

DOKUZ EYLÜL UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

**CAPABILITY-BASED DISTRIBUTED LAYOUT
DESIGN WITH UNEQUAL AREA
DEPARTMENTS UNDER STOCHASTIC
MACHINE RELIABILITY AND FUZZY DEMAND
INFORMATION**

by
Bilge VAROL

July, 2022
İZMİR

**CAPABILITY-BASED DISTRIBUTED LAYOUT
DESIGN WITH UNEQUAL AREA
DEPARTMENTS UNDER STOCHASTIC
MACHINE RELIABILITY AND FUZZY DEMAND
INFORMATION**

**A Thesis Submitted to the
Graduate School of Natural and Applied Sciences of Dokuz Eylül University
In Partial Fulfillment of the Requirements for the Degree of Master of
Science in Industrial Engineering**

**by
Bilge VAROL**

**July, 2022
İZMİR**

M.Sc THESIS EXAMINATION RESULT FORM

We have read the thesis entitled “**CAPABILITY-BASED DISTRIBUTED LAYOUT DESIGN WITH UNEQUAL AREA DEPARTMENTS UNDER STOCHASTIC MACHINE RELIABILITY AND FUZZY DEMAND INFORMATION**” completed by **BİLGE VAROL** under supervision of **ASSOC. PROF. DR. KEMAL SUBULAN** and we certify that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

.....
Assoc. Prof. Dr. Kemal SUBULAN

Supervisor

.....
Prof. Dr. Adil BAYKASOĞLU

.....
Assoc. Prof. Dr. Alper HAMZADAYI

(Jury Member)

(Jury Member)

Prof. Dr. Okan Fıstıkoğlu

Director

Graduate School of Natural and Applied Sciences

ACKNOWLEDGMENT

First of all, I would like to thank my Master's thesis supervisor, Assoc. Prof. Dr. Kemal SUBULAN, who was with me during the preparation of this thesis and who stood by me and supported me in each step of this thesis. If he wasn't my supervisor, I wouldn't have been able to develop myself this much academically and I wouldn't have been successful, so I am grateful to him. I would also thank to Prof. Dr. Adil BAYKASOĞLU for his leading throughout the preparation of this thesis.

I would also like to thank my dear parents Sibel and Serhat VAROL for their support and love in every moment of my life, and for always being there for me.

Bilge VAROL

CAPABILITY-BASED DISTRIBUTED LAYOUT DESIGN WITH UNEQUAL AREA DEPARTMENTS UNDER STOCHASTIC MACHINE RELIABILITY AND FUZZY DEMAND INFORMATION

ABSTRACT

In the first part of this thesis, a hybrid approach based on the chance-constrained stochastic program and an interactive fuzzy resolution method is proposed for modeling and solving a fuzzy-stochastic capability-based distributed layout problem with random machine availability and fuzzy demand/process flow data. Because, in today's highly dynamic manufacturing environments, most of the real-life applications involve different types of uncertainties simultaneously, like fuzziness and stochasticity. In order to achieve a robust layout, probabilistic machine availabilities should be considered along with uncertain product demand. In order to demonstrate validity and applicability of the proposed approach, a numerical example is first presented. Later, its performance is also tested on a real-life problem. Computational experiments shown that the more reliable robust layouts can be generated via the proposed approach under different scenarios and uncertainty levels.

In the second part, a novel unequal-area capability-based facility layout design (UA-CBFLD) problem is presented for the first time in the literature. In addition to the machines' unequal-area requirements, their processing capabilities are also considered. A novel mixed-integer non-linear programming model with a polyhedral inner-approximation method is first developed. Due to its NP-hard nature and highly non-linear structure, a heuristic decomposition-based iterative mathematical programming approach is employed to solve realistic size problem instances within reasonable computing times. In order to show practicality of the proposed MINLP model and solution approach, a real-life case study of a manufacturing company is presented. Computational results shown that superior layout scores can be obtained via the proposed approach under shorter run times.

