access times for radiation treatment by aligning the doctor's schemes. *Operations Research for Health Care*, *7*, 111–121.

Blickle, T., and Thiele, L. (1995). *A comparison of selection schemes used in genetic algorithms.* TIK-Report.

Bowman, E. H. (1956). Production scheduling by the transportation method of linear programming. *Operations Research*, *4*(1), 100–103.

Büyüközkan, G., Çifçi, G., and Güleryüz, S. (2011). Strategic analysis of healthcare service quality using fuzzy ahp methodology. *Expert Systems with Applications*, *38*(8), 9407–9424.

Canadian Medical Association. (2008). The economic cost of wait times in Canada. *Prepared for Canadian Medical Association by Centre for Spatial Economics. Milton, Ontario, Canada*.

Cardoen, B., Demeulemeester, E., and Beliën, J. (2009). Optimizing a multiple objective surgical case sequencing problem. *International Journal of Production Economics*, *119*(2), 354–366.

Chandra, C., Liu, Z., He, J., and Ruohonen, T. (2014). A binary branch and bound algorithm to minimize maximum scheduling cost. *Omega*, *42*(1), 9–15.

Chern, C.-C., Chien, P.-S., and Chen, S.-Y. (2008). A heuristic algorithm for the hospital health examination scheduling problem. *European Journal of Operational Research*, *186*(3), 1137–1157.

Dahlen, I., Westin, L., and Adolfsson, A. (2012). Experience of being a low priority patient during waiting time at an emergency department. *Psychology Research and Behavior Management*, *5*, 1.

Daldoul, D., Nouaouri, I., Bouchriha, H., and Allaoui, H. (2018). A stochastic model to minimize patient waiting time in an emergency department. *Operations Research for Health Care*, *18*, 16–25.

Dansky, K. H., and Miles, J. (1997). Patient satisfaction with ambulatory healthcare services: waiting time and filling time. *Journal of Healthcare Management*, *42*(2), 165.

Denton, B., Viapiano, J., and Vogl, A. (2007). Optimization of surgery sequencing and scheduling decisions under uncertainty. *Health care management science*, *10*(1), 13–24.

Deslauriers, S., Raymond, M.-H., Laliberté, M., Lavoie, A., Desmeules, F., Feldman, D. E., and Perreault, K. (2017). Access to publicly funded outpatient physiotherapy services in quebec: waiting lists and management strategies. *Disability and rehabilitation*, *39*(26), 2648–2656.

Dhouib, E., Teghem, J., and Loukil, T. (2013). Lexicographic optimization of a permutation flow shop scheduling problem with time lag constraints. *International Transactions in Operational Research*, *20*(2), 213–232.

Dimakou, S., Parkin, D., Devlin, N., and Appleby, J. (2009). Identifying the impact of government targets on waiting times in the nhs. *Health Care Management Science*, *12*(1), 1.

Doulabi, S. H. H., Rousseau, L.-M., and Pesant, G. (2014). A constraint programming-based column generation approach for operating room planning and scheduling. In *International conference on ai and or techniques in constriant programming for combinatorial optimization problems* (pp. 455–463).

Elomri, A., Elthlatiny, S., and Sidi Mohamed, Z. (2015). A goal programming model for fairly scheduling medicine residents. *International Journal of Supply Chain Management*, *4*(2).

Fernandez-Viagas, V., Ruiz, R., and Framinan, J. M. (2017). A new vision of approximate methods for the permutation flowshop to minimise makespan: State-of-the-art and computational evaluation. *European Journal of Operational Research*, *257*(3), 707–721.

Fondrevelle, J., Oulamara, A., and Portmann, M.-C. (2006). Permutation flowshop scheduling problems with maximal and minimal time lags. *Computers and Operations Research*, *33*(6), 1540–1556.

Fügener, A., Hans, E. W., Kolisch, R., Kortbeek, N., and Vanberkel, P. T. (2014). Master surgery scheduling with consideration of multiple downstream units. *European Journal of Operational Research*, *239*(1), 227–236.

