# ENERGY CONSCIOUS MACHINE AND ROBOT SPEED DECISIONS IN TWO MACHINE ROBOTIC CELL SCHEDULING

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

BETÜL NECİBE GÜZEL

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
INDUSTRIAL ENGINEERING

JUNE 2021

Approval of the thesis:

**ENERGY CONSCIOUS MACHINE AND ROBOT SPEED DECISIONS IN TWO MACHINE ROBOTIC CELL SCHEDULING**

submitted by **BETÜL NECİBE GÜZEL** in partial fulfillment of the requirements for the degree of **Master of Science in Industrial Engineering Department, Middle East Technical University** by,

Prof. Dr. Halil Kalıpçılar
Dean, Graduate School of **Natural and Applied Sciences** ______________

Prof. Dr. Esra Karasakal
Head of Department, **Industrial Engineering** ______________

Prof. Dr. Sinan Gürel
Supervisor, **Industrial Engineering, METU** ______________

**Examining Committee Members:**

Assoc. Prof. Dr. İsmail Serdar Bakal
Industrial Engineering, METU ______________

Prof. Dr. Sinan Gürel
Industrial Engineering, METU ______________

Assist. Prof. Dr. Selçuk Gören
Industrial Engineering, Abdullah Gül University ______________

Assist. Prof. Dr. Diclehan Tezcaner Öztürk
Industrial Engineering, Hacettepe University ______________

Assist. Prof. Dr. Mustafa Kemal Tural
Industrial Engineering, METU ______________

Date:

**I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.**

Name, Surname: BETÜL NECİBE GÜZEL

Signature :

# ABSTRACT

## ENERGY CONSCIOUS MACHINE AND ROBOT SPEED DECISIONS IN TWO MACHINE ROBOTIC CELL SCHEDULING

GÜZEL, BETÜL NECİBE
M.S., Department of Industrial Engineering
Supervisor: Prof. Dr. Sinan Gürel

June 2021, 63 pages

Minimizing cycle time has always been a critical objective in manufacturing cells. Recently, energy efficient and environmentally conscious manufacturing practices have received increasing attention from researchers and practitioners. Therefore, in this thesis we consider a 2-machine manufacturing cell with a material handling robot where two conflicting objectives exist: minimization of cycle time and energy consumption. It is a flow type cell where identical parts are first processed on the first machine and then on the second machine. All handling operations between part buffers and machines and loading/unloading operations are done by a robot. Most of the research on robotic cell scheduling problems focus on robot activity sequencing decisions and assume that robot and machines operate at a fixed pace. However, both robot's speed and machines' processing times are controllable and they affect the cycle time and energy consumption performance of a cell. To the best of our knowledge, this is the first study that consider robot and machine speed controllability at the same time. We study three well known cyclic robot activity sequences and develop mathematical models that find efficient solutions for the problem. Our analysis on mathematical models show that energy optimization with robot speed and machine processing time control leads to schedules in which robot's arrival to a machine for unloading operation and part completion time on the machine are synchronized. Furthermore, we provide a numerical study that explores which robotic activity sequence is better when and how much energy saving can be achieved by employing robot speed and processing time control strategies.

Keywords: Robotic cell, Scheduling, Nonlinear optimization, Energy consumption, Controllable processing times, Robot speed control

## ÖZ

**İKİ MAKİNELİ BİR ROBOTİK HÜCREDE ENERJİ ETKİN MAKİNE VE ROBOT HIZ KARARLARI İLE ÇİZELGELEME**

GÜZEL, BETÜL NECİBE
Yüksek Lisans, Endüstri Mühendisliği Bölümü
Tez Yöneticisi: Prof. Dr. Sinan Gürel

Haziran 2021 , 63 sayfa

Üretim hücrelerinin minimum çevrim süresi ile çalıştırılması her zaman kritik bir amaç olmuştur. Son zamanlarda, enerji verimli ve çevreye duyarlı üretim uygulamaları araştırmacılar ve uygulayıcılar tarafından giderek artan bir ilgi görmeye başladı. Bu nedenle, bu tezde, bir adet elleçleme robotuna sahip iki makineli bir üretim hücresini birbiriyle çelişen iki amaç varlığında ele alıyoruz: çevrim süresi ve enerji tüketimi. Ele alınan robotik hücre, özdeş parçaların önce birinci makinede daha sonra ikinci makinede işlendiği akış tipi bir hücredir. Parça stok alanları ve makineler arasındaki tüm taşıma işlemleri ve yükleme/boşaltma işlemleri bir robot tarafından gerçekleştirilir. Robotik hücre çizelgeleme problemleri üzerine yapılan araştırmaların çoğu robot aktivite sıralama kararlarına odaklanır ve robot ile makinelerin sabit bir hızda çalıştığını varsayar. Fakat, hem robot hızı hem de makine işlem süresi kontrol edilebilirdir ve çizelgenin çevrim süresini ve enerji tüketimini etkilerler. Bildiğimiz kadarıyla, bu çalışma robot ve makine hız kontrolünü aynı anda ele alan ilk çalışmadır. Bu çalışmada, iyi bilinen üç çevrimsel robot hareket dizisini inceledik ve problem için etkin çözümler bulan matematiksel modeller geliştirdik. Matematiksel modeller üzerindeki analizlerimiz, robot hızı ve makine işlem süresi kontrolü ile sağladığımız enerji optimizasyonunun, robotun boşaltma işlemi için bir makineye gelişi ile makinedeki parça tamamlanma süresinin senkronize edildiği çizelgelerle mümkün olduğunu göstermektedir. Ayrıca, robot hızı ve işlem süresi kontrol stratejileri kullanılarak hangi robotik hareket çizelgesinin ne zaman tercih edilebilir olduğunu ve ne kadar enerji tasarrufu sağlanabileceğini keşfetmek için hesaplamalı çalışmalar yürüttük.

Anahtar Kelimeler: Robotik hücre, Çizelgeleme, Doğrusal olmayan optimizasyon, Enerji tüketimi, Kontrol edilebilir işlem zamanı, Robot hız kontrolü

To my dear family and beloved fiancé Anıl Yılmaz, who brought light to my life...

## ACKNOWLEDGMENTS

First of all, I would like to thank to my supervisor Prof. Dr. Sinan Gürel for his patient guidance and useful critiques in this process. Also, I am very grateful to him for his positive attitude all the time. He believed in me when I lost my hopes and consistently steered me in the right direction whenever he thought I needed it. It was a great privilege to work and study under his guidance.

I would like to thank my company TUSAŞ INC. for letting me have time for this study. I felt continuous support of all my managers and workmates during my study. I would also like to gratefully thank M. Egemen Çolak, Ceren Sungurlu and especially V. Cenk Karakuz for their invaluable helps, motivating and understanding behaviours.

I would like to thank to Scientific and Technological Research Council of Turkey (TÜBİTAK), for providing the financial support for my MSc study with 2210/A National Graduate Scholarship Program.

I wish to express my deepest gratitude to my beloved family members, my mother Hacer Güzel, my father Ahmet Güzel and my brother Erdem Güzel for their endless love, all the sacrifices they have made so far and faith in whatever I pursue. I am very lucky to have such a great family. I wholeheartedly indebted to my loving and supportive fiancé, Anıl Yılmaz, who never made me feel alone, provide unending inspiration and continuous encouragement throughout my study.

Last but not least, I would like to express my eternal gratitude to the founder of the Republic of Turkey: Mustafa Kemal Atatürk, who enlightened us with his principles and reforms and encouraged us for science. He not only provided me an independent research environment but also gave me courage, discipline, and faith.

# TABLE OF CONTENTS

## LIST OF TABLES

TABLES

## LIST OF FIGURES

FIGURES

# LIST OF ABBREVIATIONS

| | |
|---|---|
| KKT | Karush–Kuhn–Tucker |
| MINLP | Mixed Integer Non-Linear Programming |
| NLP | Non-Linear Programming |

# CHAPTER 1

# INTRODUCTION

Increasing the efficiency of production systems has always been an important topic for engineers and scientists. Efficiency in the industry, as a term, is considered to be the ability to produce the desired output with little or no wastage of time, manpower, materials, energy, etc. Recently, energy efficiency in manufacturing has started to attract increasing attention from researchers. Increasing energy prices, regulations on $CO_2$ consumption, increasing public awareness for green products can be shown among the reasons for the popularity of energy efficiency [15]. Also, according to BP's report [8], there is an ever-increasing energy consumption around the world. Experts estimate that increase in energy consumption will accelerate in the future. This thesis is an attempt to find energy efficient robotic schedules in a manufacturing cell.

U.S. Energy Information Administration points out that industry is the sector having the highest energy consumption level all around the world (see Figure 1.1). Fysikopoulos et. al. [29] mention that approximately 20%-40% of energy consumption in industry can be redundant. Considering the companies' increasing energy costs and the society's growing sensitivity to energy consumption we see more studies on energy efficient practices in different areas such as manufacturing and transportation.

As automation systems have become widespread in the industry, energy consumption and carbon footprint levels due to industrial robots have increased. According to International Federation of Robotics [41], especially in the last ten years, the demand for industrial robots has increased significantly. The energy consumption of robots in the automotive industry is accounted for about 8% of the total energy consumption in this sector [27]. Reducing energy consumption of robots and machines will provide

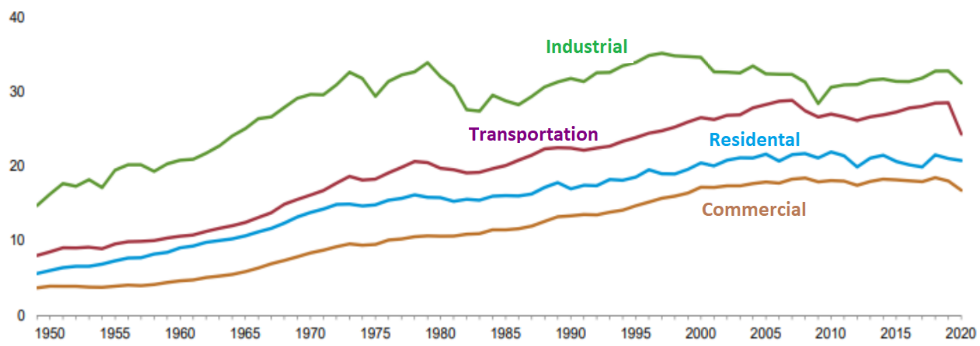


Figure 1.1: Energy consumption levels (Quadrillion Btu) by sectors [64]

significant gains in terms of cost, sustainability and will be appealing to environmentally conscious customers of production sector. Most importantly, by reducing energy consumption, the damage to natural resources and the environment will be reduced.

There are manufacturing systems where robots and machines are widely used, one of them is robotic cells. A robotic cell is a production system including one or more programmable material handling robots and machines. In a robotic cell, while the machines do manufacturing operations on a part, the robots perform material handling, loading and unloading tasks between the machines/buffers. When identical parts are produced, cyclic robot schedules are preferred. Cyclic scheduling problems which aim to decrease cycle time, manufacturing cost and energy consumption are a popular research area for the robotic cells.

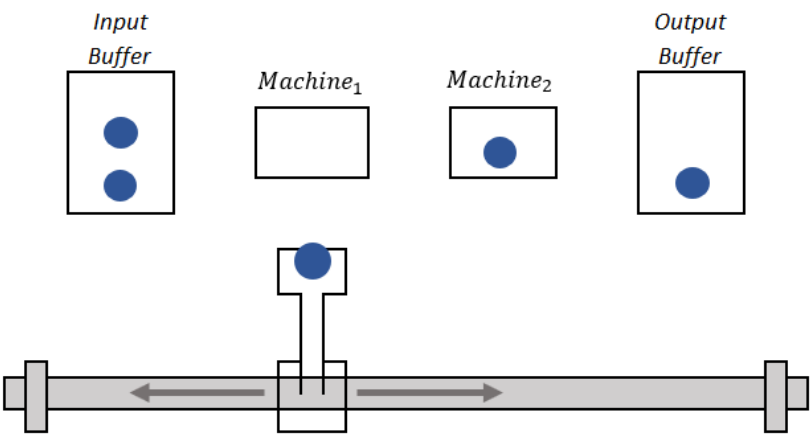


Figure 1.2: An example figure of 2-machine robotic cell

In this thesis, we study a manufacturing cell which has two machines, one input buffer, one output buffer and a material handling robot (see Figure 1.2). The robot in the cell moves on a line and carries the parts between buffers/machines. It is a flow type production. Parts are first processed on $Machine_1$ then on $Machine_2$. The two machines perform different operations on the part. The robot takes a part from a machine or a buffer and starts moving to take it to its next location. Thus, a cycle begins. Then, it performs the other tasks included in the cycle respectively. A robotic cycle is completed by the robot when the robot returns to its initial position to do the same operations in the same sequence again.

In this study, we study on 1-unit and 2-unit cycles, called $S_1$, $S_2$ and $S_{12}$. The robot move sequences in $S_1$, $S_2$ and $S_{12}$ cycles are given in Figure 1.3, 1.4, 1.5, respectively. Robot activities in cycles are explained in detail in the Section 3.2.

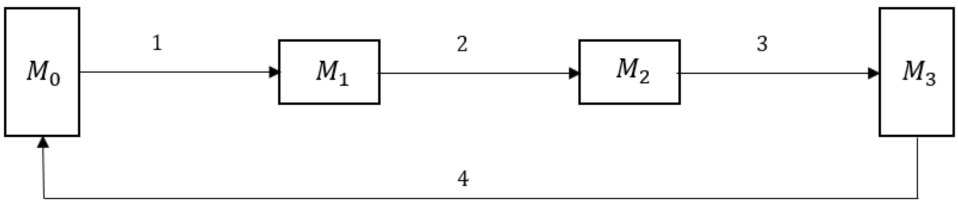


Figure 1.3: $S_1$ cycle

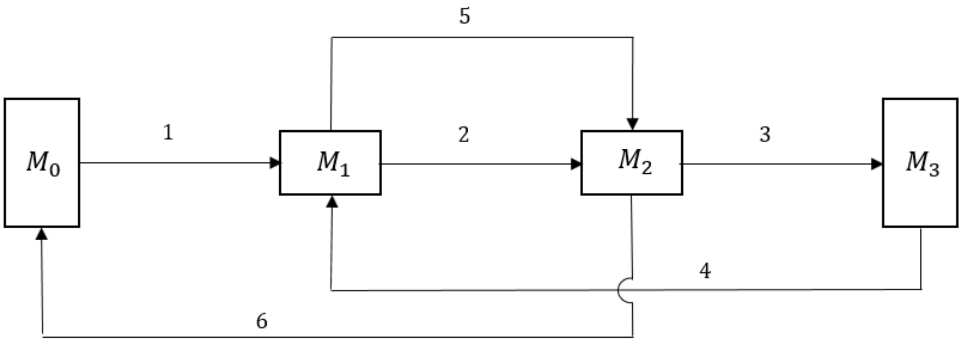


Figure 1.4: $S_2$ cycle

$S_1$ and $S_2$ cycles were introduced to the literature as 1-unit cycles. Sethi et al. [58] showed that $S_1$ and $S_2$ are the feasible cycles that give the best robot move sequence when cycle time minimization is considered in robotic cells with two machines pro-

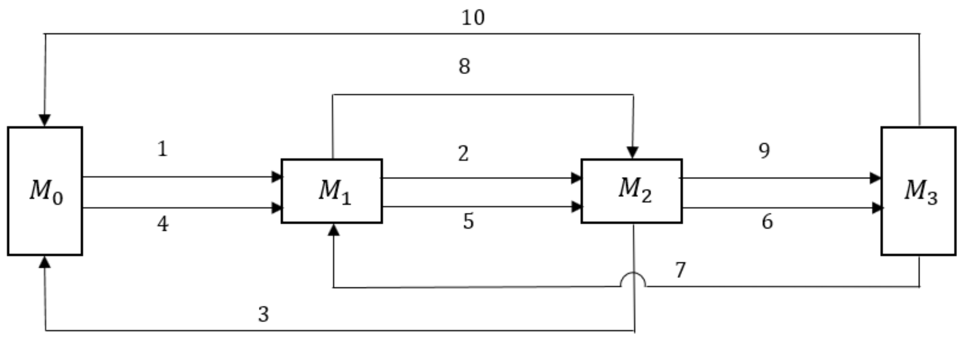


Figure 1.5: $S_{12}$ cycle

ducing identical parts. Then, Hall et al. [37] defined new robot cycles: $S_{12}$, $S_{21}$. Basically, these cycles are based on $S_1$ and $S_2$ cycles. In their study, they explained that the four cycles ($S_1$, $S_2$, $S_{12}$ and $S_{21}$) dominate the other cycles, and therefore any robot cycle can be defined with these cycles.

In this study, we consider energy consumption (both for robot and machines) and cycle time objectives at the same time in a two-machine robotic cell. Both robot and machine speeds are controllable and energy consumption of the cell depends on robot and machine speeds. We develop mathematical models that determine energy optimal schedules for a given cycle time for the selected robotic cycles. We give analytical results on these models and present an extensive numerical study.

Most of the robotic cell scheduling problems in the literature assume fixed robot speed and fixed machine processing times. There are few studies that consider energy consumption. These studies consider either robot speed control or machine processing time control but not both. In this thesis, we study two strategies at the same time.

The rest of the thesis is organized as follows. Chapter 2 gives a review of related studies in the literature. In Chapter 3, we give problem definition, mathematical models and their analysis. In Chapter 4, we present a numerical study. We give concluding remarks in Chapter 5.

# CHAPTER 2

# LITERATURE REVIEW

In this chapter, we present a review of related literature. First, in Section 2.1, we consider robotic cell scheduling problems. Within this context, we examine the development of the subject in the literature by focusing on early and recent studies on robotic cell scheduling. Then, in Section 2.2, we present the studies on energy consumption of industrial robots. Finally, in Section 2.3, we mention the studies which consider processing time control in machine scheduling literature.

## 2.1 Robotic Cell Scheduling

As mentioned before, robotic cell is a type of production system including machines and robots together. Robotic cells are classified according to some of their characteristics. Terms that are widely accepted in the literature and we have used throughout our study are presented by Dawande et al. [23]. For instance, robotic cells are called $m$-machine robotic cells according to the number of machines ($m$) in the cell. Other problem classifications are based on number of robots (i.e., single / multiple) or robot types (i.e., single or dual gripper) in a cell. There are also classifications depending on the robot's pickup selection, travel time options or variety of parts produced, etc.

Scheduling involves resource allocation, sequencing, timing decision for tasks/activities. There is a vast literature especially on machine scheduling problems. Lee et al. [47] provided an extensive survey on scheduling problem. They summarized the studies and the novel methods carried out in the field of scheduling until 1997. They also mentioned the studies on robotic cell scheduling problems in their study.

In the literature, number of robotic cell scheduling studies has increased with the widespread use of robotic cells in industry. Several literature surveys exist on robotic cell scheduling (see e.g., [19], [23], [24], [38], [48]). In one of these surveys, Levner et al. [48] pointed out the NP-hardness of robotic cell scheduling problems and the existence of unsolvable problems yet. Most of the work in robotic cell scheduling assume fixed robot speed and fixed machine speed. We give a short review of these studies in the next section.

### 2.1.1 Studies with fixed robot speed and fixed machine speed

Sethi et al. [58] studied finding the optimal robot activity sequences in 2-machine and 3-machine robotic cells with a material handling robot. They considered cycle time objective. They introduced two optimal 1-unit cycles namely $S_1$ and $S_2$ cycles which will be studied in this thesis. In a later study, Crama et. al. [20] generalized the study considered by Sethi et al. [58]. For an $m$-machine robotic cell with a material handling robot, where $m > 3$, they showed that it is possible to find the optimal cycle using dynamic programming which requires pseudo-polynomial time in $m$. For identical parts, Geismar et al. [31] studied the problem of sequencing operations in three-stage robotic cells with parallel machines and a material handling robot. They assumed that the robot's handling time between any two machines is constant and defined the cycle time with a general expression. Then, they found optimal cycles for considered robotic cell, and also constructed a formulation which provides how many machines are needed to meet the demand.