Keywords: Capability-based distributed layout design, chance-constrained stochastic programming, random machine availability, fuzzy mathematical programming, unequal area facility layout design problem, mixed-integer programming; heuristic decomposition-based solution approach.



STOKASTİK MAKİNE GÜVENİLİRLİĞİ VE BULANIK TALEP BİLGİSİ ALTINDA EŞİT OLMAYAN ALANLI DEPARTMANLARI İÇEREN YETENEK TABANLI DAĞITIK YERLEŞİM DÜZENİ TASARIMI

ÖZ

Bu tezin ilk bölümünde, rassal makine güvenilirliği ve bulanık talep/proses akış verisi içeren, yetenek tabanlı dağıtık yerleşim düzeni probleminin modellenmesi ve çözümü için, şans kısıt tabanlı stokastik programlama ve etkileşimli bulanık çözümleme tekniklerine dayanan hibrit bir yaklaşım önerilmiştir. Günümüzün oldukça dinamik üretim ortamlarında, çoğu gerçek hayat uygulaması, bulanıklık ve rassallık gibi farklı türden belirsizlikleri eş zamanlı olarak içermektedir. Bu nedenle, etkin ve efektif tesis yerleşim düzenleri elde etmek amacıyla, rassal makine kullanılabilirliği ve belirsiz ürün talebi birlikte ele alınmalıdır. Önerilen tesis yerleşim düzenlemesi yaklaşımının uygulanabilirliğinin test edilmesi amacıyla, önce küçük boyutlu bir nümerik örnek sunulmuştur. Daha sonra, büyük boyutlu bir gerçek hayat problemi üzerinde performansı test edilmiştir. Elde edilen optimizasyon sonuçları doğrultusunda, önerilen yaklaşım ile farklı senaryolar ve belirsizlik düzeyleri altında, etkin ve güvenilir dağıtık yerleşim düzeni alternatiflerinin üretilmektedir.

Tezin ikinci bölümünde ise, eşit alanlı olmayan yeni bir yetenek tabanlı tesis yerleşim düzeni problemi, literatürde ilk defa ele alınmıştır. Makinelerin eşit olmayan alan gereksinimlerine ek olarak, proses işleme yetenekleri de dikkate alınmıştır. İlk olarak, doğrusal olmayan yeni bir karışık tamsayılı programlama modeli geliştirilmiş; sonrasında ise doğrusal olmayan alan kısıtları için bir içsel yaklaşım metodundan faydalanılmıştır. Problemin NP-zor ve doğrusal olmayan yapısı nedeniyle, büyük boyutlu problemlerin makul sürelerde çözülebilmesi için, sezgisel ayrıştırma temelli yinelemeli bir matematiksel programlama yaklaşımından yararlanılmıştır. Bu yaklaşımın uygulanabilirliğinin test edilmesi amacıyla, önce küçük boyutlu bir örnek sunulmuş; sonrasında bir üretim işletmesine ait gerçek hayat uygulamasına yer verilmiştir. Elde edilen sonuçlar, büyük boyutlu problemlerin önerilen yaklaşım ile çok daha kısa sürede, daha iyi skora sahip çözümler üretebileceğini göstermektedir.

Anahtar kelimeler: Yetenek tabanlı dağıtık yerleşim düzeni tasarımı, şans kısıt tabanlı stokastik programlama, rassal makine güvenilirliği/kullanılabilirliği, bulanık matematiksel programlama, eşit alanlı olmayan tesis yerleşim düzeni problemi, karışık tamsayılı programlama, sezgisel-ayrıştırma temelli çözüm yaklaşımı.