Garey, M. R., Johnson, D. S., and Sethi, R. (1976). The complexity of flowshop and jobshop scheduling. *Mathematics of Operations Research*, *1*(2), 117-129. Retrieved from `https://doi.org/10.1287/moor.1.2.117` doi: 10.1287/moor.1.2.117

Govindan, K., Balasundaram, R., Baskar, N., and Asokan, P. (2017). A hybrid approach for minimizing makespan in permutation flowshop scheduling. *Journal of Systems Science and Systems Engineering*, *26*(1), 50–76.

Graham, R. L., Lawler, E. L., Lenstra, J. K., and Kan, A. R. (1979). Optimization and approximation in deterministic sequencing and scheduling: a survey. *Annals of Discrete Mathematics*, *5*, 287–326.

Grant, B., Gurvich, I., Mutharasan, R. K., and Van Mieghem, J. (2017, Aug 10). Optimal dynamic appointment scheduling of base and surge capacity. Retrieved from `https://ssrn.com/abstract=3092667`

Guido, R., and Conforti, D. (2016). A hybrid genetic approach for solving an integrated multi-objective operating room planning and scheduling problem. *Computers and Operations Research*.

Gupta, D., and Denton, B. (2008). Appointment scheduling in health care: Challenges and opportunities. *IIE transactions*, *40*(9), 800–819.

Hahn-Goldberg, S., Carter, M. W., Beck, J. C., Trudeau, M., Sousa, P., and Beattie, K. (2014). Dynamic optimization of chemotherapy outpatient scheduling with uncertainty. *Health Care Management Science*, *17*(4), 379–392.

Haleh, H., Maghsoudlou, H., Hadipour, H., and Nabovati, H. (2017). Scheduling single machine with random breakdown and preemptive jobs. *Journal of Industrial and Production Engineering*, *34*(4), 289–299.

Hall, R. W. (2012). *Handbook of healthcare system scheduling*. Springer.

Hamdi, I., and Loukil, T. (2011). Minimizing the makespan in the permutation flowshop problem with minimal and maximal time lags. In *Communications, computing and control applications (ccca), 2011 international conference on* (pp. 1–6).

Hancock, D. R., and Algozzine, B. (2016). *Doing case study research: A practical guide for beginning researchers*. Teachers College Press.

Health Quality Ontario. (2018, April). *Emergency department time spent by patients in ontario - health quality ontario (hqo).* Retrieved 2018-July-08, from `http://www.hqontario.ca/System-Performance/Time-spent-in-emergency-departments`

Hodson, A., Muhlemann, A., and Price, D. (1985). A microcomputer based solution to a practical scheduling problem. *Journal of the Operational Research Society*, *36*(10), 903–914.

Hooker, J. N., and Ottosson, G. (2003). Logic-based benders decomposition. *Mathematical Programming*, *96*(1), 33–60.

Huang, Y.-L. (2008). An alternative outpatient scheduling system: Improving the outpatient experience.

Huang, Y.-L., and Bach, S. M. (2016). Appointment lead time policy development to improve patient access to care. *Applied Clinical Informatics*, *7*(4), 954.

Jaumard, B., Semet, F., and Vovor, T. (1998). A generalized linear programming model for nurse scheduling. *European Journal of Operational Research*, *107*(1), 1–18.

Jian, N., and Henderson, S. G. (2015, Dec). An introduction to simulation optimization. In *2015 winter simulation conference (wsc)* (p. 1780-1794). doi: 10.1109/WSC.2015.7408295

Joanes, D., and Gill, C. (1998). Comparing measures of sample skewness and kurtosis. *Journal of the Royal Statistical Society: Series D (The Statistician)*, *47*(1), 183–189.

Johnson, S. M. (1954). Optimal two-and three-stage production schedules with setup times included. *Naval research logistics quarterly*, *1*(1), 61-68.

Karamyar, F., Sadeghi, J., and Yazdi, M. M. (2016). A benders decomposition for the location-allocation and scheduling model in a healthcare system regarding robust optimization. *Neural Computing and Applications*, 1–14.