Hall et al. [37] studied scheduling problems for 2-machine and 3-machine robotic cells with a material handling robot in order to determine optimal robot cycle and part sequences at the same time. For a 2-machine robotic cell with multiple part types, they introduced an exact algorithm to solve the cycle time optimization problem. On the other hand, for 3-machine robotic cells producing identical parts, they concluded that 1-unit cycles give the lower cycle time than 2-unit cycles. Also, they showed that defining any feasible robot cycle can be possible by utilizing the four cycle types called $S_1$, $S_2$, $S_{12}$ and $S_{21}$. In this thesis, we consider $S_1$, $S_2$ cycles. In addition, we also consider a 2-unit cycle represented by the two cycles ($S_{12}$ and $S_{21}$) described by

Hall et al. [37]. We call this cycle as $S_{12}$ cycle in the rest of the study. Based on the study of Hall et al. [37], in the late 1990s and early 2000s, many studies considering similar problems were done (e.g., [2], [10], [21]). Brauner and Finke [11] studied the performance of 1-unit cycles for cycle time objective in an m-machine robotic cell with a circular layout. They showed that a 1-unit cycle is not always optimal by showing with a counterexample for a 4-machine robotic cell. In another study, Brauner [9] stated that the proof given by Brauner and Finke [11] can be generalized for robotic cells where $m \leq 15$.

Single robot cells are simple to design and operate, but multiple robot cells also exist. Alcaide et al. [5] studied multi-robot cell scheduling by developing the methods validated for single-robot cells. They presented a graph model based on critical path problem to solve multi-robot scheduling problems and developed a polynomial time algorithm. They carried out numerical experiments with one and two material handling robots for 6-machine robotic cells. As a different type of robotic cell, Srishkandarajah et al. [62] considered a 2-machine robotic cell having a dual-gripper material handling robot with the aim of increased long-run throughput rate. They also considered multiple part types and presented a heuristic approach to schedule robots and parts.

For multiple part types, Kamoun et al. [43] presented heuristic algorithms with the intent of obtaining optimal part sequences for 3-machine and 4-machine robotic cells with a material handling robot. Their objective is to obtain minimum average steady-state cycle time based on the specified robot cycles for 3-machine robotic cells. Besides the scheduling problem, they also worked on the design of a robotic cell. In robotic cells, when the number of machines or robots increase, and multiple part types are considered, systems can become more complex. Hence, we see that meta-heuristic algorithms are becoming more widely used in recent works (e.g., [26], [70]). For example, Abdulkader et al. [1] aimed the minimum robot cycle time in a 4-machine robotic cell served by a robot. They proposed a genetic algorithm finding effective robot move and tasks sequences. They considered both identical and multiple part types.

Existing studies differ from each other with respect to the robotic cell environment, such as the number of machines, the number and type of robots or the variety of part types. Most of the studies on robotic cell scheduling problem consider fixed robot speed and fixed processing times. Also, the majority of these studies focus on only one objective, specifically the cycle time minimization.

### 2.1.2 Studies with controllable robot speed

In robotic cell scheduling studies mentioned so far, robot move times and machine processing times are fixed. As cycle time is considered, operators may prefer to operate both machines and robots at their maximum speeds to achieve high levels of throughput. However, this requires higher energy consumption. For example, Kobetski and Fabian [45] in their study stated that for flexible manufacturing systems, robot speed and acceleration significantly affect robot's energy consumption. Robot speeds or move times are controllable and this can be used to prepare energy efficient robotic schedules.

Bukata et al. [13] studied finding operation sequences, robot positions, robot's power-saving options and robot's speed in order to reach optimal energy consumption level in a robotic cell including up to 12 industrial robots performing operations such as welding, assembly, etc. For this purpose, they proposed a Mixed Integer Programming (MIP) model. However, as the number of robots in a cell increases, MIP cannot be solved in a reasonable time, so they constructed a heuristic algorithm based on a Linear Programming (LP) solver. They tested proposed methods in a case study from Skoda Auto and showed that up to 20% energy savings can be achieved by only determining robot speed level and standby times. Shortly afterwards, Bukata et al. [14] considered the same problem and same cell environment as in their previous study [13] and presented a new method based on Branch and Bound to solve the problem.

Gürel et al. [36] considered a 2-machine robotic cell scheduling problem with a material handling robot and robot speed decisions. They considered two objectives at the same time: minimizing cycle time and minimizing energy consumption, where energy consumption of the robot is a convex function. They proposed a bicriteria mathematical model and made a trade-off between the two objectives. They showed that reducing the energy consumption is possible by eliminating unnecessary waiting times of the robot.

There are few robotic cell scheduling problems with robot speed control in the literature. However, the studies mentioned in this section show that by controlling robot speed, significant energy saving can be achieved.

### 2.1.3 Studies with controllable machine processing times

In robotic cell scheduling literature, there are few studies that consider processing time control on machines. Gültekin et al. [34] studied scheduling robot moves and determining the processing times on machines for cyclic schedules in 2-machine and 3-machine robotic cells with a material handling robot. Besides cycle time minimization, they focused on a novel objective in the literature; reducing manufacturing costs. They constructed a convex manufacturing cost function composed of machining cost and tooling cost. They proposed a bicriteria optimization model since they considered the trade-off between cycle time and manufacturing costs. Thus, they obtained a set of non-dominated cycles and corresponding processing time levels for the problem. Similarly, Yıldız et al. [69] considered the same objective as Gültekin et al. [34], but in an $m$-machine robotic cell with a material handling robot. They focused on the cycles having flexible machines. They obtained the processing time values which give the minimum cycle time and manufacturing cost for given robot move sequences.

Yan et al. [68] studied finding the processing times on machines producing identical parts for an $m$-machine robotic cell with a material handling robot and a circular layout. They studied finding processing times with an objective of minimizing the cycle time. They proposed a Branch and Bound algorithm. Batur et al. [6] studied a 2-machine robotic cell with multiple part types and single material handling robot where processing times are controllable. They aimed at decreasing the cycle time. They proposed a heuristic algorithm which finds part sequence, processing time levels and robot moves that minimize the cycle time.

We noticed that there are few studies which consider processing time controllability in robotic cells. These studies focus on machines only. However, there is no study considering both robot speed control and processing time control at the same time. Also, these studies aim at mainly cycle time or manufacturing cost minimization. In this thesis, we consider controllable robot speed and machine processing times at the same time.

## 2.2 Robots and Energy Consumption

Although most of the studies in the literature focus on throughput maximization, Grau et al. [33] state that energy efficiency of industrial robots will become a popular research topic in near future. We see that energy consumption of industrial robots receive significant attention in recent studies.

Meike and Ribickis [51] studied ways to reduce energy consumption of industrial robots in automotive industry. They showed that up to 30% of energy can be saved without a negative effect on the throughput rate by changing some mechanical properties. In addition, Meike et al. [50] studied defining energy consumption levels of multi-robot assembly lines in automotive industry for cyclic production. By the help of their optimization model, they carried out a case study with four industrial robots and concluded that synchronized movement of all robots with each other can save about 7% of energy consumption.

Carabin et al. [16] presented a review of existing methods and technologies to improve the energy efficiency of industrial robotic systems. Efforts to reduce robot energy consumption have generally focused on trajectory planning and operation scheduling. By considering many electrical and mechanical properties of the robots, Hansen et al. [39], Pellicciari et al. [56] and Pellegrinelli et al. [55] presented models that find energy consumption optimized trajectories for industrial pick-and-place robots in different settings. Riazi et al. [57] introduced a trajectory optimization procedure decreasing energy consumption of multiple industrial robots in a manufacturing cell. They worked with different cost functions considering acceleration, mechanical jerk and power minimization. They tested their algorithm on real life production cell under different cell conditions, i.e., different robot types and numbers, cycle time levels, etc. As a result of their case study, they reduced the energy consumption of industrial robots by 30% and provided smooth distributed energy consumption without changing the cycle time or trajectory. Also, during their study, they measured energy consumption level of a real life industrial robot and showed that energy consumption as a function of time has convex behaviour.

There are studies that use mathematical models to achieve energy-efficient trajectories. For multiple industrial robots located in a manufacturing cell, Wigstrom et al. [66] addressed a scheduling problem. They proposed a Mixed Integer Nonlinear Programming (MINLP) model which reduces the cost of energy. They used a nonlinear cost function representing total energy consumption in terms of operation times of the robots. In another study, Wigstrom et al. [67] gave an optimization model that produces new trajectories providing lower energy costs for multiple industrial robots in the same manufacturing cell. For their optimization model, they proposed a Mixed Integer Nonlinear Programming model with a convex, nonlinear cost function representing energy consumption. Their model revises an existing trajectory to minimize execution time. Also, their computational results showed that lower energy consumption levels were achieved. Glorieux et al. [32] studied a robotic cell with multiple material handling robots. They proposed a Nonlinear Programming model which finds the optimal trajectory. Proposed model can obtain 14% energy saving in a case study on a robotic cell including 6 material handling robots.

Alatartsev et al. [4] proposed a Traveling Salesman Problem formulation which finds optimal task sequence for execution time minimization problem where flexible tasks exist for an industrial robot. They achieved near optimal task sequences within shorter computation times than the available approaches in the literature. In a robotic assembly line with four assembly robots, Janardhanan et al. [42] studied an assembly line balancing problem by considering two objectives; cycle time and total energy consumption. They proposed a heuristic algorithm based on particle swarm optimization.

In the literature, there are studies which consider the effect of robot speed on energy consumption. Paryanto et al. [12] gave a mechatronic model that estimates the power consumption of an industrial robot. They showed that the parameters that affect robot's power consumption are robot operation speed, payload of the robot and peak power level. Eggers et al. [25] studied finding energy saving levels for an industrial robot considering the travel time of the robot. On the other hand, for industrial robots, Chemnitz et al. [17] showed that decreasing the robot speed is not always the best option to reduce energy consumption. They conducted an experiment on two industrial robots and observed the power measurements. They concluded that energy consumption also depends on the model of the robot. In addition, Kolibal et al. [46] found an energy-optimal speed value of an industrial robot for an operation. These studies point out that the acceleration and deceleration of the robot's motion between two points also affect energy consumption. Gadaleta et al. [30] developed a programming based simulation tool increasing energy efficiency by finding optimum motion parameters such as velocity and acceleration limits for an industrial robot move.

Energy consumption of a robot depends on many factors. There are studies which carried out laboratory tests for certain robot models and measure energy consumption of robots for different speed, acceleration, deceleration profiles. There is no closed form expression for robot energy consumption. Many studies have used nonlinear convex energy consumption functions. Similarly, in this thesis, we use a convex energy consumption function which can be used as an approximation for energy consumption of a real material handling robot.

## 2.3 Machine Scheduling with Processing Time Control

Machine scheduling is a well established research area and processing time control is a well studied concept in machine scheduling problems. Therefore, the surveys summarizing the studies on this subject are very helpful for us to understand the developments in the literature. As an extensive survey, earlier studies on machine scheduling with controllable processing time were summarized by Nowicki and Zdrzałka [53]. This survey mostly included studies on single-machine scheduling problems. Hoogeveen [40] presented a detailed survey of multi-criteria scheduling problems, and in his paper, he especially focused on the scheduling problems with controllable processing times. Moreover, Shabtay and Steiner [60] presented a recent and wide-range survey. Finally, the latest review on this subject was presented by Shioura et al. [61].

A pioneer study in this research area is conducted by Vickson [65]. He found schedules with the lowest cost and completion time on a single machine where processing times are controllable. He represented the processing cost as a linear function. We see that most of the earlier works on this subject were on single-machine scheduling problems (see e.g., [7], [18], [22], [49], [54]). Liman et al. [49] studied single-machine scheduling problems that minimizes earliness, tardiness objectives along with the cost arising from shortening operation times. They also used a linear cost function. Similarly, Biskup et al. [7] constructed a linear objective function for compression penalties of the processing times, earliness and tardiness in single-machine environment. They proposed a mathematical model which can be solved in polynomial time.

Shabtay and Kaspi [59] considered scheduling and resource allocation problem for a single machine used in production. They aimed at minimizing total flow time by weighting according to the importance and workload of the tasks and assumed the processing time function as a nonlinear decreasing function associated with the amount of resources consumed. They provided a dynamic programming algorithm. While they suggested this algorithm for small-scale problems, they emphasized that heuristic approaches can give better results as the size of the problem increases. Considering the same objective, Gürel and Aktürk [35] studied on a bicriteria scheduling problem for a CNC turning machine. They aimed at making a trade-off between total weighted flow time and manufacturing cost. They used a nonlinear function representing manufacturing cost. They presented optimality properties and proposed a heuristic method.

There are also studies which consider processing time control on parallel machines, flow shop and job shop scheduling environments. Nowicki et al. [52] considered a 2-machine flow shop problem to determine the sequence of jobs and processing times on both machines. They used a linear cost function comprised of maximum completion time cost and processing cost. Then, they showed that this problem is NP-hard and proposed a heuristic algorithm to solve the problem. Also, Karabati and Kouvelis [44] studied both selection of optimal processing times and scheduling of multi-product $m$-machine flow-shop problems. Minimizing operation costs were aimed considering throughput rate in the study. First, Karabati and Kouvelis constructed a subproblem with LP formulation minimizing the operating cost in order to determine operation times when operation sequences are predetermined. Then, they proposed an iterative solution procedure for finding optimal processing times and schedules. Also, they stated that as machines with adjustable processing times become prevalent with the development of the technology (i.e. CNC machines), the determination of optimal processing times in machine scheduling problems has become increasingly popular. In another study, Uruk et al. [63] considered both operation assignment and processing time control in a 2-machine flow-shop environment. In their study, they considered flexible operations, i.e., operations which can be performed on any machine which is available at that moment regardless of machine type. They proposed a bicriteria scheduling model trying to minimize both makespan and manufacturing cost.

Controllable processing times have been also practiced in machine rescheduling problems. To illustrate, Aktürk et al. [3] studied on the problem of rescheduling parallel CNC machines. Considering the trade-off between match-up time and production cost simultaneously, they proposed a conic mixed-integer programming model. They used a convex cost function composed of fixed manufacturing cost and cost of expediting jobs. They also proposed a heuristic algorithm for the problem.

In conclusion, it is seen that controllable processing time is a well studied concept in machine scheduling literature. In this thesis, we study controllable processing times and robot speed in a robotic cell environment. We give mathematical models for processing time and speed selection decisions in alternative robot cycles. We show that considering processing time and robot speed decisions together saves energy.

To the best of our knowledge, our work in this thesis is the first study that considers machine processing time and robot speed control with the cycle time and energy consumption objectives in a robotic cell environment.

# CHAPTER 3

# PROBLEM DEFINITION AND SOLUTION PROCEDURE

In this thesis, we consider a two machine robotic cell with a material handling robot. The cell produces identical parts flowing from input buffer to the first machine then to the second machine and finally to output buffer. The cell follows a cyclic schedule, i.e. robot moves and machines' operations are done repetitively to produce the parts. Robot's move times between machines/buffers and machine processing times are controllable, i.e. robot can be expedited and machines can run faster. However, energy consumption of the cell (i.e. robot and machines) depends on the pace of the robot and the machines. Energy consumption of each machine and robot is assumed to be a nonlinear function. Here we study a bicriteria problem to minimize cycle time and energy consumption objectives at the same time.

First, we give the notation used throughout the study below:

## Notation

***Sets and parameters:***

| | |
|---|---|
| $m$ : | number of machines in the cell, $m = 2$. |
| $n$ : | number of parts produced in a cycle. |
| $x$ : | index for the machines in the cell. $0$ and $3$ refer to the input and output buffer, respectively. |
| $l \in \{1, 2, 12\}$ : | index for the cycle type, $S_l$. |
| $(i, j)$ : | moving direction of the robot, from machine $i$ to machine $j$. |
| $h \in \{e, f\}$ : | robot's status: $f$: full, $e$: empty |
| $g$ : | index for robot moves |

$D_l$ : set of robot moves in cycle $S_l$.

$D_l^f, D_l^e$ : sets of full ($f$) and empty ($e$) robot moves in cycle $S_l$.

$d_{ij}$ : distance between machine $i$ and machine $j$ (in meters).

$C_h$ : energy consumption constant for robot when moving in state $h \in \{e, f\}$

$C_{m_x}$ : energy consumption constant for machine $x$.

$\epsilon$ : machine loading and unloading time for robot (in seconds)

***Decision variables:***

$T_{ijh}$: moving time of robot from machine $i$ to $j$ in state $h$. (in seconds)

$p_x$ : processing time of a part on machine $x$ (in seconds)

## 3.1 Energy Consumption Function of the Robot and Machines in the Robotic Cell

We consider a robotic cell where the speed of the robot and the processing times of machines are controllable. In this system, energy is consumed in two ways. First, the robot requires energy for handling operations, in particular, during its move between the machines. Second, the machines require energy for manufacturing operations they do on parts.

We first consider robot's energy consumption. In the robotic cell considered, the robot follows a linear route. We assume that acceleration and deceleration times of the robot and energy consumption during loading and unloading are negligible. Between the machines and buffers, the robot travels at a constant speed. The robot consumes energy during a move. The sum of energy requirements of all movements comprises the total energy consumption of the robot. We use the formula in Gürel et al.'s study [36] and we give the energy consumption of the robot during a move below:

$$F(v) = C \cdot d \cdot v^k \tag{3.1}$$

In this formulation, $C$ and $k$ are constants. $C$ represents the effects of weight and frictional forces and can be different for full and empty moves. $d$ is the distance traveled by the robot. $v$ is the robot's speed during the move. Lastly, $k$ is the exponent which gives the nonlinear relation between the speed of the robot and energy consumption. $C$ and $k$ depends on the type or model of the robot. When the robot moves at a higher speed, it needs more energy. In addition, it is more expensive to increase the speed at higher speeds. So, we assume $k \geq 1$ which also makes energy consumption function convex.

Using $T_{ijh} = \frac{d_{ij}}{v_{ijh}}$, we can express the energy consumption as a function of move time. In this case, as shown in Figure 3.1, the function is a convex, decreasing function as below:

$$F(T) = C \cdot d^{k+1} \cdot T^{-k} \tag{3.2}$$

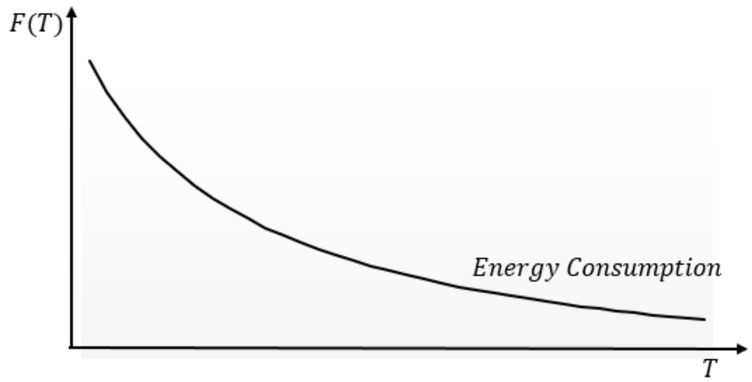


Figure 3.1: Energy consumption with respect to robot moving time

Second, we consider energy consumption of machines. It is correlated with processing times on machines. Expediting jobs on a machine requires energy. As the processing time of a job decreases by changing machining parameters, the machine spends higher energy. An example is the CNC machining where processing times can be decreased by increasing cutting speed and feed rate parameters. In order to formulate the energy consumption of machines, we considered the relationships between manufacturing costs and processing times in the literature, (e.g., [3], [34], [35]). We model energy consumption of a process on a machine using the function below:

$$F(p) = C \cdot p^{-s} \tag{3.3}$$

$C$ and $s$ are constants where $C > 0$, $s \geq 1$, and $p$ is the processing time. $s$ represents the relation between the processing time and energy consumption. $s$ can be different for different processes and machines. For instance, a CNC machine carrying out milling operations can process a part at different milling modes. Depending on the complication of operations, these modes might have different energy requirements. Since $s \geq 1$, function is convex and nonlinear for all $s$ values (see Figure 3.2).