CONTENTS

	Page
M.Sc THESIS EXAMINATION RESULT FORM.....	ii
ACKNOWLEDGMENT.....	iii
ABSTRACT.....	iv
ÖZ	vi
LIST OF FIGURES	x
LIST OF TABLES	xii
CHAPTER 1-INTRODUCTION	1
1.1 Background and Research Motivations.....	1
1.2 Contributions and Framework of the Thesis	3
1.3 Outline of the Thesis	6
1.4 Produced Publications Based on Research in the Scope of This Thesis	7
CHAPTER 2-LITERATURE REVIEW	9
2.1 Literature Review on Robust Facility Layout Design Problems under Uncertainty	9
2.2 Literature Survey on the Unequal-Area Facility Layout Design Problems	18
CHAPTER 3-DESIGNING ROBUST CAPABILITY-BASED DISTRIBUTED MACHINE LAYOUTS WITH RANDOM MACHINE AVAILABILITY AND FUZZY DEMAND/PROCESS FLOW INFORMATION.....	32
3.1 Problem Description and Mathematical Formulation of the Robust FS- BCBDL Problem	32
3.1.1 The Mixed-Integer Programming (MIP) Formulation	33
3.1.2 The Proposed Hybrid Chance-Constrained Stochastic Program with A Fuzzy Resolution Approach	40
3.2 Computational Study	44
3.2.1 An Illustrative Example with Comparative Results	44
3.2.2 Application in a Real-Life Manufacturing System.....	62

CHAPTER 4-UNEQUAL-AREA CAPABILITY-BASED FACILITY LAYOUT DESIGN PROBLEM WITH A HEURISTIC DECOMPOSITION-BASED ITERATIVE MATHEMATICAL PROGRAMMING APPROACH.....	75
4.1 Problem Formulation and Mathematical Formulation of the UA-CBFLD ...	75
4.1.1 The Proposed MINLP Model Formulation for the UA-CBFLD Problem	78
4.1.2 Linearization of the Non-Linear Constraints and Objective Function..	84
4.1.3 Polyhedral Inner-Approximation Method for Area Constraint Linearization.....	87
4.1.4 Heuristic Decomposition-Based Iterative Mathematical Programming Approach	90
4.2 Computational Study on UA-CBFLD	92
4.2.1 An Illustrative Example with Diverse Capability Overlap Cases	94
4.2.2 Application Study in a Real-Life Manufacturing System	103
CHAPTER 5-CONCLUSIONS AND FUTURE RESEARCHES	117
5.1 Summary of the Thesis and Concluding Remarks	117
5.2 Future Research Directions	119
REFERENCES.....	120
APPENDICES	131

LIST OF FIGURES

	Page
Figure 3.1 A schematic depiction of the proposed robust BCBDL problem under a mixed fuzzy-stochastic environment.....	34
Figure 3.2 Summary of the proposed hybrid chance-constrained stochastic program with a fuzzy resolution approach for the robust FS-BCBDL problem ...	46
Figure 3.3 a) No, b) low, c) medium, d) high, e) very high, and f) fully machine-capability overlap cases.....	47
Figure 3.4 Machine-location assignments & RE distributions under scenario-2 of the fuzzy-stochastic case-1 ($\alpha = 0$).....	55
Figure 3.5 RE distributions of the most risky robust layout under scenario-2 of the fuzzy-stochastic case-1 ($\alpha = 0$).....	55
Figure 3.6 Machine-location assignments & RE distributions under scenario-2 of the fuzzy-stochastic case-1 ($\alpha = 1$).....	56
Figure 3.7 RE distributions of the risk-free robust layout under scenario-2 of the fuzzy-stochastic case-1 ($\alpha = 1$).....	56
Figure 3.8 Machine-location assignments & RE distributions for medium overlap case in deterministic optimization.....	56
Figure 3.9 RE distributions of the generated layout for medium overlap case in deterministic optimization.....	57
Figure 3.10 The existing cellular layout of the manufacturing facility with part assignments	63
Figure 3.11 Machine-location assignments and the RE distributions for the most risky case under scenario#4 ($\alpha = 0$)	69
Figure 3.12 RE distribution tables of the robust layout design for the most risky case under scenario#4 ($\alpha = 0$)	69
Figure 3.13 Machine-location assignments & RE distributions for the most risky case under scenario#2 ($\alpha = 0$)	70
Figure 3.14 RE distribution tables of the robust layout design for the most risky case under scenario#2 ($\alpha = 0$)	70
Figure 3.15 Machine-location assignments and the RE distributions for the risk-free case under scenario#4 ($\alpha = 1$)	71