Laesanklang, W., Landa-Silva, D., and Salazar, J. A. C. (2015). Mixed integer programming with decomposition to solve a workforce scheduling and routing problem. In *Proceedings of the international conference on operations research and enterprise systems* (p. 283-293). doi: 10.5220/0005223602830293

Latorre-Núñez, G., Lüer-Villagra, A., Marianov, V., Obreque, C., Ramis, F., and Neriz, L. (2016). Scheduling operating rooms with consideration of all resources, post anesthesia beds and emergency surgeries. *Computers and Industrial Engineering*, *97*, 248–257.

Liang, B., and Turkcan, A. (2015). Acuity-based nurse assignment and patient scheduling in oncology clinics. *Health care management science*, 1–20.

Liang, B., Turkcan, A., Ceyhan, M. E., and Stuart, K. (2015). Improvement of chemotherapy patient flow and scheduling in an outpatient oncology clinic. *International Journal of Production Research*, *53*(24), 7177–7190.

Lim, G. J., Mobasher, A., Bard, J. F., and Najjarbashi, A. (2016). Nurse scheduling with lunch break assignments in operating suites. *Operations Research for Health Care*, *10*, 35–48.

Lim, G. J., Mobasher, A., and Côté, M. J. (2012). Multi-objective nurse scheduling models with patient workload and nurse preferences. *Management*, *2*(5), 149–160.

Luh, P. B., and Hoitomt, D. J. (1993). Scheduling of manufacturing systems using the lagrangian relaxation technique. *IEEE Transactions on Automatic Control*, *38*(7), 1066–1079.

Marynissen, J., and Demeulemeester, E. (2016). *Literature review on integrated hospital scheduling problems* (Tech. Rep.). KU Leuven, Faculty of Economics and Business. Retrieved from `https://ssrn.com/abstract=2873413`

Matis, T., Farris, J., McAllister, M., Dunavan, C., and Snider, A. (2015). Target times for inpatient discharge scheduling. *IIE Transactions on Healthcare Systems Engineering*, *5*(1), 33–41.

Meskens, N., Duvivier, D., and Hanset, A. (2013). Multi-objective operating room scheduling considering desiderata of the surgical team. *Decision Support Systems*, *55*(2), 650–659.

Ministry of Health and Long-Term Care . (2008, May). *Ontario's $109 millon investment to reduce wait times in the emergency room.* Retrieved 2018-August-01, from `http://files.news.ontario.ca.s3-website-us-east-1.amazonaws.com/mohltc/en/learnmore/ontario_tackles_er_waits_with_109_million_investment/er_alc_strategy_combined_bg_04_20080529.pdf`

Montgomery, D. C. (2017). *Design and analysis of experiments*. John wiley and sons.

Nagano, M. S., Ruiz, R., and Lorena, L. A. N. (2008). A constructive genetic algorithm for permutation flowshop scheduling. *Computers and Industrial Engineering*, *55*(1), 195–207.

Nawaz, M., Enscore Jr, E. E., and Ham, I. (1983). A heuristic algorithm for the m-machine, n-job flow-shop sequencing problem. *Omega*, *11*(1), 91–95.

Nova Scotia Department of Health and Wellness. (2010, December). *Better care sooner the plan to improve emergency care* (Tech. Rep.). Retrieved from `https://novascotia.ca/dhw/publications/Better-Care-Sooner-plan.pdf`

Othman, S. B., Hammadi, S., and Quilliot, A. (2015). Multi-objective evolutionary for multi-skill health care tasks scheduling. *IFAC-PapersOnLine*, *48*(3), 704–709.

Petrovic, D., Morshed, M., and Petrovic, S. (2011). Multi-objective genetic algorithms for scheduling of radiotherapy treatments for categorised cancer patients. *Expert Systems with Applica-*

tions, *38*(6), 6994–7002.

Prentice, J. C., and Pizer, S. D. (2007). Delayed access to health care and mortality. *Health Services Research*, *42*(2), 644–662.