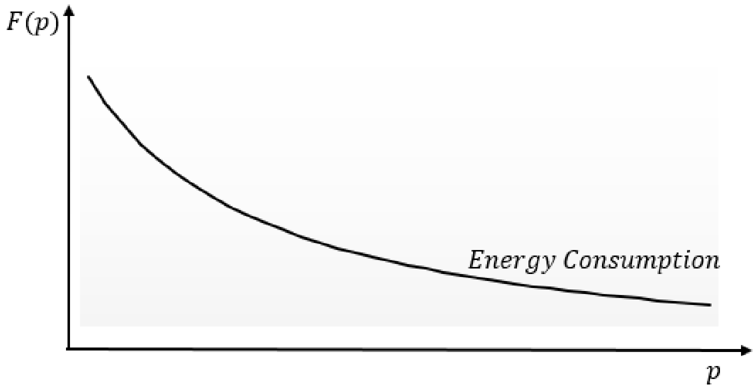


Figure 3.2: Energy consumption with respect to processing time

Since the energy consumption function of all system is sum of the robot's and machines' energy consumption, energy consumption function of the cell is convex, too and it can be expressed as:

$$F(T, p) = \sum_{(i,j,h)} C_h \cdot d_{ij}^{k+1} \cdot T_{ijh}^{-k} + \sum_{x} C_{m_x} \cdot p^{-s} \tag{3.4}$$

Another objective which has been studied in most of the studies in robotic cell scheduling is the cycle time. In the next section, we describe alternative cycles and give cycle time calculations.

## 3.2 Cycle Time of the Robot

The robot performs a number of activities to complete a cycle. Duration of these activities, i.e. robot's travels, loading and unloading actions and waiting times, constitute the cycle time. In an $n$-unit cycle, the cell produces $n$ parts in one cycle. Cycle time expression for $n$-unit cycle is given as in (3.5) where $w_x$ denotes waiting time of the robot in front of machine $x$.

$$\mathcal{CT}^{S_l} = 2n(m+1)\epsilon + \sum_{g \in D_l} T_g + \sum_{x=1}^{m} w_x \tag{3.5}$$

Each machine is loaded and unloaded once, when 1-unit cycle is considered. Also the robot takes a part from input buffer once and loads the output buffer once. Therefore, loading and unloading actions occur $2(m+1)$ times in a cycle. Similarly, total number of loading and unloading actions in an n-unit cycle can be calculated as $2n(m+1)$, since there are $n$ parts in a cycle.

In a cycle, the robot travels between machines and buffers continually. Total time for these moves is given in the second term of equation (3.5).

The robot waiting time ($w_x$), if exists, is the time the robot waits in front of the machine $x$ until the process is over at that machine. If the robot loads a machine and leaves for doing other activities, and return to the machine for unload then waiting may still occur. This is called *partial waiting*. When the robot arrives at a machine which is already loaded and running, the robot may have to wait in front of the machine until the process finishes. If the machine already finished its process when the robot arrives for unloading, then the robot immediately unloads. So, waiting time of the robot is less than or equal to the processing time. Alternatively, if the robot loads a machine and stays until the process finishes, this waiting is called *full waiting*. Waiting time of the robot can be expressed mathematically as in (3.6). $\mu_{lx}$ denotes the set of moves between the loading and unloading of machine $x$ in cycle $S_l$, and $\eta_{lx}$ denotes the number of loading/unloading operations done by the robot in the same interval. If $\mu_{lx} = \emptyset$ and $\eta_{lx} = 0$, then $w_x = P_x$, and it means *full waiting*.

$$w_x = \max\left\{0, P_x - \left(\eta_{lx} \cdot \epsilon + \sum_{g \in \mu_{lx}} T_g\right)\right\}. \tag{3.6}$$

In this thesis, we study three different robot cycles called $S_1$, $S_2$ and $S_{12}$ given by Sethi et al. [58], Hall et al. [37].

### $S_1$ cycle:

First, we define the robot move sequence in $S_1$ cycle which is given in Figure 3.3. In the figure, a dashed line indicates an empty move, i.e. robot moves without a part, whereas a continuous line indicates a full move. In $S_1$ cycle, the robot performs the following operations: take a part ($\epsilon$) from $M_0$, then move to $M_1$ ($T_{01f}$) and drop the part ($\epsilon$), wait in front of $M_1$ during the process ($w_1$), unload the part ($\epsilon$) from $M_1$, take it to $M_2$ ($T_{12f}$) and load ($\epsilon$), wait ($w_2$) machining time in front of $M_2$, finally unload $M_2$ ($\epsilon$) and take the part to output buffer $M_3$ ($T_{23f}$) and load ($\epsilon$). Next, it returns to $M_0$ ($T_{30e}$). Thus, a cycle is completed. Total time required for all these operations gives the cycle time expressed as below:

$$\begin{aligned}\mathcal{CT}^{S_1} &= \epsilon + T_{01f} + \epsilon + w_1 + \epsilon + T_{12f} + \epsilon + w_2 + \epsilon + T_{23f} + \epsilon + T_{30e} \\ &= 6\epsilon + w_1 + w_2 + T_{01f} + T_{12f} + T_{23f} + T_{30e}\end{aligned}$$

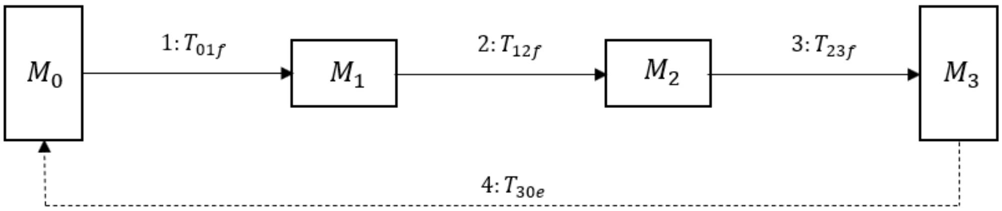


Figure 3.3: Robot activity sequence of $S_1$ cycle

The robot waits in front of $M_1$ and $M_2$ (full waiting), $w_1 = p_1$, $w_2 = p_2$. Then, cycle time expression takes its final form.

$$\mathcal{CT}^{S_1} = 6\epsilon + p_1 + p_2 + T_{01f} + T_{12f} + T_{23f} + T_{30e} \tag{3.7}$$

$S_1$ is a simple and easy to implement cycle.

**$S_2$ cycle:**

In $S_2$ cycle, the robot activity sequence is illustrated in Figure 3.4. At the beginning of $S_2$ cycle, $M_2$ is full and already processing a part. First, the robot takes another part ($\epsilon$) from $M_0$ and takes it to $M_1$ ($T_{01f}$) and loads ($\epsilon$), then $M_1$ starts processing. Meanwhile, the robot moves to $M_2$ ($T_{12e}$) and waits ($w_2$) for the process to be finished on $M_2$, then it takes the processed part ($\epsilon$) and takes it to $M_3$ ($T_{23f}$) and drops ($\epsilon$). Thus, only one incompleted part remains in the cycle. Thereafter, the robot turns back to $M_1$ ($T_{31e}$) and waits until the process is over ($w_1$), then unloads the part ($\epsilon$) and moves to $M_2$ ($T_{12f}$) and drops ($\epsilon$). Lastly, the robot moves to its initial position, i.e. to $M_0$, ($T_{20e}$). Thus, a cycle is over. Sum of the duration of all activities and waiting times give the cycle time:

$$\begin{aligned}\mathcal{CT}^{S_2} &= \epsilon + T_{01f} + \epsilon + T_{12e} + w_2 + \epsilon + T_{23f} + \epsilon + T_{31e} + w_1 + \epsilon + T_{12f} + \epsilon + T_{20e} \\ &= 6\epsilon + T_{01f} + T_{12e} + T_{23f} + T_{31e} + T_{12f} + T_{20e} + w_1 + w_2\end{aligned}$$

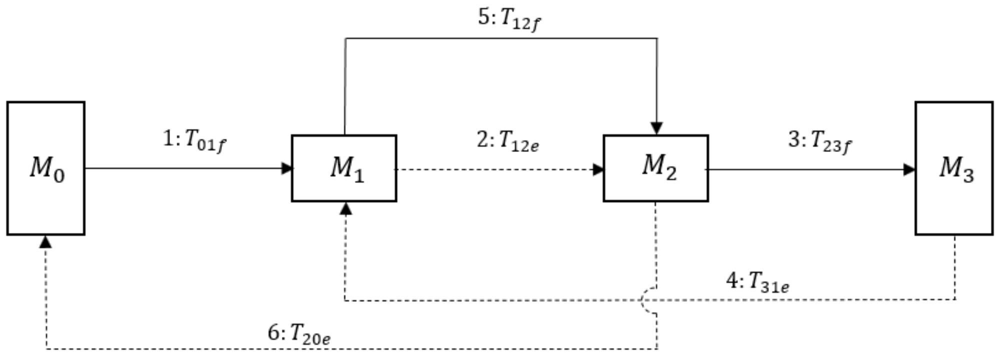


Figure 3.4: Robot activity sequence of $S_2$ cycle

Waiting times occurred by the robot in $S_2$ can be expressed as below:

$$w_1 = \max\{0, p_1 - T_{12e} - w_2 - \epsilon - T_{23f} - \epsilon - T_{31e}\} \quad (3.8)$$

$$w_2 = \max\{0, p_2 - T_{20e} - \epsilon - T_{01f} - \epsilon - T_{12e}\} \quad (3.9)$$

Substituting (3.8) and (3.9) with $w_1$ and $w_2$ respectively, we obtain:

$$\begin{aligned}\mathcal{CT}^{S_2} =& 6\epsilon + T_{01f} + T_{12e} + T_{23f} + T_{31e} + T_{12f} + T_{20e} + \max\{0, \\ & p_1 - T_{12e} - \epsilon - T_{23f} - \epsilon - T_{31e}, p_2 - T_{20e} - \epsilon - T_{01f} - \epsilon - T_{12e}\}\end{aligned} \quad (3.10)$$

$S_1$ and $S_2$ are the two possible 1-unit cycles in a 2-machine robotic cell. Next, we describe a 2-unit cycle.

### $S_{12}$ cycle:

In $S_{12}$ cycle, two parts are produced in one cycle. For this reason, this cycle has more robot activities and machine operations than $S_1$ (and $S_2$) has. Similar to $S_1$ and $S_2$, the robot starts the cycle in front of $M_0$. Figure 3.5 gives the robot activity sequence in $S_{12}$ cycle. Different than Figure 3.3 and 3.4, in Figure 3.5, lines labeled as $T^{(1)}_{ijf}$

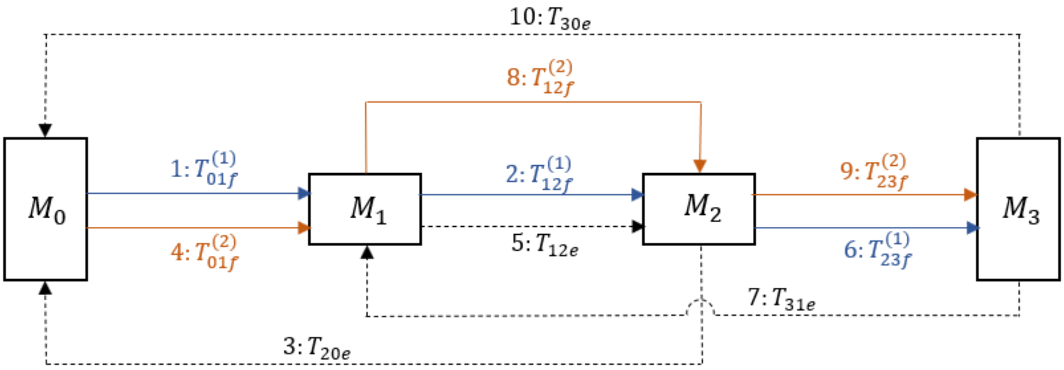


Figure 3.5: Robot activity sequence of $S_{12}$ cycle

show the robot moves for the first part, and lines labeled as $T_{ijf}^{(2)}$ show the moves for the second part.

Robot carries out the following operations in this cycle. Initially, both two machines are idle. Robot takes the first part from $M_0$ ($\epsilon$). It carries the part to $M_1$ ($T_{01f}^{(1)}$) and loads ($\epsilon$). It waits ($p_{11}$) until processing is over on $M_1$. Then, it unloads the part ($\epsilon$), takes it to $M_2$ ($T_{12f}^{(1)}$) and loads ($\epsilon$). The robot does not wait, comes back to $M_0$ ($T_{20e}$). It takes a new part ($\epsilon$), moves to $M_1$ ($T_{01f}^{(2)}$) and loads ($\epsilon$). Next, it moves to $M_2$ ($T_{12e}$), waits until the process is over ($w_2$). Then, it takes the first part ($\epsilon$), moves to $M_3$ ($T_{23f}^{(1)}$) and drops the part ($\epsilon$). Thus, first part is finished. Then, the robot moves to $M_1$ ($T_{31e}$). If the process on $M_1$ is not over, robot waits ($w_1$), then unloads the second part ($\epsilon$), and takes it to $M_2$ ($T_{12f}^{(2)}$) and loads ($\epsilon$). Then, it waits until the operation finishes ($p_{22}$), takes the part ($\epsilon$) and goes to $M_3$ ($T_{23f}^{(2)}$). Then, drops the second part ($\epsilon$). Finally, the robot returns to $M_0$ ($T_{30e}$), i.e. its initial position. Cycle $S_{12}$ is completed. Accordingly, mentioned steps create the cycle time.

$$\begin{aligned} \mathcal{CT}^{S_{12}} &= \epsilon + T_{01f}^{(1)} + \epsilon + p_{11} + \epsilon + T_{12f}^{(1)} + \epsilon + T_{20e} + \epsilon + T_{01f}^{(2)} + \epsilon + T_{12e} + w_2 + \epsilon \\ &+ T_{23f}^{(1)} + \epsilon + T_{31e} + w_1 + \epsilon + T_{12f}^{(2)} + \epsilon + p_{22} + \epsilon + T_{23f}^{(2)} + \epsilon + T_{30e} \end{aligned}$$

$$\begin{aligned} \mathcal{CT}^{S_{12}} &= 12\epsilon + p_{11} + p_{22} + T_{01f}^{(1)} + T_{12f}^{(1)} + T_{20e} + T_{01f}^{(2)} + T_{12e} + T_{23f}^{(1)} + T_{31e} + T_{12f}^{(2)} \\ &+ T_{23f}^{(2)} + T_{30e} + w_2 + w_1 \end{aligned}$$

In the formulation, $w_2$ denotes waiting time for the first part in front of $M_2$ and $w_1$ denotes waiting time for the second part in front of $M_1$. Partial waiting times occurred in $S_{12}$ can be expressed as below:

$$w_2 = max\{0, p_{12} - T_{20e} - \epsilon - T^{(2)}_{01f} - \epsilon - T_{12e}\} \tag{3.11}$$

$$w_1 = max\{0, p_{21} - T_{12e} - w_2 - \epsilon - T^{(1)}_{23f} - \epsilon - T_{31e}\} \tag{3.12}$$

Then, cycle time expression converts to a new form due to waiting time expressions (3.11) and (3.12).

$$\begin{aligned} \mathcal{CT}^{S_{12}} &= 12\epsilon + p_{11} + p_{22} + T^{(1)}_{01f} + T^{(1)}_{12f} + T_{20e} + T^{(2)}_{01f} \\ &+ T_{12e} + T^{(1)}_{23f} + T_{31e} + T^{(2)}_{12f} + T^{(2)}_{23f} + T_{30e} \\ &+ \max\{0, p_{12} - 2\epsilon - T_{20e} - T^{(2)}_{01f} - T_{12e}, p_{21} - 2\epsilon - T_{12e} - T^{(1)}_{23f} - T_{31e}\} \end{aligned} \tag{3.13}$$

In the next section, we will give mathematical models and analysis for all cycles.

### 3.3 Mathematical Models and Analysis

In this thesis, we study the robotic cell scheduling problem where we want to find robot cycle, robot move times and machines' processing times that minimize cycle time and total energy consumption per part produced.

For the given cell, we formulate the problem as follows:

$$\min \ \mathcal{F} : \ \mathcal{F}^{S_l}$$

$$\min \ \mathcal{CT} : \ \mathcal{CT}^{S_l}$$

s.t. SELECT ONE CYCLE $l \in L$

$$\mathcal{F}^{S_1} = C_f \cdot \sum_{g \in D^f_l} d^{k+1}_g \cdot T^{-k}_g + C_e \cdot \sum_{g \in D^e_l} d^{k+1}_g \cdot T^{-k}_g + \sum_{x=1}^{2} C_{m_x} \cdot {p_x}^{-s} \quad \forall l \tag{3.14}$$

$$\mathcal{CT}^{S_l} = 2(m+1)\epsilon + \sum_{g \in D_l} T_g + \sum_{x=1}^{2} w_x \quad \forall l \tag{3.15}$$

$$T^{LB} \leq T_g \leq T^{UB} \quad \forall g \in D_l, \forall l \tag{3.16}$$

$$p^{LB} \leq p_x \leq p^{UB} \quad \forall x = 1, 2 \tag{3.17}$$

When a robot move time increases, robot's energy consumption decreases. However, in such a case, the cycle time may increase. Similarly, if processing time on a machine is decreased, then cycle time may decrease and energy consumption will increase. In short, cycle time and energy consumption objectives conflicts with each other. We need to find efficient solutions (Pareto optimal) for the problem.

In the literature, there are different methods used for these type of problem. These methods accept different assumptions. For example, some methods are based on a composite objective function including both goals, while there is also another method evaluating the two goals by ranking them in order of importance. In this study, we make use of $\epsilon$-constraint approach which finds efficient solutions for a bicriteria problem. The method sets one of the goals as an objective function and tries to minimize it. The other objective is handled as a constraint in the form of inequality. In this method, an upper bound is determined for the constrainted objective. The upper bounds are changeable and the effect of the change in the upper bounds on the solution can be easily analyzed.

Given a certain demand level for produced parts, the decision maker can change the cycle time level that would meet the demand. Given this cycle time level, the decision maker would design a cycle that minimizes the robotic cell's energy consumption. Hence, we keep energy consumption function in the objective and move cycle time objective to the constraints. We impose the target cycle time level ($\mathcal{K}$) as an upper bound on cycle time objective. Thus, for a given cycle $S_l$ we obtain the following mathematical model:

$$Min \quad C_f \cdot \sum_{g \in D_l^f} d_g^{k+1} \cdot T_g^{-k} + C_e \cdot \sum_{g \in D_l^e} d_g^{k+1} \cdot T_g^{-k} + \sum_{x=1}^{2} C_{m_x} \cdot {p_x}^{-s} \qquad (3.18)$$

s.t.

$$2(m+1)\epsilon + \sum_{g \in D_l} T_g + \sum_{x=1}^{2} w_x \leq \mathcal{K} \qquad (3.19)$$

$$T^{LB} \leq T_g \leq T^{UB} \qquad \forall g \in D_l \qquad (3.20)$$

$$p^{LB} \leq p_x \leq p^{UB} \qquad \forall x \in \{1,2\} \qquad (3.21)$$

Cycle specific mathematical models are presented in next sections.