Figure 3.16 RE distribution tables of the robust layout design for the risk-free case under scenario#4 ($\alpha = 1$)	72
Figure 3.17 Machine-location assignments and the RE distributions for the risk-free case under scenario#2 ($\alpha = 1$)	72
Figure 3.18 RE distribution tables of the robust layout design for the risk-free case under scenario#2 ($\alpha = 1$)	73
Figure 3.19 Comparison of the layout scores under different uncertainty levels	74
Figure 4.1 Depiction of the processing capability overlaps among different departments	76
Figure 4.2 The key components of the proposed UA-CBFLD problem.....	79
Figure 4.3 Secant lines of the inner-approximation technique for the non-convex area linearization (Xie et al., 2016).....	88
Figure 4.4 Schematic depiction of the heuristic decomposition-based iterative mathematical programming approach.....	93
Figure 4.5 Several capability overlap situations for the machines located in the departments	95
Figure 4.6 Unequal-area facility layout designs under distinct capability overlap situations.....	100
Figure 4.7 The existing equal-area cellular layout of the facility with product-cell assignments	104
Figure 4.8 The best capability-based UA-FLD options by the proposed MINLP model with or without constraint linearization	110
Figure 4.9 The existing cellular facility layout of the company with equal or unequal machine areas	112
Figure 4.10 The proposed best capability-based UA-FLD via the heuristic decomposition-based approach	113

LIST OF TABLES

	Page
Table 1.1 Framework of the thesis.....	6
Table 2.1 Summary of the literature review on the robust facility layout design problems under uncertainty.....	16
Table 2. 2 Summary of the literature review on different variants of the unequal-area facility layout problem	28
Table 3.1 Mathematical nomenclature of the MIP model.....	33
Table 3.2 Machining capabilities that are defined in terms of the REs	45
Table 3.3 Fuzzy demand quantities and manufacturing routes of the parts in terms of REs	47
Table 3.4 Fuzzy transition rates (or uncertain part flow patterns) between different REs	48
Table 3.5 Unit part flow costs among fixed locations	48
Table 3.6 Comparative results for the deterministic, fuzzy, stochastic and fuzzy-stochastic optimization cases (Sample size = 5; machine availabilities = 90%; probability for breakdowns = 10%; probabilities for the chance-constraint sets = 80% and 90%, respectively).....	50
Table 3.7 Machine availabilities under different scenarios and the resulted constraint satisfactions for medium overlap case	54
Table 3.8 Detailed calculation of the part flow-based layout objective in deterministic optimization.....	57
Table 3.9 Calculation of the part flow-based layout objective under scenario-2 of the fuzzy-stochastic case-1 ($\alpha = 0$).....	58
Table 3.10 Calculation of the part flow-based objective under scenarios-1, 3 and 4 of the fuzzy- stochastic case-1 ($\alpha = 0$).....	59
Table 3.11 Computational analysis on the fuzzy-stochastic optimization cases (Sample size = 5; machine availabilities = 85%; probability for breakdowns = 15%; probabilities for the chance-constraint sets = 90%)	60
Table 3.12 Computational analysis on the fuzzy-stochastic optimization cases (Sample size = 5; machine availabilities = 85%; probability for breakdowns = 15%; probabilities for the chance-constraint sets = 60%)	61

Table 3.13 Fuzzy demand quantities and process flow information of different parts in terms of REs	64
Table 3.14 Processing capabilities in terms of the REs and the number of available copies of the machine tools (Baykasoğlu and Gindy 2000).....	65
Table 3.15 Machine availabilities under different scenarios and the resulted constraint satisfactions	68
Table 3.16 Comparison of the present cellular layout with the proposed FS-BCBDL design method	74
Table 4.1 Mathematical nomenclature for the MINLP model of the UA-CBFLD problem.....	80
Table 4.2 Main steps of the heuristic decomposition-based iterative mathematical programming approach	92
Table 4.3 Manufacturing routes, processing times and demand quantities of the products	96
Table 4.4 Product flows between different REs and their total processing requirements (in min).....	96
Table 4.5 Minimum/maximum side lengths of the departments	96
Table 4.6 The slopes and intercepts of the secant lines for the illustrative example .	97
Table 4.7 Comparative results provided by the proposed MINLP model and the heuristic decomposition-based approach	97
Table 4.8 Optimization details of the proposed heuristic decomposition-based solution approach	99
Table 4.9 Product flows among various departments and their capacity usages under distinct overlap situations.....	101
Table 4.10 Details of product flows between departments under medium capability overlap case	102
Table 4.11 REs utilized by the departments under high capability overlap situation	103
Table 4.12 The number of machine copies with their capabilities in terms of the REs (Baykasoğlu and Gindy, 2000).....	105
Table 4.13 Demands quantities, process plan and processing times of the products	106