Qu, X., Peng, Y., Kong, N., and Shi, J. (2013). A two-phase approach to scheduling multi-category outpatient appointments–a case study of a women's clinic. *Health Care Management Science*, *16*(3), 197–216.

Rahman, H. F., Sarker, R., and Essam, D. (2015). A genetic algorithm for permutation flow shop scheduling under make to stock production system. *Computers and Industrial Engineering*, *90*, 12–24.

Rahmani, D., and Ramezanian, R. (2016). A stable reactive approach in dynamic flexible flow shop scheduling with unexpected disruptions: A case study. *Computers and Industrial Engineering*, *98*, 360–372.

Rajendran, S., Rajendran, C., and Leisten, R. (2017). Heuristic rules for tie-breaking in the implementation of the neh heuristic for permutation flow-shop scheduling. *International Journal of Operational Research*, *28*(1), 87–97.

Riise, A., Mannino, C., and Lamorgese, L. (2016). Recursive logic-based benders' decomposition for multi-mode outpatient scheduling. *European Journal of Operational Research*, *255*(3), 719–728.

Ruiz, R., and Stützle, T. (2007). A simple and effective iterated greedy algorithm for the permutation flowshop scheduling problem. *European Journal of Operational Research*, *177*(3), 2033–2049.

Saaty, T. L. (2008). Decision making with the analytic hierarchy process. *International Journal of Services Sciences*, *1*(1), 83–98.

Sadki, A., Xie, X., and Chauvin, F. (2011). Appointment scheduling of oncology outpatients. In *Automation science and engineering (case), 2011 ieee conference on* (pp. 513–518).

Saremi, A., Jula, P., ElMekkawy, T., and Wang, G. G. (2015). Bi-criteria appointment scheduling of patients with heterogeneous service sequences. *Expert Systems with Applications*, *42*(8), 4029–4041.

Singh, A., and Prasher, A. (2017). Measuring healthcare service quality from patients' perspective: using fuzzy ahp application. *Total Quality Management and Business Excellence*, 1–17.

Sinko, A., Nikolova, S., Sutton, M., et al. (2015). *Targets for maximum waiting times and patient prioritisation: evidence from england* (Tech. Rep.).

Smalley, H. K., and Keskinocak, P. (2016). Automated medical resident rotation and shift scheduling to ensure quality resident education and patient care. *Health care management Science*, *19*(1), 66–88.

Srinivas, S., and Ravindran, A. R. (2018). Optimizing outpatient appointment system using machine learning algorithms and scheduling rules: A prescriptive analytics framework. *Expert Systems with Applications*, *102*, 245–261.

Topaloglu, S. (2006). A multi-objective programming model for scheduling emergency medicine residents. *Computers and Industrial Engineering*, *51*(3), 375–388.

Vallada, E., Ruiz, R., and Framinan, J. M. (2015). New hard benchmark for flowshop scheduling problems minimising makespan. *European Journal of Operational Research*, *240*(3), 666–677.

Vermeulen, M. J., Stukel, T. A., Boozary, A. S., Guttmann, A., and Schull, M. J. (2016). The effect of pay for performance in the emergency department on patient waiting times and quality of care in ontario, Canada: a difference-in-differences analysis. *Annals of Emergency Medicine*, *67*(4), 496–505.

Wait Time for Canada Wait Time Alliance. (2014). `http://www.waittimealliance.ca/benchmarks/`. (Accessed: 2019-01-09)

Wang, B., Huang, K., and Li, T. (2018). Permutation flowshop scheduling with time lag constraints and makespan criterion. *Computers and Industrial Engineering*, *120*, 1–14.

Willcox, S., Seddon, M., Dunn, S., Edwards, R. T., Pearse, J., and Tu, J. V. (2007). Measuring and reducing waiting times: a cross-national comparison of strategies. *Health Affairs*, *26*(4), 1078–1087.

Zhao, L., Chien, C.-F., and Gen, M. (2015). A bi-objective genetic algorithm for intelligent rehabilitation scheduling considering therapy precedence constraints. *Journal of Intelligent Manufacturing*, 1–16.