### 3.3.1 $S_1$ Cycle:

For $S_1$ cycle, mathematical formulation is given below:

$$Min \quad C_f \cdot \left[d_{01f}^{k+1} \cdot T_{01f}^{-k} + d_{12f}^{k+1} \cdot T_{12f}^{-k} + d_{23f}^{k+1} \cdot T_{23f}^{-k}\right] + C_e \cdot d_{30e}^{k+1} \cdot T_{30e}^{-k} + C_{m_1} \cdot {p_1}^{-s} + C_{m_2} \cdot {p_2}^{-s} \tag{3.22}$$

$$\text{s.t.} \quad 6\epsilon + p_1 + p_2 + T_{01f} + T_{12f} + T_{23f} + T_{30e} \leq \mathcal{K} \tag{3.23}$$

$$0 \leq T_{01f}, T_{12f}, T_{23f}, T_{30e} \leq T^{UB} \tag{3.24}$$

$$0 \leq p_1, p_2 \leq p^{UB} \tag{3.25}$$

The objective function (3.22) consists of terms $C \cdot d^{k+1} \cdot T^{-k}$ and $C \cdot p^{-s}$. Since $k \geq 1$ and $s \geq 1$, each term is convex and objective function is convex. All constraints are linear, and they define a convex set. Therefore, the problem is a convex optimization problem.

Constraint (3.23) is the cycle time constraint. Constraints (3.24) and (3.25) guarantee that robot move time and processing times are within their bounds. Bounds of travel time might vary depending on the distance traveled in a move. However, processing time bounds are same for all machines.

We assume that lower bounds of all decision variables are zero. Also, we assume that $\mathcal{K} > 6\epsilon$ holds, otherwise, we cannot achieve a feasible solution for $S_1$ cycle. It is a Nonlinear Programming Problem (NLP). In order to achieve optimality properties, we carry out Karush-Kuhn-Tucker (KKT) analysis.

If we omit upper bound constraints in (3.24) and (3.25), then Lagrange function is expressed as below:

$$\begin{aligned} L(T_{ijh}, p_x, \mu) = C_f \cdot & \left[d_{01f}^{k+1} \cdot T_{01f}^{-k} + d_{12f}^{k+1} \cdot T_{12f}^{-k} + d_{23f}^{k+1} \cdot T_{23f}^{-k}\right] + C_e \cdot d_{30e}^{k+1} \cdot T_{30e}^{-k} \\ & + C_{m_1} \cdot {p_1}^{-s} + C_{m_2} \cdot {p_2}^{-s} \\ & + \mu \cdot [\mathcal{K} - 6\epsilon - p_1 - p_2 - T_{01f} - T_{12f} - T_{23f} - T_{30e}] \end{aligned}$$

Where $\mu$ is the Lagrangian dual for constraint 3.23, it is known that in an optimal solution partial derivatives of $L(T_{ijh}, p_x, \mu)$ with respect to decision variables should be equal to zero i.e. $\frac{\partial L}{\partial T_{ijh}} = 0, \forall ijh$ and $\frac{\partial L}{\partial p_x} = 0, \forall x$. Then, gradient equations given below must hold at optimality.

*Gradient Equations:*

$$\begin{aligned}
\frac{\partial L}{\partial T_{01f}} &= - k \cdot C_f \cdot d_{01f}^{k+1} \cdot T_{01f}^{-k-1} - \mu = 0 \\
\frac{\partial L}{\partial T_{12f}} &= - k \cdot C_f \cdot d_{12f}^{k+1} \cdot T_{12f}^{-k-1} - \mu = 0 \\
\frac{\partial L}{\partial T_{23f}} &= - k \cdot C_f \cdot d_{23f}^{k+1} \cdot T_{23f}^{-k-1} - \mu = 0 \\
\frac{\partial L}{\partial T_{30e}} &= - k \cdot C_f \cdot d_{30e}^{k+1} \cdot T_{30e}^{-k-1} - \mu = 0 \\
\frac{\partial L}{\partial p_1} &= - s \cdot C_{m_1} \cdot p_1^{-s-1} - \mu = 0 \\
\frac{\partial L}{\partial p_2} &= - s \cdot C_{m_2} \cdot p_2^{-s-1} - \mu = 0
\end{aligned}$$

Other KKT conditions are given below.

*Complementary Slackness:*

$$\mu \cdot [\mathcal{K} - 6\epsilon - p_1 - p_2 - T_{01f} - T_{12f} - T_{23f} - T_{30e}] = 0$$

*Primal feasibility:*

$$\begin{aligned}
&6\epsilon + p_1 + p_2 + T_{01f} + T_{12f} + T_{23f} + T_{30e} \leq \mathcal{K} \\
&T_{01f}, T_{12f}, T_{23f}, T_{30e}, p_1, p_2 \geq 0
\end{aligned}$$

*Lagrange multipliers sign restrictions:* $\mu \leq 0$

Since this problem is a convex optimization problem as we discussed before, any local minimum is also a global minimum.

In our thesis, we assume lower bounds of robot handling time and processing time are zero. However, we know that robot travel and processing times can never be zero, since zero value of traveling and processing times does not express a meaningful solution. Therefore, dual variables for constraint (3.24) and (3.25) must be zero as the upper bounds are assumed to be sufficiently large and lower bounds are never achievable. Then, Proposition 1 states that cycle time constraint of the mathematical model will be tight at optimality.

**Proposition 1.** *Suppose that $\mathcal{K}$ is finite and $T_{ijh}^{UB}$ and $p_x^{UB}$ values are sufficiently large, then cycle time constraint (3.23) is always binding in an optimal solution.*

*Proof.* If we use the gradient equations of $S_1$ cycle problem, we can rewrite the equations as follows.

$$-k \cdot C_h \cdot d_{ijh}^{k+1} \cdot T_{ijh}^{-k-1} = \mu \tag{3.26}$$

$$-s \cdot C_{m_x} \cdot p_x^{-s-1} = \mu \tag{3.27}$$

Because $k \geq 1, s \geq 1$ and all remaining parameters can have only positive values, left hand side of the equations (3.26) and (3.27) are strictly lower than zero. Hence, $\mu$ cannot be zero. Then, due to the complementary slackness, expression (3.23) must be always tight at optimality. □

Proposition 1 implies that at an efficient solution increasing $T_{ijh}$ or $p_x$ to improve energy objective will always increase cycle time.

We next consider the case where $C_{m_1} = C_{m_2}$ and show that processing times at $M_1$ and $M_2$ must be equal at optimality.

**Proposition 2.** *If two machines have the same energy consumption function, processing times must be equal at optimality.*

*Proof.* $T_{ijh}$ and $p_x$ variables can be expressed by using equations (3.26) and (3.27).

$$T_{ijh} = \sqrt[-k-1]{\frac{\mu}{-k \cdot C_h \cdot d_{ijh}^{k+1}}} \qquad \forall (ijh) \in D_l \tag{3.28}$$

$$p_x = \sqrt[-s-1]{\frac{\mu}{-s \cdot C_{m_x}}} \qquad where \quad x = 1, 2 \tag{3.29}$$

Also, it is known that the cycle time constraint is tight at optimality. Then;

$$\sum_{(ijh)} \sqrt[-k-1]{\frac{\mu}{-k \cdot C_h \cdot d_{ijh}^{k+1}}} = \mathcal{K} - 6\epsilon - p_1 - p_2 \tag{3.30}$$

$$\mu = \left( \frac{\sum_{(ijh)} (d_{ijh} \cdot \sqrt[k+1]{-k \cdot C_h})}{\mathcal{K} - 6\epsilon - p_1 - p_2} \right)^{k+1} \tag{3.31}$$

If we replace $\mu$ value with (3.31) in equation (3.28); then we get

$$T_{ijh} = \frac{[\sqrt[k+1]{-k \cdot C_h} \cdot d_{ijh}] \cdot [\mathcal{K} - 6\epsilon - p_1 - p_2]}{\sum_{(ijh)}(d_{ijh} \cdot \sqrt[k+1]{-k \cdot C_{h_1}})} \tag{3.32}$$

Using (3.29) and (3.31) we get:

$$p_x = \left( \left( \frac{\sum_{(ijh)} d_{ijh} \cdot \sqrt[k+1]{-k \cdot C_{h_1}}}{\sum_{(ijh)} T_{ijh}} \right)^{k+1} \cdot \frac{1}{-s \cdot C_{m_x}} \right)^{\frac{1}{-s-1}} \tag{3.33}$$

$C_{m_x}$ and $s$ depend on machine and process characteristics. Remaining parameters are the same for both machines in the robotic cell. Therefore, energy consumption functions are same and optimal processing time values will be equal on both machines. □

**Proposition 3.** *In an optimal solution, marginal cost of decreasing travel time of the robot is equal to the marginal cost of decreasing the processing time.*

*Proof.* This statement can be proved by comparing partial derivatives of the objective function with respect to robot's move time and machine processing time at an optimal solution.

When partial derivatives of the objective function with respect to $T_{ijh}$ and $p_x$ are derived, marginal costs are equal to each other due to KKT conditions.

$$\frac{\partial f}{\partial T_{ijh}} = -k \cdot C_h \cdot T_{ijh}^{-k-1} \cdot d_{ijh}^{k+1} = \mu$$
$$\frac{\partial f}{\partial p_x} = -s \cdot C_{m_x} \cdot p_x^{-s-1} = \mu$$

□

Proposition 3 shows that at an optimal solution robot move times and processing times are chosen in such a way that it is not possible to improve energy consumption by increasing or decreasing robot move times and processing times.

### 3.3.2 $S_2$ Cycle:

Now, we give the mathematical model for $S_2$ cycle. First, we obtain cycle time constraints by using equation (3.10). We obtain these constraints by linearizing the max term in equation (3.10).

$$
\begin{aligned}
&6\epsilon + T_{01f} + T_{12e} + T_{23f} + T_{31e} + T_{12f} + T_{20e} \leq \mathcal{K} \\
&4\epsilon + p_1 + T_{01f} + T_{12f} + T_{20e} \leq \mathcal{K} \\
&4\epsilon + p_2 + T_{23f} + T_{31e} + T_{12f} \leq \mathcal{K}
\end{aligned}
$$

Then, mathematical model of $S_2$ cycle is as follows:

$$
\begin{aligned}
Min \quad & C_f \cdot \left[d_{01f}^{k+1} \cdot T_{01f}^{-k} + d_{12f}^{k+1} \cdot T_{12f}^{-k} + d_{23f}^{k+1} \cdot T_{23f}^{-k}\right] + \qquad (3.34)\\
& C_e \cdot \left[d_{12e}^{k+1} \cdot T_{12e}^{-k} + d_{31e}^{k+1} \cdot T_{31e}^{-k} + d_{20e}^{k+1} \cdot T_{20e}^{-k}\right] + C_{m_1} \cdot p_1^{-s} + C_{m_2} \cdot p_2^{-s}
\end{aligned}
$$

$$
\begin{aligned}
\text{s.t.} \quad 6\epsilon + T_{01f} + T_{12e} + T_{23f} + T_{31e} + T_{12f} + T_{20e} &\leq \mathcal{K} && (3.35)\\
4\epsilon + p_1 + T_{01f} + T_{12f} + T_{20e} &\leq \mathcal{K} && (3.36)\\
4\epsilon + p_2 + T_{23f} + T_{31e} + T_{12f} &\leq \mathcal{K} && (3.37)\\
0 \leq T_{01f}, T_{12f}, T_{23f}, T_{12e}, T_{31e}, T_{20e} &\leq T^{UB} && (3.38)\\
0 \leq p_1, p_2 &\leq p^{UB} && (3.39)
\end{aligned}
$$

Similar to the model given for $S_1$ cycle, this model is also a convex optimization problem. We write KKT conditions, and we deduce some optimality properties for the problem.

First, we give the Lagrangian function.

$$
\begin{aligned}
L(T_{ijh}, p_x, \mu) = {} & C_f \cdot \left[d_{01f}^{k+1} \cdot T_{01f}^{-k} + d_{12f}^{k+1} \cdot T_{12f}^{-k} + d_{23f}^{k+1} \cdot T_{23f}^{-k}\right] \\
& + C_e \cdot \left[d_{12e}^{k+1} \cdot T_{12e}^{-k} + d_{31e}^{k+1} \cdot T_{31e}^{-k} + d_{20e}^{k+1} \cdot T_{20e}^{-k}\right] + C_{m_1} \cdot p_1^{-s} + C_{m_2} \cdot p_2^{-s} \\
& + \mu_1 \cdot [\mathcal{K} - 6\epsilon - T_{01f} - T_{12e} - T_{23f} - T_{31e} - T_{12f} - T_{20e}] \\
& + \mu_2 \cdot [\mathcal{K} - 4\epsilon - p_1 - T_{01f} - T_{12f} - T_{20e}] \\
& + \mu_3 \cdot [\mathcal{K} - 4\epsilon - p_2 - T_{23f} - T_{31e} - T_{12f}]
\end{aligned}
$$

$\mu_1$, $\mu_2$ and $\mu_3$ are the Lagrange multipliers for the constraints (3.35), (3.36) and (3.37), respectively. Partial derivatives of the Lagrangian function with respect to decision variables must be zero in an optimal solution. Then, following gradient equations must hold.

*Gradient Equations:*

$$\begin{aligned}
\frac{\partial L}{\partial T_{01f}} &= -k \cdot C_f \cdot d_{01f}^{k+1} \cdot T_{01f}^{-k-1} - \mu_1 - \mu_2 = 0 \\
\frac{\partial L}{\partial T_{12f}} &= -k \cdot C_f \cdot d_{12f}^{k+1} \cdot T_{12f}^{-k-1} - \mu_1 - \mu_2 - \mu_3 = 0 \\
\frac{\partial L}{\partial T_{23f}} &= -k \cdot C_f \cdot d_{23f}^{k+1} \cdot T_{23f}^{-k-1} - \mu_1 - \mu_3 = 0 \\
\frac{\partial L}{\partial T_{12e}} &= -k \cdot C_e \cdot d_{12e}^{k+1} \cdot T_{12e}^{-k-1} - \mu_1 = 0 \\
\frac{\partial L}{\partial T_{31e}} &= -k \cdot C_e \cdot d_{31e}^{k+1} \cdot T_{31e}^{-k-1} - \mu_1 - \mu_3 = 0 \\
\frac{\partial L}{\partial T_{20e}} &= -k \cdot C_e \cdot d_{20e}^{k+1} \cdot T_{20e}^{-k-1} - \mu_1 - \mu_2 = 0 \\
\frac{\partial L}{\partial p_1} &= -s \cdot C_{m_1} \cdot p_1^{-s-1} - \mu_2 = 0 \\
\frac{\partial L}{\partial p_2} &= -s \cdot C_{m_2} \cdot p_2^{-s-1} - \mu_3 = 0
\end{aligned}$$

We also write the other KKT conditions as below:

*Complementary Slackness:*

$$\begin{aligned}
&\mu_1 \cdot [\mathcal{K} - 6\epsilon - T_{01f} - T_{12e} - T_{23f} - T_{31e} - T_{12f} - T_{20e}] = 0 \\
&\mu_2 \cdot [\mathcal{K} - 4\epsilon - p_1 - T_{01f} - T_{12f} - T_{20e}] = 0 \\
&\mu_3 \cdot [\mathcal{K} - 4\epsilon - p_2 - T_{23f} - T_{31e} - T_{12f}] = 0
\end{aligned}$$

*Primal Feasibility:*

$$\begin{aligned}
6\epsilon + T_{01f} + T_{12e} + T_{23f} + T_{31e} + T_{12f} + T_{20e} &\leq \mathcal{K} \\
4\epsilon + p_1 + T_{01f} + T_{12f} + T_{20e} &\leq \mathcal{K} \\
4\epsilon + p_2 + T_{23f} + T_{31e} + T_{12f} &\leq \mathcal{K} \\
T_{01f}, T_{12f}, T_{23f}, T_{12e}, T_{31e}, T_{20e}, p_1, p_2 &\geq 0
\end{aligned}$$

*Lagrange multipliers sign restrictions:* $\mu_y \leq 0 \qquad \forall y = 1, 2, 3$

The following proposition shows that all cycle time constraints of mathematical model are always tight.

**Proposition 4.** *Suppose that $\mathcal{K}$ is finite and $T_{ijh}^{UB}$ and $p_x^{UB}$ are sufficiently large, then cycle time constraints of $S_2$ cycle, constraints (3.35), (3.36) and (3.37), are always binding in an optimal solution.*

*Proof.* Consider the following gradient equations;

$$-k \cdot C_e \cdot d_{12e}^{k+1} \cdot T_{12e}^{-k-1} = \mu_1 \quad (3.40)$$

$$-s \cdot C_{m_1} \cdot p_1^{s-1} = \mu_2 \quad (3.41)$$

$$-s \cdot C_{m_2} \cdot p_2^{s-1} = \mu_3 \quad (3.42)$$

As explained in proof of Proposition 1, $\mu_1$, $\mu_2$ and $\mu_3$ cannot be zero. Due to complementary slackness (3.35), (3.36) and (3.37) are always binding in an optimal solution. □

Proposition 4 shows that at optimality, cycle time of the schedule is always equal to $\mathcal{K}$, i.e. robot move times or processing times cannot be increased further to improve energy consumption.

In robotic cell schedules, waiting times can exist, i.e., the robot can wait for the machine or vice versa. Next, we show that, in $S_2$ cycle, when energy consumption is minimized for a given cycle time, neither the robot waits for unloading a full machine nor a full machine waits robot after finishing its operation on a part.

**Proposition 5.** *In $S_2$ cycle, if $\mathcal{K}$ is sufficiently large, at optimality, part completion time on a machine and robot arrival time at the machine are synchronized i.e. both robot's and machine's waiting times are zero.*

*Proof.* We have already given robot's waiting times in (3.8) and (3.9):

$$w_1 = \max(0, p_1 - 2\epsilon - T_{12e} - T_{23f} - T_{31e} - w_2)$$
$$w_2 = \max(0, p_2 - 2\epsilon - T_{20e} - T_{01f} - T_{12e})$$

As given in Proposition 4, constraints (3.35), (3.36) and (3.37) are tight in the optimal solution. Then, $p_1$ and $p_2$ can be expressed as:

$$p_1 = 2\epsilon + T_{12e} + T_{23f} + T_{31e} \tag{3.43}$$

$$p_2 = 2\epsilon + T_{01f} + T_{12e} + T_{20e} \tag{3.44}$$

If we reconsider the robot's waiting time equations by utilizing (3.43) and (3.44), it is concluded that $w_1 = 0$ and $w_2 = 0$. Thus, robot's waiting times are eliminated in an optimal solution.

Let $w_{M_1}$ and $w_{M_2}$ denote the machines' waiting times for the robot. Their mathematical expressions can be constructed as;

$$w_{M_1} = \max(0, T_{12e} + \epsilon + T_{23f} + w_2 + \epsilon + T_{31e} - p_1) \tag{3.45}$$

$$w_{M_2} = \max(0, T_{20e} + \epsilon + T_{01f} + \epsilon + T_{12e} - p_2) \tag{3.46}$$

Similarly, if we use equations (3.43) and (3.44), we conclude that $w_{M_1} = 0$ and $w_{M_2} = 0$. It means that the robot arrives at a machine just when the machine completes a part. □

In $S_2$ cycle, robot speed control and machine processing time control strategies eliminate waiting times by slowing down the operations and hence save energy.

### 3.3.3 $S_{12}$ Cycle:

So far, we have examined 1-unit cycles. In this section, we will present our analysis on a 2-unit cycle called $S_{12}$. This is the only 2-unit cycle that is shown in the literature.