Table 4. 14 Product flows among distinct REs with their total processing necessities	106
Table 4.15 Minimum/maximum side lengths of departments where the machines are located	108
Table 4.16 The slopes of the secant lines for the real-life case study	109
Table 4.17 The intercepts of the secant lines for the real-life case study	109
Table 4.18 Optimization results of the proposed heuristic decomposition-based solution approach for the real-life case study.....	111
Table 4.19 Summary of the comparative results for the existing cellular layout and the proposed layout design options	111
Table 4.20 Comparison of the required and actual machine areas with error rates for the inner-approximation method	114
Table 4.21 The resulting product flows (units) between different machines for the real- life case study	114
Table 4.22 The REs used by different machines and their worst-case capacity utilizations	115
Table 4.23 Details of the product flows among distinct machines for the real-life case study	115

CHAPTER 1

INTRODUCTION

1.1 Background and Research Motivations

Capability-based distributed layout design approach was first introduced by Baykasoğlu (2003) in order to adapt dynamically changing manufacturing environments. In that original approach, machining capabilities which are described in terms of Resource Elements (REs) were taken into account as the basic capability units and proper distribution of these REs was targeted to provide distributed machine layouts. When compared to the classical machine based distributed layout design approaches (where optimal distribution of machines considered only over the factory floor), capability based distribution was proven to be more effective option as it can also reveal the hidden flexibility in the manufacturing systems (Baykasoğlu and Göçken 2010, Baykasoğlu and Subulan 2020). Because, some processing capabilities can be covered by alternative machines and therefore, capability-based description for the processing requirements of the manufactured parts may offer more flexibility than a machine-based route description where a part processing requirement is commonly allocated to a single machine. Moreover, in most of the real-world settings, manufacturing routes are frequently described in terms of production operations (or machine's processing capabilities/REs). Due to these facts, several other studies have also been published in the recent literature on the capability-based distributed layout design approach (Hamedi et al. 2012a, b; Hamedi and Esmaeilian 2015; Shafigh et al. 2015; Shafigh et al. 2017). However, it should be noted here that none of these studies considered the demand and process flow data while distributing machining capabilities over the factory floor. On the other hand, some dominant part flow patterns may arise among different REs and this may also affect optimal capability distribution. It was also suggested by Benjaafar and Sheikhzadeh (2000) and Benjaafar et al. (2002) that, if available, demand and process flow data may lead to more effective distributed layouts. Based on this motivation, Baykasoğlu and Subulan (2020) have extended the previous studies (Baykasoğlu 2003; Baykasoğlu and Göçken 2010) by taking into account the demand and process flow data while generating capability based distributed layouts.

Consequently, dominant process flow patterns among different capabilities were taken into account while designing effective distributed layouts. Furthermore, Baykasoğlu and Subulan (2020) also developed a mixed-integer linear programming (MILP) formulation of this problem, which was named as biased CB-DL design. However, uncertainties in demand data, process flow information and the machine reliabilities/availabilities were not taken into account in (Baykasoğlu and Subulan 2020). Nevertheless, these uncertainties are important for designing robust distributed layouts and may affect the performance of the facilities considerably.