Now, we give the mathematical model for $S_{12}$ cycle. First, we construct cycle time constraints. We linearize the max term in the cycle time expression given in equation (3.13). Then, we obtain three different constraints.

$$12\epsilon + p_{11} + p_{22} + T^{(1)}_{01f} + T^{(1)}_{12f} + T_{20e} + T^{(2)}_{01f} + T_{12e} + T^{(1)}_{23f} + T_{31e} + T^{(2)}_{12f} + T^{(2)}_{23f} + T_{30e} \leq \mathcal{K}$$

$$10\epsilon + p_{11} + p_{22} + p_{12} + T^{(1)}_{01f} + T^{(1)}_{12f} + T^{(1)}_{23f} + T_{31e} + T^{(2)}_{12f} + T^{(2)}_{23f} + T_{30e} \leq \mathcal{K}$$

$$10\epsilon + p_{11} + p_{22} + p_{21} + T^{(1)}_{01f} + T^{(1)}_{12f} + T_{20e} + T^{(2)}_{01f} + T^{(2)}_{12f} + T^{(2)}_{23f} + T_{30e} \leq \mathcal{K}$$

Then, mathematical formulation for $S_{12}$ cycle is given below:

$$\begin{aligned} Min \quad & C_f.\Big[d^{k+1}_{01f} \cdot {T^{(1)}_{01f}}^{-k} + d^{k+1}_{12f} \cdot {T^{(1)}_{12f}}^{-k} + d^{k+1}_{01f} \cdot {T^{(2)}_{01f}}^{-k} + \\ & \quad d^{k+1}_{23f} \cdot {T^{(1)}_{23f}}^{-k} + d^{k+1}_{12f} \cdot {T^{(2)}_{12f}}^{-k} + d^{k+1}_{23f} \cdot {T^{(2)}_{23f}}^{-k}\Big] + \\ & C_e.\Big[d^{k+1}_{20e} \cdot T^{-k}_{20e} + d^{k+1}_{12e} \cdot T^{-k}_{12e} + d^{k+1}_{31e} \cdot T^{-k}_{31e} + d^{k+1}_{30e} \cdot T^{-k}_{30e}\Big] + \\ & C_{m_1} \cdot \big[p^{-s}_{11} + p^{-s}_{21}\big] + C_{m_2} \cdot \big[p^{-s}_{12} + p^{-s}_{22}\big] \end{aligned} \tag{3.47}$$

s.t.

$$12\epsilon + p_{11} + p_{22} + T^{(1)}_{01f} + T^{(1)}_{12f} + T_{20e} + T^{(2)}_{01f} + T_{12e} + T^{(1)}_{23f} + T_{31e} + T^{(2)}_{12f} + T^{(2)}_{23f} + T_{30e} \leq \mathcal{K} \tag{3.48}$$

$$10\epsilon + p_{11} + p_{22} + p_{12} + T^{(1)}_{01f} + T^{(1)}_{12f} + T^{(1)}_{23f} + T_{31e} + T^{(2)}_{12f} + T^{(2)}_{23f} + T_{30e} \leq \mathcal{K} \tag{3.49}$$

$$10\epsilon + p_{11} + p_{22} + p_{21} + T^{(1)}_{01f} + T^{(1)}_{12f} + T_{20e} + T^{(2)}_{01f} + T^{(2)}_{12f} + T^{(2)}_{23f} + T_{30e} \leq \mathcal{K} \tag{3.50}$$

$$0 \leq T^{(1)}_{01f}, T^{(1)}_{12f}, T_{20e}, T^{(2)}_{01f}, T_{12e}, T^{(1)}_{23f}, T_{31e}, T^{(2)}_{12f}, T^{(2)}_{23f}, T_{30e} \leq T^{UB} \tag{3.51}$$

$$0 \leq p_{11}, p_{12}, p_{21}, p_{22} \leq p^{UB} \tag{3.52}$$

We first derive KKT conditions for the model.

Lagrangian function of this model is given below:

$$
\begin{aligned}
L(T_{ijh}, p_x, \mu) = C_f \cdot \Bigg[ & d_{01f}^{k+1} \cdot {T_{01f}^{(1)}}^{-k} + d_{12f}^{k+1} \cdot {T_{12f}^{(1)}}^{-k} + d_{01f}^{k+1} \cdot {T_{01f}^{(2)}}^{-k} + \\
& d_{23f}^{k+1} \cdot {T_{23f}^{(1)}}^{-k} + d_{12f}^{k+1} \cdot {T_{12f}^{(2)}}^{-k} + d_{23f}^{k+1} \cdot {T_{23f}^{(2)}}^{-k} \Bigg] + \\
& + C_e \cdot \left[ d_{20e}^{k+1} \cdot T_{20e}^{-k} + d_{12e}^{k+1} \cdot T_{12e}^{-k} + d_{31e}^{k+1} \cdot T_{31e}^{-k} + d_{30e}^{k+1} \cdot T_{30e}^{-k} \right] \\
& + C_{m_1} \cdot \left[ p_{11}^{-s} + p_{21}^{-s} \right] + C_{m_2} \cdot \left[ p_{12}^{-s} + p_{22}^{-s} \right] \\
& + \mu_1 \cdot \Bigg[ \mathcal{K} - 12\epsilon - p_{11} - p_{22} - T_{01f}^{(1)} - T_{12f}^{(1)} - T_{20e} - T_{01f}^{(2)} - T_{12e} - T_{23f}^{(1)} \\
& \quad - T_{31e} - T_{12f}^{(2)} - T_{23f}^{(2)} - T_{30e} \Bigg] \\
& + \mu_2 \cdot \Bigg[ \mathcal{K} - 10\epsilon - p_{11} - p_{22} - p_{12} - T_{01f}^{(1)} - T_{12f}^{(1)} - T_{23f}^{(1)} - T_{31e} - T_{12f}^{(2)} \\
& \quad - T_{23f}^{(2)} - T_{30e} \Bigg] \\
& + \mu_3 \cdot \Bigg[ \mathcal{K} - 10\epsilon - p_{11} - p_{22} - p_{21} - T_{01f}^{(1)} - T_{12f}^{(1)} - T_{20e} - T_{01f}^{(2)} - T_{12f}^{(2)} \\
& \quad - T_{23f}^{(2)} - T_{30e} \Bigg]
\end{aligned}
$$

Then, following KKT conditions must hold, where $\mu_1$, $\mu_2$ and $\mu_3$ are the Lagrange multipliers for constrains (3.48), (3.49) and (3.50), respectively.

*Gradient Equations:*

$$
\begin{aligned}
\frac{\partial L}{\partial T_{01f}^{(1)}} &= -k.C_f \cdot d_{01f}^{k+1} \cdot {T_{01f}^{(1)}}^{-k-1} - \mu_1 - \mu_2 - \mu_3 = 0 \\
\frac{\partial L}{\partial T_{12f}^{(1)}} &= -k \cdot C_f \cdot d_{12f}^{k+1} . {T_{12f}^{(1)}}^{-k-1} - \mu_1 - \mu_2 - \mu_3 = 0 \\
\frac{\partial L}{\partial T_{01f}^{(2)}} &= -k \cdot C_f \cdot d_{01f}^{k+1} . {T_{01f}^{(2)}}^{-k-1} - \mu_1 - \mu_3 = 0 \\
\frac{\partial L}{\partial T_{23f}^{(1)}} &= -k \cdot C_f \cdot d_{23f}^{k+1} . {T_{23f}^{(1)}}^{-k-1} - \mu_1 - \mu_2 = 0 \\
\frac{\partial L}{\partial T_{12f}^{(2)}} &= -k \cdot C_f \cdot d_{12f}^{k+1} . {T_{12f}^{(2)}}^{-k-1} - \mu_1 - \mu_2 - \mu_3 = 0
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial L}{\partial T_{23f}^{(2)}} &= -k \cdot C_f \cdot d_{23f}^{k+1} . {T_{23f}^{(2)}}^{-k-1} - \mu_1 - \mu_2 - \mu_3 = 0 \\
\frac{\partial L}{\partial T_{20e}} &= -k \cdot C_e \cdot d_{20e}^{k+1} \cdot T_{20e}^{-k-1} - \mu_1 - \mu_3 = 0 \\
\frac{\partial L}{\partial T_{12e}} &= -k \cdot C_e \cdot d_{12e}^{k+1} \cdot T_{12e}^{-k-1} - \mu_1 = 0 \\
\frac{\partial L}{\partial T_{31e}} &= -k \cdot C_e \cdot d_{31e}^{k+1} \cdot T_{31e}^{-k-1} - \mu_1 - \mu_2 = 0 \\
\frac{\partial L}{\partial T_{30e}} &= -k \cdot C_e \cdot d_{30e}^{k+1} \cdot T_{30e}^{-k-1} - \mu_1 - \mu_2 - \mu_3 = 0 \\
\frac{\partial L}{\partial p_{11}} &= -s \cdot C_{m_1} \cdot p_{11}^{-s-1} - \mu_1 - \mu_2 - \mu_3 = 0 \\
\frac{\partial L}{\partial p_{21}} &= -s \cdot C_{m_1} \cdot p_{21}^{-s-1} - \mu_3 = 0 \\
\frac{\partial L}{\partial p_{12}} &= -s \cdot C_{m_2} \cdot p_{12}^{-s-1} - \mu_2 = 0 \\
\frac{\partial L}{\partial p_{22}} &= -s \cdot C_{m_2} \cdot p_{22}^{-s-1} - \mu_1 - \mu_2 - \mu_3 = 0
\end{aligned}
$$

*Complementary Slackness:*

$$
\begin{aligned}
\mu_1 \cdot &\Big[\mathcal{K} - 12\epsilon - p_{11} - p_{22} - T_{01f}^{(1)} - T_{12f}^{(1)} - T_{20e} - T_{01f}^{(2)} - T_{12e} - T_{23f}^{(1)} \\
&- T_{31e} - T_{12f}^{(2)} - T_{23f}^{(2)} - T_{30e}\Big] = 0 \\
\mu_2 \cdot &\Big[\mathcal{K} - 10\epsilon - p_{11} - p_{22} - p_{12} - T_{01f}^{(1)} - T_{12f}^{(1)} - T_{23f}^{(1)} - T_{31e} - T_{12f}^{(2)} \\
&- T_{23f}^{(2)} - T_{30e}\Big] = 0 \\
\mu_3 \cdot &\Big[\mathcal{K} - 10\epsilon - p_{11} - p_{22} - p_{21} - T_{01f}^{(1)} - T_{12f}^{(1)} - T_{20e} - T_{01f}^{(2)} - T_{12f}^{(2)} \\
&+ T_{23f}^{(2)} + T_{30e}\Big] = 0
\end{aligned}
$$

*Primal feasibility:*

$$
\begin{aligned}
12\epsilon + p_{11} + p_{22} + T_{01f}^{(1)} + T_{12f}^{(1)} + T_{20e} + T_{01f}^{(2)} + T_{12e} + T_{23f}^{(1)} + T_{31e} + T_{12f}^{(2)} \\
+ T_{23f}^{(2)} + T_{30e} \leq \mathcal{K} \\
10\epsilon + p_{11} + p_{22} + p_{12} + T_{01f}^{(1)} + T_{12f}^{(1)} + T_{23f}^{(1)} + T_{31e} + T_{12f}^{(2)} + T_{23f}^{(2)} + T_{30e} \leq \mathcal{K} \\
10\epsilon + p_{11} + p_{22} + p_{21} + T_{01f}^{(1)} + T_{12f}^{(1)} + T_{20e} + T_{01f}^{(2)} + T_{12f}^{(2)} + T_{23f}^{(2)} + T_{30e} \leq \mathcal{K} \\
T_{01f}^{(1)}, T_{12f}^{(1)}, T_{20e}, T_{01f}^{(2)}, T_{12e}, T_{23f}^{(1)}, T_{31e}, T_{12f}^{(2)}, T_{23f}^{(2)}, T_{30e}, p_{11}, p_{12}, p_{21}, p_{22} \geq 0
\end{aligned}
$$

*Lagrange multipliers sign restrictions:* $\mu_y \leq 0 \qquad \forall y = 1, 2, 3$

If we assume $T^{UB}$ and $p^{UB}$ are sufficiently large, i.e. time bounds are loose at an optimal solution, then corresponding dual variables are zero due to complementary slackness. Proposition 6 shows that cycle time related constraints (3.48), (3.49), (3.50) must be tight at optimality.

**Proposition 6.** *Suppose that $\mathcal{K}$ is finite and $T_{ijh}^{UB}$ and $p_x^{UB}$ are sufficiently large, then cycle time constraints of $S_{12}$ cycle, constraints (3.48), (3.49), (3.50) are always binding in an optimal solution.*

*Proof.* Let $\mu_1, \mu_2$ and $\mu_3$ be the Lagrange multipliers corresponding to constraints (3.48), (3.49), (3.50), respectively. Consider the gradient equations below:

$$-k \cdot C_e \cdot d_{12e}^{k+1} \cdot T_{12e}^{-k-1} = \mu_1 \tag{3.53}$$

$$-s \cdot C_{m_1} \cdot p_{21}^{-s-1} = \mu_3 \tag{3.54}$$

$$-s \cdot C_{m_2} \cdot p_{12}^{-s-1} = \mu_2 \tag{3.55}$$

We can see left hand side of the equations above cannot be zero, then $\mu_1$, $\mu_2$ and $\mu_3$ values are not zero. Then, complementary slackness theorem says that constraints (3.48), (3.49), (3.50) are always equal to cycle time value $\mathcal{K}$ in an optimal solution. □

Similar to Propositions 1 and 4, Proposition 6 shows that at an efficient solution obtained by mathematical model above, energy consumption cannot be improved without increasing cycle time.

Proposition 7 shows that machine and robot partial waiting times are eliminated at an optimal solution.

**Proposition 7.** *If $\mathcal{K}$ is large enough, in an optimal solution, when both first part is processing on $M_2$ and second part is processing on $M_1$, robot's arrival and part completion on machines are synchronized i.e. both robot's and machine's waiting times are zero at optimality.*

*Proof.* We have already defined robot's partial waiting times in equations (3.11) and (3.12) as follows:

$$w_2 = \max\{0, p_{12} - T_{20e} - \epsilon - T^{(2)}_{01f} - \epsilon - T_{12e}\}$$
$$w_1 = \max\{0, p_{21} - T_{12e} - w_2 - \epsilon - T^{(1)}_{23f} - \epsilon - T_{31e}\}$$

When we consider constraints (3.48), (3.49) and (3.50) in the mathematical model, we know the all constraints are tight.

$$12\epsilon + p_{11} + p_{22} + T^{(1)}_{01f} + T^{(1)}_{12f} + T_{20e} + T^{(2)}_{01f} + T_{12e} + T^{(1)}_{23f} + T_{31e} + T^{(2)}_{12f} + T^{(2)}_{23f} + T_{30e} = \mathcal{K} \quad (3.56)$$

$$10\epsilon + p_{11} + p_{22} + p_{12} + T^{(1)}_{01f} + T^{(1)}_{12f} + T^{(1)}_{23f} + T_{31e} + T^{(2)}_{12f} + T^{(2)}_{23f} + T_{30e} = \mathcal{K} \quad (3.57)$$

$$10\epsilon + p_{11} + p_{22} + p_{21} + T^{(1)}_{01f} + T^{(1)}_{12f} + T_{20e} + T^{(2)}_{01f} + T^{(2)}_{12f} + T^{(2)}_{23f} + T_{30e} = \mathcal{K} \quad (3.58)$$

If we subtract equation (3.57) from (3.56), we get following expression:

$$p_{12} = 2\epsilon + T_{20e} + T^{(2)}_{01f} + T_{12e} \quad (3.59)$$

which gives $w_2 = 0$. Similarly, if we consider equations (3.56) and (3.58), we get:

$$p_{21} = 2\epsilon + T_{12e} + T^{(1)}_{23f} + T_{31e} \quad (3.60)$$

which together with $w_2 = 0$ gives $w_1 = 0$.

Let $w_{M_2}$ and $w_{M_1}$ denote the waiting time of $M_2$ for the robot after processing first part and the waiting time of $M_1$ for the robot after processing second part, respectively. Their mathematical expressions are:

$$w_{M_2} = max(0, 2\epsilon + T_{12e} + T^{(1)}_{23f} + T_{31e} + w_2 + w_1 - p_{21}) \quad (3.61)$$
$$w_{M_1} = max(0, 2\epsilon + T_{20e} + T^{(2)}_{01f} + T_{12e} + w_2 - p_{12}) \quad (3.62)$$

We can conclude that $w_{M_2} = 0$ and $w_{M_1} = 0$ by utilizing equations (3.59) and (3.60). Therefore, the moment $M_1$ completes the second part, the robot arrives at $M_1$ without any delay. Similarly, the robot arrives at $M_2$ as soon as $M_2$ completes the first part. □

Proposition 7 shows that in $S_{12}$ cycle, in an efficient solution robot's arrival time to a full machine and the machine's task completion time are equal, so that no waiting occurs. Eliminating waiting times by slowing down robot and machines saves energy.

In this chapter, we first gave the problem definition. We defined energy consumption functions of the robot and machines. We introduced robot activities for $S_1$, $S_2$ and $S_{12}$ cycles and give cycle time calculations for these cycles. Then, using $\epsilon$-constraint approach that finds efficient solution, we developed mathematical models. Finally, we carried out Karush-Kuhn-Tucker (KKT) analysis and gave several optimality properties for the efficient solutions.

In the next chapter, we will present our numerical experiments for problems.

# CHAPTER 4

# COMPUTATIONAL RESULTS

In this section, first we give a numerical study on the efficient frontier for the energy consumption and the cycle time objectives for three cycles: $S_1$, $S_2$ and $S_{12}$. Then, we compare these robotic cycles in terms of energy consumption and cycle time objectives. Lastly, we evaluate how speed control and processing time control strategies can contribute to energy savings in robotic cells.

## 4.1 Cycle Time vs. Energy Consumption

In this section, for a selected problem instance, we find a set of efficient solutions. We solve the mathematical models for $S_1$, $S_2$ and $S_{12}$ cycles for selected cycle time levels.

In this instance, $d_{01} = d_{12} = d_{23} = 2$, $d_{20} = d_{31} = 4$ , $d_{30} = 6$, $C_f = 4$, $C_e = 2$, $C_{m_1} = C_{m_2} = 400$, $v^{UB} = 2.2$, $p^{LB} = 5$, $k = 2$, $s = 1$ and $\epsilon = 4$. All values with increment of five in the range $[45, 85]$ are used for $\mathcal{K}$, i.e. $45, 50, 55, 60.., 85$. We consider machines and buffers are located within a linear layout in the robotic cell, and we assume that there are 2 meters between any pair of successive machines or a buffer and closest machine. In addition, we assume that when the robot is loaded, it consumes more energy during the move than when it is empty. Also, we consider identical machines. In the literature, we encounter studies taking maximum speed of a robot as $2.2m/s$. Therefore, we choose $2.2m/s$ as the upper limit of robot's speed. We solve all mathematical models via MS Excel Solver [28]. First, we solve $S_1$ model, and find the optimal energy consumption levels for all $\mathcal{K}$ values. A set of solutions representing the efficient frontier is given in Figure 4.1.

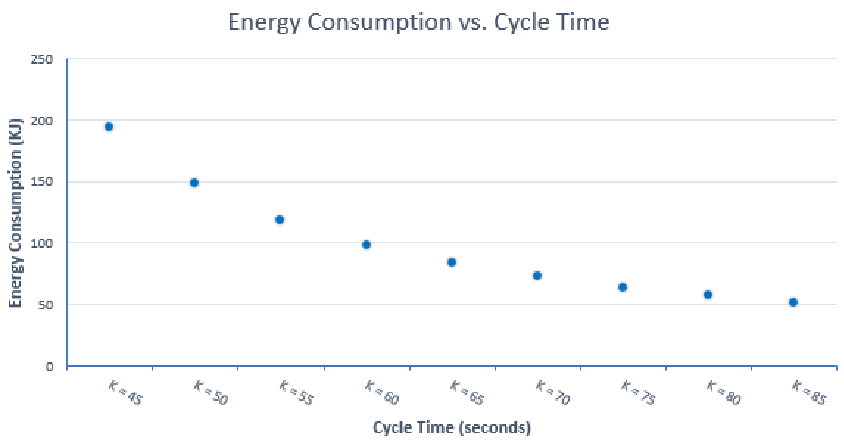


Figure 4.1: A set of efficient solutions for $S_1$ cycle

Figure 4.1 shows that as cycle time ($\mathcal{K}$) increases, efficient solutions have smaller energy consumption. The reason is that when cycle time is lower, the robot intends to move faster and the machines try to complete jobs in shorter times. The robot and the machines consume more energy to complete the cycle at shorter times as cycle time is decreased. As seen in Figure 4.1, as cycle time gets smaller, decreasing cycle time further becomes more costly in terms of energy consumption. This behaviour is due to convexity of energy consumption functions.

Operation planner can make use of the trade-off between energy consumption and cycle time to save energy. For example, if high throughput rate is required, the cell would work at lower cycle times albeit consuming more energy. On the other hand, energy consumption may be more crucial for decision maker when due dates for customer orders are relatively flexible. Then, the robotic cell can work at higher cycle times which would decrease the energy consumption. As can be concluded from Figure 4.1, operation planner can achieve energy saving by carefully planning robot speeds and processing times.

We summarize the results of $S_1$ cycle in Table 4.1. It includes energy consumption level, optimal robot move times and optimal processing times for given cycle time levels. When optimal travel times are examined, it can be realized that travel times for loaded moves are equal to each other. On the other hand, when the robot is empty, it moves faster compared to full moves, i.e. $T_{30e}$ has the highest value. This is because

$C_e < C_f$. Also, as we mention in Proposition 2, we see that $p_1 = p_2$ for considered cases given in Table 4.1, since we use the same energy consumption function for both machines.

Table 4.1: Cycle time, energy consumption and optimal results for $S_1$ cycle

| $\bar{\mathcal{K}}$ | EC (KJ) | $T_{01f}$ | $T_{12f}$ | $T_{23f}$ | $T_{30e}$ | $p_1$ | $p_2$ | Avg. Robot Speed $(m/s)$ |
|---|---|---|---|---|---|---|---|---|
| 45 | 194.4 | 1.8 | 1.8 | 1.8 | 4.2 | 5.8 | 5.8 | 1.2 |
| 50 | 147.8 | 2.1 | 2.1 | 2.1 | 4.9 | 7.4 | 7.4 | 1.0 |
| 55 | 118.3 | 2.4 | 2.4 | 2.4 | 5.6 | 9.1 | 9.1 | 0.9 |
| 60 | 98.1 | 2.7 | 2.7 | 2.7 | 6.3 | 10.8 | 10.8 | 0.8 |
| 65 | 83.5 | 2.9 | 2.9 | 2.9 | 7.0 | 12.6 | 12.6 | 0.7 |
| 70 | 72.4 | 3.2 | 3.2 | 3.2 | 7.6 | 14.4 | 14.4 | 0.7 |
| 75 | 63.8 | 3.5 | 3.5 | 3.5 | 8.3 | 16.2 | 16.2 | 0.6 |
| 80 | 56.9 | 3.7 | 3.7 | 3.7 | 8.9 | 18.0 | 18.0 | 0.6 |
| 85 | 51.3 | 4.0 | 4.0 | 4.0 | 9.5 | 19.8 | 19.8 | 0.5 |

Similarly, we generated a set of efficient solutions for $S_2$ cycle. The behavior of obtained solutions is given in Figure 4.2 and details are provided in Table 4.2.

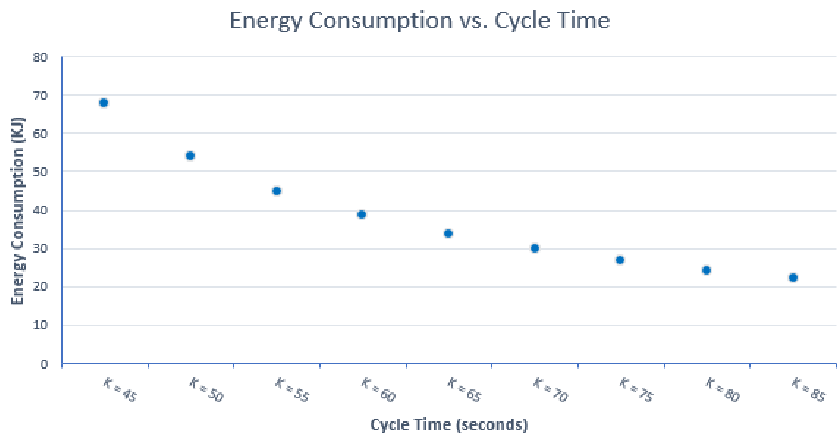


Figure 4.2: A set of efficient solutions for $S_2$ cycle

Table 4.2 shows that all processing times are equal to each other in each efficient solution. In $S_1$ cycle, we have full waiting times that cannot be eliminated. In order to catch target cycle time level, the robot and machines have to run faster. On the other hand, in $S_2$ cycle, although the robot travels a longer distance, it can move at a slower pace and save energy. Similarly, machines can run slower and save energy. As shown in Proposition 5, robot's arrival time to a machine for unloading and job completion time on the machine are synchronized. This is also observed in this example. In Table 4.3, we give the sum of duration of the operations performed by the robot during machine processing and the machine's processing time. It is seen that total handling time is exactly equal to machines' processing times. Neither the robot nor the machine waits the other, both finish their operations at the same time and the robot immediately unloads the machine after it arrives.

Table 4.2: Cycle time, energy consumption and optimal times for $S_2$ cycle

| $\bar{\mathcal{K}}$ | EC (KJ) | $T_{01f}$ | $T_{12e}$ | $T_{23f}$ | $T_{31e}$ | $T_{12f}$ | $T_{20e}$ | $p_1$ | $p_2$ | Avg. Robot Speed $(m/s)$ |
|---|---|---|---|---|---|---|---|---|---|---|
| 45 | 68.0 | 3.0 | 2.9 | 3.0 | 4.7 | 2.6 | 4.7 | 18.7 | 18.7 | 0.7 |
| 50 | 54.0 | 3.6 | 4.1 | 3.6 | 5.8 | 3.1 | 5.8 | 21.5 | 21.5 | 0.6 |
| 55 | 45.0 | 4.2 | 5.6 | 4.2 | 6.7 | 3.5 | 6.7 | 24.6 | 24.6 | 0.5 |
| 60 | 38.5 | 4.8 | 7.5 | 4.8 | 7.6 | 3.9 | 7.6 | 27.8 | 27.8 | 0.4 |
| 65 | 33.6 | 5.2 | 9.7 | 5.2 | 8.3 | 4.2 | 8.3 | 31.2 | 31.2 | 0.4 |
| 70 | 29.8 | 5.7 | 12.1 | 5.7 | 9.0 | 4.5 | 9.0 | 34.8 | 34.8 | 0.4 |
| 75 | 26.7 | 6.1 | 14.6 | 6.1 | 9.7 | 4.9 | 9.7 | 38.4 | 38.4 | 0.3 |
| 80 | 24.2 | 6.5 | 17.2 | 6.5 | 10.3 | 5.2 | 10.3 | 42.0 | 42.0 | 0.3 |
| 85 | 22.1 | 6.9 | 19.9 | 6.9 | 10.9 | 5.5 | 10.9 | 45.7 | 45.7 | 0.3 |

Table 4.3: Synchronization of robot and machines in $S_2$ cycle

| | Robot handling times while $M_1$ is busy | | | | | Robot handling times while $M_2$ is busy | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| $\bar{\mathcal{K}}$ | $T_{12e}$ | $T_{23f}$ | $T_{31e}$ | $2\epsilon$ | $p_1$ | $T_{01f}$ | $T_{12e}$ | $T_{20e}$ | $2\epsilon$ | $p_2$ |
| 45 | 2.9 | 3.0 | 4.7 | 8.0 | 18.7 | 3.0 | 2.9 | 4.7 | 8.0 | 18.7 |
| 50 | 4.1 | 3.6 | 5.8 | 8.0 | 21.5 | 3.6 | 4.1 | 5.8 | 8.0 | 21.5 |
| 55 | 5.6 | 4.2 | 6.7 | 8.0 | 24.6 | 4.2 | 5.6 | 6.7 | 8.0 | 24.6 |
| 60 | 7.5 | 4.8 | 7.6 | 8.0 | 27.8 | 4.8 | 7.5 | 7.6 | 8.0 | 27.8 |
| 65 | 9.7 | 5.2 | 8.3 | 8.0 | 31.2 | 5.2 | 9.7 | 8.3 | 8.0 | 31.2 |
| 70 | 12.1 | 5.7 | 9.0 | 8.0 | 34.8 | 5.7 | 12.1 | 9.0 | 8.0 | 34.8 |
| 75 | 14.6 | 6.1 | 9.7 | 8.0 | 38.4 | 6.1 | 14.6 | 9.7 | 8.0 | 38.4 |
| 80 | 17.2 | 6.5 | 10.3 | 8.0 | 42.0 | 6.5 | 17.2 | 10.3 | 8.0 | 42.0 |
| 85 | 19.9 | 6.9 | 10.9 | 8.0 | 45.7 | 6.9 | 19.9 | 10.9 | 8.0 | 45.7 |

Lastly, we find the efficient solutions for $S_{12}$ cycle. Different from the previous cycles, we use $2\mathcal{K}$ cycle time for this time i.e. we work with the range $[90, 170]$ because in $S_{12}$, each cycle produces two parts. Figure 4.3 shows the efficient frontier of $S_{12}$ cycle.

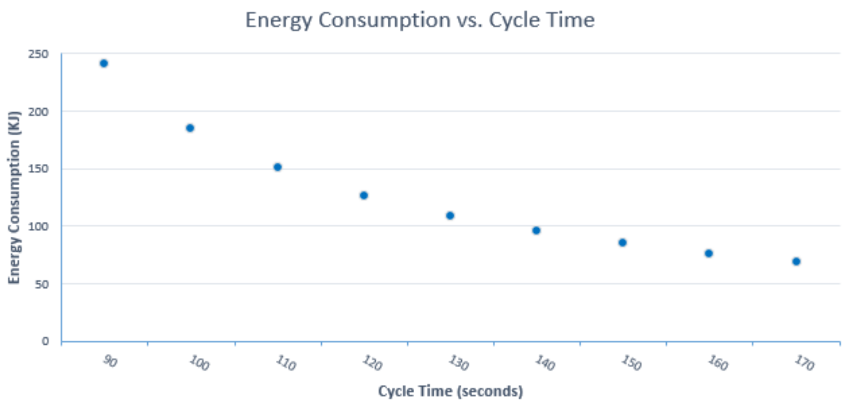


Figure 4.3: A set of efficient solutions for $S_{12}$ cycle

All optimal results obtained from solving mathematical model of $S_{12}$ cycle are given in Table 4.4.

Table 4.4: Cycle time, energy consumption and optimal times for $S_{12}$ cycle

| $\bar{\mathcal{K}}$ | EC (KJ) | $T^{(1)}_{01f}$ | $T^{(1)}_{12f}$ | $T_{20e}$ | $T^{(2)}_{01f}$ | $T_{12e}$ | $T^{(1)}_{23f}$ | $T_{31e}$ | $T^{(2)}_{12f}$ | $T^{(2)}_{23f}$ | $T_{30e}$ | $p_{11}$ | $p_{21}$ | $p_{12}$ | $p_{22}$ | Avg. Robot Speed $(m/s)$ | Avg. Processing Time $(s)$ |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 90 | 240.5 | 2.1 | 2.1 | 3.6 | 2.3 | 2.0 | 2.3 | 3.6 | 2.1 | 2.1 | 5.0 | 7.5 | 15.9 | 15.9 | 7.5 | 1.0 | 11.7 |
| 100 | 185.3 | 2.5 | 2.5 | 4.4 | 2.8 | 2.6 | 2.8 | 4.4 | 2.5 | 2.5 | 5.9 | 9.7 | 17.8 | 17.8 | 9.7 | 0.8 | 13.8 |
| 110 | 150.4 | 2.8 | 2.8 | 5.2 | 3.3 | 3.4 | 3.3 | 5.2 | 2.8 | 2.8 | 6.7 | 11.8 | 19.9 | 19.9 | 11.8 | 0.7 | 15.9 |
| 120 | 126.3 | 3.2 | 3.2 | 6.0 | 3.8 | 4.4 | 3.8 | 6.0 | 3.2 | 3.2 | 7.5 | 14.0 | 22.1 | 22.1 | 14.0 | 0.6 | 18.1 |
| 130 | 108.7 | 3.5 | 3.5 | 6.7 | 4.2 | 5.6 | 4.2 | 6.7 | 3.5 | 3.5 | 8.3 | 16.2 | 24.5 | 24.5 | 16.2 | 0.6 | 20.4 |
| 140 | 95.3 | 3.8 | 3.8 | 7.4 | 4.6 | 7.0 | 4.6 | 7.4 | 3.8 | 3.8 | 9.0 | 18.4 | 27.1 | 27.1 | 18.4 | 0.5 | 22.8 |
| 150 | 84.7 | 4.1 | 4.1 | 8.0 | 5.0 | 8.8 | 5.0 | 8.0 | 4.1 | 4.1 | 9.7 | 20.6 | 29.8 | 29.8 | 20.6 | 0.5 | 25.2 |
| 160 | 76.2 | 4.4 | 4.4 | 8.6 | 5.4 | 10.7 | 5.4 | 8.6 | 4.4 | 4.4 | 10.4 | 22.7 | 32.7 | 32.7 | 22.7 | 0.4 | 27.7 |
| 170 | 69.1 | 4.6 | 4.6 | 9.2 | 5.8 | 12.7 | 5.8 | 9.2 | 4.6 | 4.6 | 11.0 | 24.9 | 35.7 | 35.7 | 24.9 | 0.4 | 30.3 |

As shown in Proposition 7, robot's arrival time to a machine for unloading and job completion time on the machine are synchronized. This is also validated with the results given in Table 4.4. It is seen that total robot handling time while the second part is processing on $M_1$ is exactly equal to processing times of the second part on $M_1$. Similarly, total robot handling time while the first part is processing on $M_2$ is exactly equal to processing times of the first part on $M_2$.

In addition, we give average robot speed and average machine processing times in Table 4.4. We observe that as cycle time increases, robot speed and machine processing times decreases on the average. Since, higher cycle times give room to slow down the robot and machines.

## 4.2 Which Robot Cycle to Choose? $S_1$, $S_2$ or $S_{12}$ Cycle?

In this study, we present $S_1$, $S_2$ and $S_{12}$ cycles. The productivity of the cycles may vary depending on conditions. Therefore, choosing the right robot cycle is important.

Cycle time is the significant restriction for decision makers, so working with different cycle times may cause to choose different robot cycles. We analyze energy consumption for different cycle times with the same parameter values as used in previous section. We work with the range $[45, 85]$ for cycle time values.

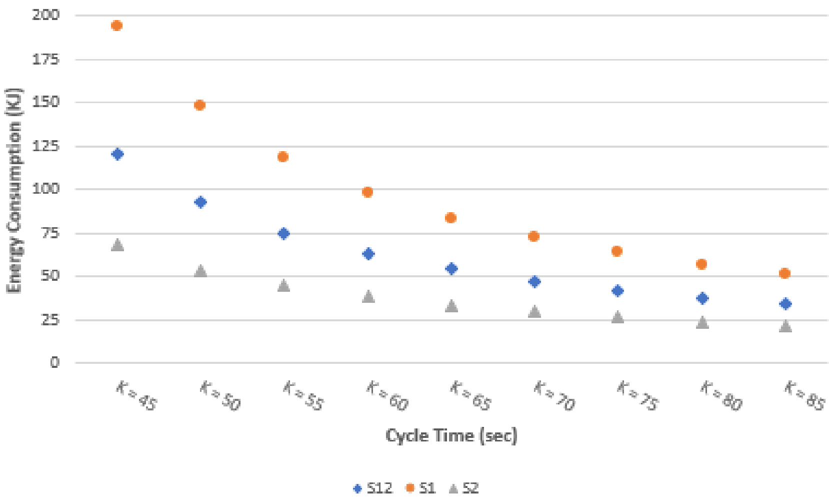


Figure 4.4: The efficiency of cycles in terms of cycle times

In Figure 4.4, $S_2$ is the best cycle for the considered instance because it gives the lowest energy consumption for all cycle time levels.

However, $S_2$ is not always the best cycle in terms of energy consumption. As parameters change, it is possible that the lowest energy consumption is given by a different cycle. We compute optimal energy consumption levels for three cycles assuming all machines and buffers are equidistant to each other, where $d$ is in the range $[4, 12]$.

Also, we assume cycle time is 65 seconds for $S_1$ and $S_2$ cycles. For $S_{12}$ cycle, we use the cycle time as 130 seconds and find the optimal energy consumption level per part produced. Remaining parameters we used for analysis are $C_f = 4$, $C_e = 2$, $C_{m_1} = C_{m_2} = 400$, $v^{UB} = 2.2$, $p^{LB} = 5$, $k = 2$, $s = 1$ and $\epsilon = 4$. Then, obtained results are summarized in Figure 4.5.

In Figure 4.5, we see that as the distance between machines and buffers increases, $S_2$ is no longer the most energy efficient alternative robot cycle. For $d < 10$, $S_2$ cycle has the lowest energy consumption. While $S_{12}$ provides the lowest energy consumption level for a quite narrow range of distance ($d \in [9.65, 9.73]$), for $d \geq 10$, $S_1$ cycle gives the lowest energy consumption as compared with other cycles. Hence, we understand from the Figure 4.5 that all cycles can be preferable under different conditions. When $S_2$ cycle is compared to $S_1$ and $S_{12}$ cycles, the robot is required to travel a longer distance to complete a cycle in $S_2$ cycle. As distance increases, it gets more expensive for $S_2$ cycle to achieve a given cycle time level. Since, $S_2$ cycle needs faster robot moves for the same cycle time. Thus, $S_2$ cycle becomes the worst alternative for energy consumption.

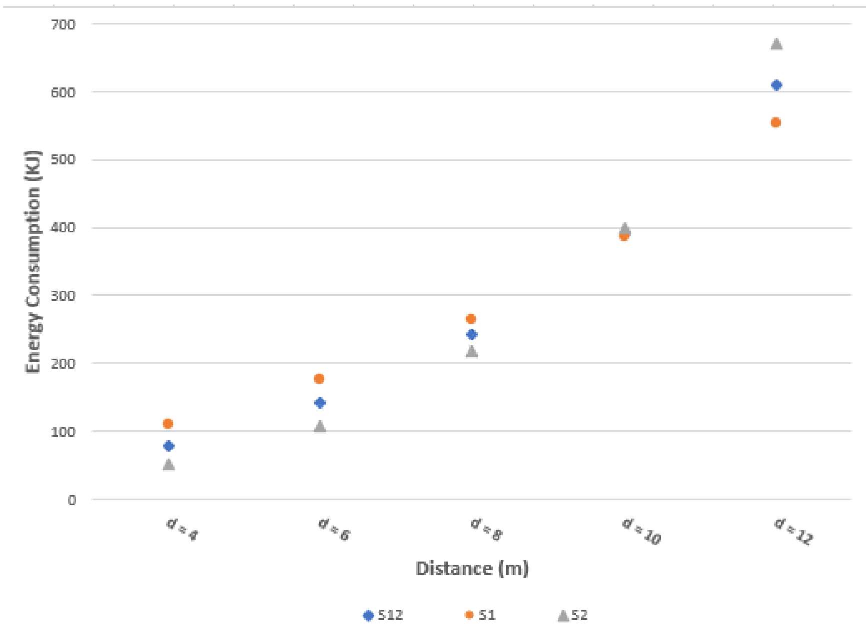


Figure 4.5: The efficiency of cycles in terms of distance

In practice, sometimes cycle time objective can be more important than energy consumption. In such a case, it is required to use the cycle which gives the minimum cycle time. We experiment considering the same parameters and the same assumptions as previous and observe which cycle gives feasible solutions for which cycle time levels. We consider the range [32,40] for $\mathcal{K}$. In Table 4.5, for the instance under consideration, $S_2$ cycle gives the lowest cycle time level. Then, $S_{12}$ cycle gives the second lowest cycle time.

Table 4.5: Minimum cycle time levels

| | EC (KJ) | | |
|---|---|---|---|
| $\mathcal{K}$ | $S_1$ | $S_2$ | $S_{12}$ |
| 32 | *Infeasible* | 236.4 | *Infeasible* |
| 33 | *Infeasible* | 198.3 | *Infeasible* |
| 34 | *Infeasible* | 170.4 | *Infeasible* |
| 35 | *Infeasible* | 149.3 | *Infeasible* |
| 36 | *Infeasible* | 132.9 | 267.5 |
| 37 | *Infeasible* | 119.8 | 229.9 |
| 38 | *Infeasible* | 109.1 | 203.8 |
| 39 | *Infeasible* | 100.2 | 184.8 |
| 40 | 298.7 | 92.7 | 169.9 |

To sum up, we conclude that there is no one cycle as the best cycle for all conditions in the robotic cells. Different conditions require to use a different cycle to obtain lowest energy consumption. Therefore, decision makers should carefully decide which robot cycle type to use and plan robot and machine speeds.

### 4.3 Benefits of Processing Time and Robot Speed Control

Typically, in practice, robots and machines operate at their maximum speed so as to obtain maximum throughput via achieving minimum cycle time. However, processing time and robot speed control can be used to achieve lower energy consumption levels. In this section, we analyze how much energy we save by utilizing the idea of processing time control along with robot speed control.

Table 4.6 gives experimental parameters and their values used in this study. In our experiments, we use four types of distance scenarios to analyze how different cell layout may affect energy consumption. Details of distance scenarios are given in Table 4.7. Constant scenario implies that all machines and buffers are equidistant to each other. Additive Identical and Additive General represent that all machines and buffers are located on a line in the robotic cell. Thus, additivity assumption is valid for these layouts, so $d_{ij} + d_{jk} = d_{ik}$ always holds where $i, j, k \in 0, 1, 2, 3$. There is a difference between two Additive General scenarios. Additive General I represents that machines are close to each other while the buffers are far from the machines. In contrast, in Additive General II, the buffers are close to the machines while machines are far from each other.

Table 4.6: Experimental settings

| Distance Scenarios: | Constant, Additive Identical, Additive General I, Additive General II |
|---|---|
| $k$: | 1, 2 |
| $s$: | 1, 2 |
| $C_f - C_e$: | $2.0 - 2.0$, $4.0 - 4.0$, $4.0 - 2.0$ |
| $C_{m_1} - C_{m_2}$: | $400.0 - 400.0$, $600.0 - 600.0$, $400.0 - 600.0$ |
| $v^{UB}$: | 1.5, 2.0 |
| $p^{LB}$: | 5-5, 15-15, 15-20 |

Table 4.7: The sets of distance values (given in meters) for each distance case

| Distance Case | $d_{01}$ | $d_{12}$ | $d_{23}$ | $d_{31}$ | $d_{20}$ | $d_{30}$ |
|---|---|---|---|---|---|---|
| Constant | 2 | 2 | 2 | 2 | 2 | 2 |
| Additive-Identical | 1.5 | 1.5 | 1.5 | 3 | 3 | 4.5 |
| Additive-General I | 2 | 1 | 2 | 3 | 3 | 5 |
| Additive General II | 1 | 2 | 1 | 3 | 3 | 4 |

In order to see how the shapes of energy consumption functions affect the results we considered two different values for both $k$ and $s$, $k = 1, k = 2$ and $s = 1, s = 2$. Also, we consider three alternative cases for $C_f$ and $C_e$. In two cases $C_f = C_e$, in the third case $C_f > C_e$. We want to see how the levels of $C_f$ and $C_e$ affect the results and also check the situation where loaded robot moves require higher energy. In order to see the effects of $C_{m_1}$ and $C_{m_2}$, we considered three cases, $C_{m_1} = C_{m_2}$ (low and high) and $C_{m_1} < C_{m_2}$. This will show us how different machine energy consumption functions affect robot speed and processing time decisions. The studies in the literature point out that speed of a handling robot varies from $0.05\ m/s$ to $2.2\ m/s$. Hence, we use two speed upper bound levels: $1.5\ m/s$ and $2.0\ m/s$. Also, we want to analyze the energy consumption when the machines can work fast, slow or they have different speeds, $p^{LB} : 5-5, 15-15, 15-20$. In the experiments, we take $\epsilon = 4$.

In our experiments, for robotic cycles $S_2$ and $S_{12}$, we create robotic cell schedules using four scenarios and then compare energy consumption levels. In the baseline scenario, both the robot and machines operate at their fastest pace to achieve the lowest possible cycle time ($\mathcal{CT}_B^{S_l}$). In the second scenario, only robot speed control is allowed and the mathematical model for cycle $S_l$ is solved to achieve $\mathcal{CT}_B^{S_l}$. Note that processing time decision variables are fixed at lower bounds. In the third scenario, only processing time control is allowed and robot moving time decision variables are fixed at lower bounds. Finally, both robot speed and processing time control is allowed. The energy consumption values are denoted by $\mathcal{F}_B^{S_l}$, $\mathcal{F}_{RC}^{S_l}$, $\mathcal{F}_{PC}^{S_l}$, $\mathcal{F}_{RC+PC}^{S_l}$ where $RC$ means robot speed control, $PC$ means processing time control. In the rest of the study, we will refer to these scenarios as $Scenario^{RC}$, $Scenario^{PC}$ and $Scenario^{RC+PC}$, respectively. Since there is no possibility that improvement on energy consumption occurs without worsening cycle time in $S_1$ cycle, we analyze only $S_2$ and $S_{12}$ cycles here.

In our experiments, for our three scenarios, we focus on how much energy savings can be obtained compared to the baseline situation. We present the improvement of saving on energy consumption as $Saving^{RC}$, $Saving^{PC}$ and $Saving^{RC+PC}$. Saving is calculated as $\frac{\mathcal{F}_B^{S_l} - \mathcal{F}_t^{S_l}}{\mathcal{F}_B^{S_l}}$ where $t \in \{RC, PC, RC+PC\}$ and $l \in \{2\}, \{12\}$.

### 4.3.1 Effects of different machine energy consumption functions

We compare the energy consumption savings for different machine energy consumption functions. For this analysis, we solve the mathematical models for different values of parameters $s$, $p^{LB}$ and $C_{m_1} - C_{m_2}$. Also, we use the Additive Identical distance scenario with $k = 1$, $C_f = 4$, $C_e = 2$, $v^{UB} = 1.5$ and $\epsilon = 4$. We give the results in Table 4.8 for $S_2$ and $S_{12}$ cycles, respectively.

Table 4.8: Energy saving levels on different machine energy consumption functions

| | | | $S_2$ | | | $S_{12}$ | | |
|---|---|---|---|---|---|---|---|---|
| $s$ | $p^{LB}$ | $C_{m_1} - C_{m_2}$ | $Saving^{RC}$ (%) | $Saving^{PC}$ (%) | $Saving^{RC+PC}$ (%) | $Saving^{RC}$ (%) | $Saving^{PC}$ (%) | $Saving^{RC+PC}$ (%) |
| s = 1 | 5 - 5 | 400 - 400 | 0.0 | 44.6 | 44.6 | 0.0 | 22.8 | 22.8 |
| | | 600 - 600 | 0.0 | 48.4 | 48.4 | 0.0 | 24.6 | 24.6 |
| | | 400 - 600 | 0.0 | 46.8 | 46.8 | 0.0 | 23.8 | 23.8 |
| | 15 - 15 | 400 - 400 | 3.3 | 0.0 | 3.3 | 1.7 | 0.0 | 1.7 |
| | | 600 - 600 | 2.6 | 0.0 | 2.6 | 1.4 | 0.0 | 1.4 |
| | | 400 - 600 | 2.9 | 0.0 | 2.9 | 1.5 | 0.0 | 1.5 |
| | 15 - 20 | 400 - 400 | 15.4 | 6.9 | 16.8 | 5.9 | 3.6 | 8.8 |
| | | 600 - 600 | 12.4 | 8.4 | 15.1 | 4.7 | 4.4 | 7.8 |
| | | 400 - 600 | 14.0 | 6.3 | 15.2 | 7.3 | 3.3 | 7.9 |
| s = 2 | 5 - 5 | 400 - 400 | 0.0 | 32.5 | 32.5 | 0.0 | 17.2 | 17.2 |
| | | 600 - 600 | 0.1 | 40.7 | 40.7 | 0.0 | 21.3 | 21.3 |
| | | 400 - 600 | 0.1 | 36.9 | 36.9 | 0.0 | 19.4 | 19.4 |
| | 15 - 15 | 400 - 400 | 6.4 | 0.0 | 6.4 | 3.5 | 0.0 | 3.5 |
| | | 600 - 600 | 6.2 | 0.0 | 6.2 | 3.4 | 0.0 | 3.4 |
| | | 400 - 600 | 6.3 | 0.0 | 6.3 | 3.4 | 0.0 | 3.4 |
| | 15 - 20 | 400 - 400 | 28.3 | 1.5 | 28.4 | 15.5 | 0.8 | 15.5 |
| | | 600 - 600 | 27.6 | 2.2 | 27.6 | 11.1 | 1.2 | 15.1 |
| | | 400 - 600 | 28.1 | 1.5 | 28.1 | 11.3 | 0.8 | 15.4 |

Usually, it is possible to say that both cycles show similar trends when parameters change. First, we consider different levels of $p^{LB}$. When $p^{LB}$ is low, $Saving^{RC}$ is low as the robot still has to move fast to catch up the machines and achieve corresponding $\mathcal{CT}_B^{S_l}$. On the other hand, when $p^{LB}$ is low, robot speed determines corresponding $\mathcal{CT}_B^{S_l}$ and there is room for slowing down the machines. Therefore, processing time control provides high savings on energy ($Saving^{PC}$). In contrast, as $p^{LB}$ rises, it becomes possible to slow down the robot speed as processing time of machines determine corresponding $\mathcal{CT}_B^{S_l}$. This time, robot speed control strategy gives high saving rates. In the case where the machines have high and different processing times from each other, $p^{LB} = 15 - 20$, robot speed control provides its highest improvements.

Moreover, we encounter the highest energy saving (up to 48.4%) by processing time control strategy when $s$ is low and $C_{m_1} - C_{m_2}$ are high. In this setting, machine energy consumption forms a larger portion of the total energy consumption. This leads to a higher saving by processing time control strategy. According to the selected parameters, the portion of the robot energy consumption in the total energy consumption can be quite low. In this case, as the level of improvement obtained from robot speed control is also low, the saving rate in the total energy consumption is small. On the contrary, when studied with a quadratic energy consumption function with lower $C_{m_1} - C_{m_2}$, i.e. the robot has larger portion of the total energy consumption, we see that scenarios controlling robot speed provides higher improvements. These observations hold for both cycles.

Table 4.9: Overall savings in terms of machine energy consumption function

| | | $S_2$ | | | $S_{12}$ | | |
|---|---|---|---|---|---|---|---|
| **Parameters** | **Levels** | Avg. $Saving^{RC}$ **(%)** | Avg. $Saving^{PC}$ **(%)** | Avg. $Saving^{RC+PC}$ **(%)** | Avg. $Saving^{RC}$ **(%)** | Avg. $Saving^{PC}$ **(%)** | Avg. $Saving^{RC+PC}$ **(%)** |
| $s$ | 1 | 5.6 | 17.9 | 21.7 | 2.5 | 9.2 | 11.1 |
| | 2 | 11.4 | 12.8 | 23.7 | 5.3 | 6.8 | 12.7 |
| $p^{LB}$ | 5 - 5 | 0.0 | 41.6 | 41.6 | 0.0 | 21.5 | 21.5 |
| | 15 - 15 | 4.6 | 0.0 | 4.6 | 2.5 | 0.0 | 2.5 |
| | 15 - 20 | 21.0 | 4.5 | 21.9 | 9.3 | 2.4 | 11.8 |
| $C_{m_1} - C_{m_2}$ | 400 - 400 | 8.9 | 14.2 | 22.0 | 4.4 | 7.4 | 11.6 |
| | 600 - 600 | 8.1 | 16.6 | 23.4 | 3.4 | 8.6 | 12.3 |
| | 400 - 600 | 8.6 | 15.2 | 22.7 | 3.9 | 7.9 | 11.9 |
| | *Overall* | **8.5** | **15.4** | **22.7** | **3.9** | **8.0** | **11.9** |

We summarize all the results from this analysis in Table 4.9 based on parameters and scenarios. For both cycles, when robot speed and processing times are controllable, we observe the highest energy saving as 22.7% and 11.9%, for $S_2$ and $S_{12}$ cycles, respectively. Then, when we could only control processing time, we calculate the second highest improvement on energy consumption with 15.4% and 8.0%, respectively. $Scenario^{RC}$ also gives the effective results in both cycles. When we consider overall results in terms of cycles, we can see that the saving rates getting from the $S_2$ cycle are better than the $S_{12}$ cycle for all scenarios.

### 4.3.2 Effects of different robot energy consumption function

We analyze energy savings for different robot energy consumption functions. We use different levels of $k$, $v^{UB}$ and $C_f - C_e$. We use Additive Identical distance scenario with $s = 1$, $C_{m_1} = C_{m_2} = 400$, $C_f = 4$, $C_e = 2$, $p^{LB} = 15$ or $20$ and $\epsilon = 4$. We present the results in Table 4.10 for $S_2$ and $S_{12}$ cycles.

Table 4.10: Energy saving levels on different robot energy consumption functions

| | | | $S_2$ | | | $S_{12}$ | | |
|---|---|---|---|---|---|---|---|---|
| $k$ | $v^{UB}$ | $C_f - C_e$ | $Saving^{RC}$ (%) | $Saving^{PC}$ (%) | $Saving^{RC+PC}$ (%) | $Saving^{RC}$ (%) | $Saving^{PC}$ (%) | $Saving^{RC+PC}$ (%) |
| k = 1 | 1.5 | 2.0 - 2.0 | 14.3 | 8.1 | 16.6 | 7.6 | 4.3 | 8.8 |
| | | 4.0 - 4.0 | 20.0 | 5.6 | 20.5 | 10.8 | 3.0 | 10.8 |
| | | 4.0 - 2.0 | 15.5 | 6.9 | 16.8 | 5.9 | 3.6 | 8.8 |
| | 2.0 | 2.0 - 2.0 | 18.5 | 7.0 | 19.8 | 9.9 | 3.8 | 9.9 |
| | | 4.0 - 4.0 | 24.5 | 4.7 | 24.7 | 13.4 | 2.6 | 13.5 |
| | | 4.0 - 2.0 | 19.4 | 5.9 | 20.0 | 10.2 | 3.1 | 10.2 |
| k = 2 | 1.5 | 2.0 - 2.0 | 23.6 | 6.6 | 24.8 | 12.6 | 3.6 | 12.6 |
| | | 4.0 - 4.0 | 30.8 | 4.3 | 30.9 | 16.8 | 2.4 | 16.9 |
| | | 4.0 - 2.0 | 24.6 | 5.5 | 25.2 | 13.0 | 2.9 | 13.0 |
| | 2.0 | 2.0 - 2.0 | 31.0 | 4.7 | 31.4 | 16.9 | 2.6 | 16.9 |
| | | 4.0 - 4.0 | 37.1 | 2.8 | 37.1 | 20.6 | 1.6 | 20.6 |
| | | 4.0 - 2.0 | 31.0 | 3.7 | 31.1 | 16.6 | 2.0 | 16.6 |

From the results given in Table 4.10, it can be seen that when the robot has a cubic energy consumption function instead of quadratic, $Scenario^{RC}$ and $Scenario^{RC+PC}$, can achieve higher savings compared to the $Scenario^{PC}$. In other words, as robot energy consumption function gets steeper robot speed decisions become more critical for energy consumption.

When $k$ and $C_f - C_e$ are increased, speeding up the robot becomes more expensive which also means slowing down gives higher energy saving. This is observed in Table 4.10 as expected. For our instances, we obtain up to 37.1% energy savings with high $k$ and $C_f - C_e$ values.

In addition, differentiating upper bound of robot speed affects energy saving. Higher $v^{UB}$ values, in other words faster robot, have more potential for the use of robot speed control strategy. In Table 4.10, we observe that when $v^{UB}$ increases more robot energy saving is possible.

Improvements on $Scenario^{PC}$ are relatively lower than $Scenario^{RC}$ and $Scenario^{RC+PC}$. The highest savings in terms of processing time control strategy are achieved when parameter values are low. In base scenario, when $k$, $C_f - C_e$ and $v^{UB}$ are low, the portion of robot energy consumption in total consumption decreases. In this case, percent saving achieved by processing time control increases.

Table 4.11: Overall savings in terms of robot energy consumption function

| | | $S_2$ | | | $S_{12}$ | | |
|---|---|---|---|---|---|---|---|
| **Parameters** | **Levels** | Avg. $Saving^{RC}$ **(%)** | Avg. $Saving^{PC}$ **(%)** | Avg. $Saving^{RC+PC}$ **(%)** | Avg. $Saving^{RC}$ **(%)** | Avg. $Saving^{PC}$ **(%)** | Avg. $Saving^{RC+PC}$ **(%)** |
| $k$ | 1 | 18.7 | 6.4 | 19.7 | 9.6 | 3.4 | 10.3 |
| | 2 | 29.7 | 4.6 | 30.1 | 16.1 | 2.5 | 16.1 |
| $v^{UB}$ | 1.5 | 21.5 | 6.2 | 22.5 | 11.1 | 3.3 | 11.8 |
| | 2.0 | 26.9 | 4.8 | 27.3 | 14.6 | 2.6 | 14.6 |
| $C_f - C_e$ | 2.0 - 2.0 | 21.9 | 6.6 | 23.2 | 11.7 | 3.5 | 12.1 |
| | 4.0 - 4.0 | 28.1 | 4.4 | 28.3 | 15.4 | 2.4 | 15.4 |
| | 4.0 - 2.0 | 22.6 | 5.5 | 23.3 | 11.4 | 2.9 | 12.2 |
| | *Overall* | **24.2** | **5.5** | **24.9** | **12.9** | **3.0** | **13.2** |

Overall results for this experiment is given in Table 4.11. As can be seen from the table, $Scenario^{RC+PC}$ gives the best energy consumption levels for our experiments. On the average, we can provide $24.9\%$ and $13.2\%$ energy savings for $S_2$ and $S_{12}$ cycle, respectively. The results show that energy saving due to robot speed control is sensitive to robot energy cost function parameters.

### 4.3.3 Effects of the robot's and machines' speed

In this experiment, we change $v^{UB}$ and $p^{LB}$ simultaneously. We use Additive Identical distance scenario with $k = 1$, $s = 1$, $C_{m_1} = C_{m_2} = 400$, $C_f = 4$, $C_e = 2$ and $\epsilon = 4$. We present the results in Table 4.12 for $S_2$ and $S_{12}$ cycles.

When machines are slow, the robot speed control strategy provides higher energy

Table 4.12: Energy saving levels on different robotic cell speed level

| $v^{UB}$ | $p^{LB}$ | $S_2$ $Saving^{RC}$ (%) | $S_2$ $Saving^{PC}$ (%) | $S_2$ $Saving^{RC+PC}$ (%) | $S_{12}$ $Saving^{RC}$ (%) | $S_{12}$ $Saving^{PC}$ (%) | $S_{12}$ $Saving^{RC+PC}$ (%) |
|---|---|---|---|---|---|---|---|
| 1.5 | 5 - 5 | 0.0 | 44.6 | 44.6 | 0.0 | 22.8 | 22.8 |
| | 15 - 15 | 3.3 | 0.0 | 3.3 | 1.7 | 0.0 | 1.7 |
| | 15 - 20 | 15.5 | 6.9 | 16.8 | 8.1 | 3.6 | 8.8 |
| 2.0 | 5 - 5 | 0.0 | 38.6 | 38.6 | 0.0 | 19.8 | 19.8 |
| | 15 - 15 | 4.3 | 0.0 | 4.2 | 2.2 | 0.0 | 2.2 |
| | 15 - 20 | 19.4 | 5.9 | 20.0 | 10.2 | 3.1 | 10.2 |

savings. Longer machine processing times lead to longer partial waiting which gives room to slow down the robot. When the robot is slower, we encounter the situations that the machines can wait for the robot. In this case, processing time control strategy saves higher energy.

It is seen from Table 4.12 that robot speed control strategy achieves highest saving when machines are slow and processing time control strategy achieves highest saving when robot is slow.

### 4.3.4 Effects of distance scenarios

In this section, we compare energy saving levels for different distance scenarios. We consider the total distance traveled by the robot in the cycle so that the scenarios are comparable with each other. For $S_2$, distances traveled by the robot in all scenarios are equal, and in $S_{12}$, distances traveled by the robot in all scenarios are very close to each other. We use parameter values $k = 1$, $s = 1$, $C_{m_1} = C_{m_2} = 400$, $C_f = 4$, $C_e = 2$, $v^{UB} = 1.5$, $p^{LB} = 15$ or 20 and $\epsilon = 4$. We summarize the results in Table 4.13.

As can be seen from Table 4.13, robot speed control strategy achieves highest energy saving level in *Constant* distance scenario. On the other hand, processing time control strategy performs better than other scenarios for *Additive General II*. However, the

Table 4.13: Energy saving levels on different distance cases

| **Distance** | $S_2$ | | | $S_{12}$ | | |
|---|---|---|---|---|---|---|
| | $Saving^{RC}$ (%) | $Saving^{PC}$ (%) | $Saving^{RC+PC}$ (%) | $Saving^{RC}$ (%) | $Saving^{PC}$ (%) | $Saving^{RC+PC}$ (%) |
| Constant | 16.0 | 6.6 | 17.5 | 8.5 | 3.5 | 9.3 |
| Additive Identical | 15.5 | 6.9 | 16.8 | 5.9 | 3.6 | 8.8 |
| Additive General I | 15.7 | 6.8 | 16.5 | 4.7 | 3.5 | 8.5 |
| Additive General II | 14.9 | 7.0 | 17.0 | 7.9 | 3.8 | 8.0 |
| *Overall* | **15.5** | **6.9** | **17.0** | **6.8** | **3.6** | **8.7** |

results are not far from each other. Therefore, we can say that energy savings achieved for different distance scenarios are close to each other.

All computational results in this section show that highest energy saving levels are obtained when the robot travel and machine processing times are controlled together. On the average, we observe that savings of up to 22.2 % and 11.7 % can be achieved by $S_2$ and $S_{12}$ cycles, respectively. In Table 4.14, we summarize the overall saving levels obtained from all instances given in this section. As expected, in $S_{12}$ cycle, we encounter lower saving levels compared to $S_2$ cycle. $S_2$ cycle includes partial waiting while the $S_{12}$ cycle includes both partial and full waiting. Since mathematical models can only improve on partial waiting times without affecting cycle times, it is an expected result that the energy saving levels in $S_{12}$ cycle are smaller.

Table 4.14: Overall energy saving levels for selected instances

| | $S_2$ | | | $S_{12}$ | | |
|---|---|---|---|---|---|---|
| | $Saving^{RC}$ (%) | $Saving^{PC}$ (%) | $Saving^{RC+PC}$ (%) | $Saving^{RC}$ (%) | $Saving^{PC}$ (%) | $Saving^{RC+PC}$ (%) |
| *Average* | 13.9 | 11.0 | 22.2 | 7.0 | 5.8 | 11.7 |

Computational results show that in a robotic cell schedule if robot waiting (i.e. partial waiting) and/or machine waiting times occur, then there is an opportunity to slow down these equipment and save energy while achieving the same throughput level.

# CHAPTER 5

# CONCLUSIONS

In this study, we consider a robotic cell with two machines having an input and an output buffer. While the machines in the robotic cell perform operations on a part, the robot carries out handling (between the machines or buffers), loading and unloading operations. We assume that we can control the robot speed and machines' processing times in this robotic cell. We perform our study on three different robot cycles; $S_1$, $S_2$ and $S_{12}$. Our objective is to minimize energy consumption and cycle time in a cycle. We apply $\epsilon$-constraint approach to the solution. We perform KKT analysis of the given mathematical models for each cycle. Then, we present properties of efficient solutions.

Our analysis on $S_2$ and $S_{12}$ cycles show that when the robot speed and machine processing times are controllable, partial waiting times for both robot and machines are eliminated to slow down the robot and the machines. Thus, we provide significant savings in energy consumption levels with the our proposed model.

We carry out an experimental study to see the effects of different factors on the results. We analyze the behavior of cycle time and energy consumption objectives by generating a set of efficient solutions representing the efficient frontier. We show that as the cycle time increases, the energy consumption decreases. Especially if the cycle time is shorter, we observe that increasing the cycle time a little more provides higher reductions in energy consumption. In addition, our analyses show that robot and machines speed control strategy achieves significant energy saving.

In addition, we study which cycle performs better in different distance scenarios and cycle times under robot speed and processing time control strategy. Also, we study how energy consumption is affected when robot speed and machine processing times are controlled together. We compare the proposed model with different assumptions. We consider the situation where machines and the robot work at their highest speed as base case. Computational results show that when robot speed and processing time control are considered together, significant saving can be obtained compared to the cases where two strategies are used individually.

As the use of robotic cells become widespread in manufacturing, energy efficient scheduling in these cells will become more important. In this study, it is explained that significant energy savings can be achieved by carefully planning robot and machine speeds by the help of proposed models.

Robotic cell configurations used in industries are various. There are many studies considering different types of robotic cells in the literature. These studies have particularly focused on trajectory optimization. As a future study, different robotic cell configurations can be considered. When the robot and machine speed control is also evaluated besides trajectory optimization, energy efficient results can be obtained.

Another future research could be to reconsider our models with different energy consumption functions. A variety of energy consumption functions are used for robotic cell studies in the literature. A more realistic energy consumption function considering acceleration/deceleration, stand-by mode etc. can provide more realistic results.

Also, the mathematical models and analyzes given in this study can be carried out for robotic cells with three or more machines. Increasing the number of machines of the robotic cell can create different requirements in terms of decision variables and constraints. In addition, the solution of the problem will be more time consuming. Thus, this recommendation can be considered as another research topic in the future.

# REFERENCES

[1] M. Abdulkader, M. ElBeheiry, N. Afia, and A. El-Kharbotly. Scheduling and sequencing in four machines robotic cell: Application of genetic algorithm and enumeration techniques. *Ain Shams Engineering Journal*, 4(3):465–474, 2013.

[2] A. Agnetis. Scheduling no-wait robotic cells with two and three machines. *European Journal of Operational Research*, 123(2):303–314, 2000.

[3] M. S. Aktürk, A. Atamtürk, and S. Gürel. Parallel machine match-up scheduling with manufacturing cost considerations. *Journal of Scheduling*, 13(1):95–110, 2010.

[4] S. Alatartsev, V. Mersheeva, M. Augustine, and F. Ortmeier. On optimizing a sequence of robotic tasks. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 217–223. IEEE, 2013.

[5] D. Alcaide, C. Chu, V. Kats, E. Levner, and G. Sierksma. Cyclic multiple-robot scheduling with time-window constraints using a critical path approach. *European Journal of Operational Research*, 177(1):147–162, 2007.

[6] G. D. Batur, O. E. Karasan, and M. S. Aktürk. Multiple part-type scheduling in flexible robotic cells. *International Journal of Production Economics*, 135(2):726–740, 2012.

[7] D. Biskup and T. E. CHENG. Single-machine scheduling with controllable processing times and earliness, tardiness and completion time penalties. *Engineering Optimization*, 31(3):329–336, 1999.

[8] BP. Statistical Review of World Energy 2020. Technical report, 2020.

[9] N. Brauner. Identical part production in cyclic robotic cells: Concepts, overview and open questions. *Discrete Applied Mathematics*, 156(13):2480–2492, 2008.

[10] N. Brauner and G. Finke. On a conjecture about robotic cells: new simplified proof for the three-machine case. *INFOR: Information Systems and Operational Research*, 37(1):20–36, 1999.

[11] N. Brauner and G. Finke. Cycles and permutations in robotic cells. *Mathematical and Computer Modelling*, 34(5-6):565–591, 2001.

[12] M. Brossog, M. Bornschlegl, J. Franke, et al. Reducing the energy consumption of industrial robots in manufacturing systems. *The International Journal of Advanced Manufacturing Technology*, 78(5-8):1315–1328, 2015.

[13] L. Bukata, P. Šůcha, Z. Hanzálek, and P. Burget. Energy optimization of robotic cells. *IEEE Transactions on Industrial Informatics*, 13(1):92–102, 2017.

[14] L. Bukata, P. Sucha, and Z. Hanzálek. Optimizing energy consumption of robotic cells by a branch & bound algorithm. *Computers & Operations Research*, 102:52–66, 2019.

[15] K. Bunse, M. Vodicka, P. Schönsleben, M. Brülhart, and F. O. Ernst. Integrating energy efficiency performance in production management–gap analysis between industrial needs and scientific literature. *Journal of Cleaner Production*, 19(6-7):667–679, 2011.

[16] G. Carabin, E. Wehrle, and R. Vidoni. A review on energy-saving optimization methods for robotic and automatic systems. *Robotics*, 6(4):39, 2017.

[17] M. Chemnitz, G. Schreck, and J. Krüger. Analyzing energy consumption of industrial robots. In *ETFA2011*, pages 1–4. IEEE, 2011.

[18] Z.-L. Chen, Q. Lu, and G. Tang. Single machine scheduling with discretely controllable processing times. *Operations Research Letters*, 21(2):69–76, 1997.

[19] Y. Crama, V. Kats, J. Van de Klundert, and E. Levner. Cyclic scheduling in robotic flowshops. *Annals of Operations Research*, 96(1):97–124, 2000.

[20] Y. Crama and J. Van De Klundert. Cyclic scheduling of identical parts in a robotic cell. *Operations Research*, 45(6):952–965, 1997.

[21] Y. Crama and J. Van de Klundert. Cyclic scheduling in 3-machine robotic flow shops. *Journal of Scheduling*, 2(1):35–54, 1999.

[22] R. L. Daniels and R. K. Sarin. Single machine scheduling with controllable processing times and number of jobs tardy. *Operations Research*, 37(6):981–984, 1989.

[23] M. Dawande, H. N. Geismar, S. P. Sethi, and C. Sriskandarajah. Sequencing and scheduling in robotic cells: Recent developments. *Journal of Scheduling*, 8(5):387–426, 2005.

[24] M. W. Dawande, H. N. Geismar, S. P. Sethi, and C. Sriskandarajah. *Throughput Optimization in Robotic Cells*, volume 101. Springer Science & Business Media, 2007.

[25] K. Eggers, Z. Ziaukas, J. Kotlarski, and T. Ortmaier. On the relationship of travel time and energy efficiency of industrial robots. In *2018 International Conference on Industrial Enterprise and System Engineering (ICoIESE 2018)*, pages 115–121. Atlantis Press, 2019.

[26] A. Elmi and S. Topaloglu. A scheduling problem in blocking hybrid flow shop robotic cells with multiple robots. *Computers & Operations Research*, 40(10):2543–2555, 2013.

[27] J. Engelmann. *Methoden und Werkzeuge zur Planung und Gestaltung energieeffizienter Fabriken*. Wissenschaftliche Schriftenreihe des Institutes für Betriebswissenschaften und Fabriksysteme: Institut für Betriebswissenschaften und Fabriksysteme. IBF, 2009.

[28] D. H. Fylstra, L. Lasdon, J. Watson, and A. Waren. Design and use of the microsoft excel solver. *Interfaces*, 28:29–55, 1998.

[29] A. Fysikopoulos, D. Anagnostakis, K. Salonitis, and G. Chryssolouris. An empirical study of the energy consumption in automotive assembly. *Procedia Cirp*, 3:477–482, 2012.

[30] M. Gadaleta, M. Pellicciari, and G. Berselli. Optimization of the energy consumption of industrial robots for automatic code generation. *Robotics and Computer-Integrated Manufacturing*, 57:452–464, 2019.

[31] H. N. Geismar, M. Dawande, and C. Sriskandarajah. Robotic cells with parallel machines: Throughput maximization in constant travel-time cells. *Journal of Scheduling*, 7(5):375–395, 2004.

[32] E. Glorieux, S. Riazi, and B. Lennartson. Productivity/energy optimisation of trajectories and coordination for cyclic multi-robot systems. *Robotics and Computer-Integrated Manufacturing*, 49:152–161, 2018.

[33] A. Grau, M. Indri, L. L. Bello, and T. Sauter. Industrial robotics in factory automation: From the early stage to the internet of things. In *IECON 2017-43rd Annual Conference of the IEEE Industrial Electronics Society*, pages 6159–6164. IEEE, 2017.

[34] H. Gültekin, M. S. Aktürk, and O. E. Karasan. Bicriteria robotic cell scheduling. *Journal of Scheduling*, 11(6):457–473, 2008.

[35] S. Gürel and M. S. Aktürk. Considering manufacturing cost and scheduling performance on a CNC turning machine. *European Journal of Operational Research*, 177(1):325–343, 2007.

[36] S. Gürel, H. Gültekin, and V. E. Akhlaghi. Energy conscious scheduling of a material handling robot in a manufacturing cell. *Robotics and Computer-Integrated Manufacturing*, 58:97–108, 2019.

[37] N. G. Hall, H. Kamoun, and C. Sriskandarajah. Scheduling in robotic cells: classification, two and three machine cells. *Operations Research*, 45(3):421–439, 1997.

[38] N. G. Hall and C. Sriskandarajah. A survey of machine scheduling problems with blocking and no-wait in process. *Operations Research*, 44(3):510–525, 1996.

[39] C. Hansen, J. Öltjen, D. Meike, and T. Ortmaier. Enhanced approach for energy-efficient trajectory generation of industrial robots. In *2012 IEEE International Conference on Automation Science and Engineering (CASE)*, pages 1–7. IEEE, 2012.

[40] H. Hoogeveen. Multicriteria scheduling. *European Journal of Operational Research*, 167(3):592–623, 2005.

[41] I. International Federation of Robotics. Executive Summary World Robotics 2020 Industrial Robots. Technical report, 2020.

[42] M. Janardhanan, G. Q. Huang, and S. G. Ponnambalam. An investigation on minimizing cycle time and total energy consumption in robotic assembly line systems. *Journal of Cleaner Production*, 90:311–325, 2015.

[43] H. Kamoun, N. G. Hall, and C. Sriskandarajah. Scheduling in robotic cells: Heuristics and cell design. *Operations Research*, 47(6):821–835, 1999.

[44] S. Karabati and P. Kouvelis. Flow-line scheduling problem with controllable processing times. *IIE Transactions*, 29(1):1–14, 1997.

[45] A. Kobetski and M. Fabian. Velocity balancing in flexible manufacturing systems. In *2008 9th International Workshop on Discrete Event Systems*, pages 358–363. IEEE, 2008.

[46] Z. Kolíbal and A. Smetanová. Experimental implementation of energy optimization by robot movement. In *19th International Workshop on Robotics in Alpe-Adria-Danube Region (RAAD 2010)*, pages 333–339. IEEE, 2010.

[47] C.-Y. Lee, L. Lei, and M. Pinedo. Current trends in deterministic scheduling. *Annals of Operations Research*, 70:1–41, 1997.

[48] E. Levner, V. Kats, D. A. L. De Pablo, and T. E. Cheng. Complexity of cyclic scheduling problems: A state-of-the-art survey. *Computers & Industrial Engineering*, 59(2):352–361, 2010.

[49] S. D. Liman, S. S. Panwalkar, and S. Thongmee. A single machine scheduling problem with common due window and controllable processing times. *Annals of Operations Research*, 70:145–154, 1997.

[50] D. Meike, M. Pellicciari, and G. Berselli. Energy efficient use of multirobot production lines in the automotive industry: Detailed system modeling and optimization. *IEEE Transactions on Automation Science and Engineering*, 11(3):798–809, 2013.

[51] D. Meike and L. Ribickis. Energy efficient use of robotics in the automobile

industry. In *2011 15th International Conference on Advanced Robotics (ICAR)*, pages 507–511. IEEE, 2011.

[52] E. Nowicki and S. Zdrzałka. A two-machine flow shop scheduling problem with controllable job processing times. *European Journal of Operational Research*, 34(2):208–220, 1988.

[53] E. Nowicki and S. Zdrzałka. A survey of results for sequencing problems with controllable processing times. *Discrete Applied Mathematics*, 26(2-3):271–287, 1990.

[54] S. Panwalkar and R. Rajagopalan. Single-machine sequencing with controllable processing times. *European Journal of Operational Research*, 59(2):298–302, 1992.

[55] S. Pellegrinelli, S. Borgia, N. Pedrocchi, E. Villagrossi, G. Bianchi, and L. M. Tosatti. Minimization of the energy consumption in motion planning for single-robot tasks. *Procedia Cirp*, 29:354–359, 2015.

[56] M. Pellicciari, G. Berselli, F. Leali, and A. Vergnano. A method for reducing the energy consumption of pick-and-place industrial robots. *Mechatronics*, 23(3):326–334, 2013.

[57] S. Riazi, O. Wigström, K. Bengtsson, and B. Lennartson. Energy and peak power optimization of time-bounded robot trajectories. *IEEE Transactions on Automation Science and Engineering*, 14(2):646–657, 2017.

[58] S. P. Sethi, C. Sriskandarajah, G. Sorger, J. Blazewicz, and W. Kubiak. Sequencing of parts and robot moves in a robotic cell. *International Journal of Flexible Manufacturing Systems*, 4(3):331–358, 1992.

[59] D. Shabtay and M. Kaspi. Minimizing the total weighted flow time in a single machine with controllable processing times. *Computers & Operations Research*, 31(13):2279–2289, 2004.

[60] D. Shabtay and G. Steiner. A survey of scheduling with controllable processing times. *Discrete Applied Mathematics*, 155(13):1643–1666, 2007.

[61] A. Shioura, N. V. Shakhlevich, and V. A. Strusevich. Preemptive models of scheduling with controllable processing times and of scheduling with imprecise computation: A review of solution approaches. *European Journal of Operational Research*, 266(3):795–818, 2018.

[62] C. Sriskandarajah, I. Drobouchevitch, S. P. Sethi, and R. Chandrasekaran. Scheduling multiple parts in a robotic cell served by a dual-gripper robot. *Operations Research*, 52(1):65–82, 2004.

[63] Z. ürük, H. Gültekin, and M. S. Aktürk. Two-machine flowshop scheduling with flexible operations and controllable processing times. *Computers & Operations Research*, 40(2):639–653, 2013.

[64] U.S. Energy Information Administration. January 2021 Monthly Energy Review. Technical Report 3, 2021.

[65] R. Vickson. Two single machine sequencing problems involving controllable job processing times. *AIIE Transactions*, 12(3):258–262, 1980.

[66] O. Wigström and B. Lennartson. Sustainable production automation-energy optimization of robot cells. In *2013 IEEE International Conference on Robotics and Automation*, pages 252–257. IEEE, 2013.

[67] O. Wigstrom, B. Lennartson, A. Vergnano, and C. Breitholtz. High-level scheduling of energy optimal trajectories. *IEEE Transactions on Automation Science and Engineering*, 10(1):57–64, 2012.

[68] P. Yan, C. Chu, N. Yang, and A. Che. A branch and bound algorithm for optimal cyclic scheduling in a robotic cell with processing time windows. *International Journal of Production Research*, 48(21):6461–6480, 2010.

[69] S. Yildiz, M. S. Aktürk, and O. E. Karasan. Bicriteria robotic cell scheduling with controllable processing times. *International Journal of Production Research*, 49(2):569–583, 2011.

[70] M. F. Zarandi, H. Mosadegh, and M. Fattahi. Two-machine robotic cell scheduling problem with sequence-dependent setup times. *Computers & Operations Research*, 40(5):1420–1434, 2013.