In the first part of this thesis, the recent study of Baykasoğlu and Subulan (2020) is extended by considering uncertainties in machine reliabilities, product demand and process flow quantities. Because, under highly volatile manufacturing environments, the aforementioned biased CB-DL problem may also involve several uncertainties in machine reliabilities, product demand and process flow quantities. Actually, machine reliability, which corresponds to the random machine breakdowns (or machine unavailability risks) may considerably affect ease of access for the machining capabilities and may have also influence on the part flows among machines. Furthermore, flow patterns between different REs, which can be obtained from the processing routes of several parts may not be determined precisely due to fuzzy demand quantities and process flow information. Alternatively, these flow patterns among REs can also be derived as closeness relationships based on linguistic evaluations by facility planners according to their knowledge and expertise. For all of these reasons, most of the real-life biased CB-DL problems may contain different types of uncertainties simultaneously and therefore, they should be addressed together to provide a more reliable and robust layout design option. Therefore, a robust FS-BCBDL problem with stochastic machine reliability and fuzzy demand/process flow data is addressed in the first part of this thesis.

Recently, a new capability-based machine layout technique which extends the traditional QAP-based machine layout model was developed by Baykasoğlu et al. (2022) with a matheuristic-based solution methodology. However, all of the previous studies (Baykasoğlu, 2003; Hamed et al., 2012a & 2012b; Shafigh et al., 2015; Shafigh et al., 2017; Pourvaziri et al., 2021; Baykasoğlu and Göçken 2010, Baykasoğlu

and Subulan 2020; Baykasoğlu et al., 2022) on capability-based facility layout design approaches have considered equal machine or department areas. In other words, they aimed to generate capability-based layouts where the machines are assigned to proper facility locations with equal sizes. However, machines' area requirements cannot be always equal to each other and may differ in sizes for most of the real-life manufacturing environments. Moreover, none of the former studies on different variants of the classical unequal-area facility layout design problem (UA-FLD) have considered machining capabilities as well as their total processing requirements.

Based on this motivation, in the second part of this thesis, the recent study of Baykasoğlu et al. (2022) is extended by incorporating the machines' unequal area requirements into the capability-based machine layout problem. In other words, a new variant of the traditional UA-FLD problem is introduced, namely UA-CBFLD which can consider the proper dispersal of these processing capabilities over the facility-floor. In contrast to the conventional UA-FLD approaches, product flows among the machines are defined as decision variables (instead of input data) in the proposed UA-CBFLD. Because, manufacturing routes of the products are generally described based on their processing requirements (in terms of the REs) in most of the real-life applications. This also enables the facility planners to use existing flexibility in the production systems more effectually.

1.2 Contributions and Framework of the Thesis

As mentioned before, this thesis consists of two different problems. Therefore, the contributions of this thesis are divided into two parts for the FS-BCBDL and UA-CBFLD problems, respectively. Furthermore, the framework of this thesis is also presented in Table 1.1.

For the FS-BCBDL problem in Chapter 3, the following contributions are made:

- A new fuzzy-stochastic mathematical programming model is formulated for the biased CB-DL problem, which is mainly based on the original deterministic MILP model that was developed by Baykasoğlu & Subulan (2020). Apart from that original research, random machine availabilities and fuzzy demand/process flow data are also taken into consideration to

generate more reliable and robust layout alternatives. In other words, fuzzy demand data and therefore, fuzzy flow patterns among different REs are considered together with the random machine availabilities.

- In order to solve the proposed fuzzy-stochastic biased CB-DL problem, a hybrid transformation approach, which depends on a chance-constrained stochastic program and an interactive fuzzy resolution method is devised. To do this, some special functions and commands that are available in LINGO 19.0 optimization software are utilized to define the machine availabilities as discrete random variables, their best fit statistical distribution and chance constraint sets. Thereafter, stochastic components of the stated problem can be automatically transformed into the deterministic equivalent form and then several solutions (layout design options) are acquired by making use of the stochastic programming solver of LINGO 19.0 under various scenarios.
- In addition to the stochastic parameters, i.e., random machine availabilities, a very well-known interactive resolution method of Jimenez (2007) is also embedded into this stochastic program in order to transform the fuzzy constraints including uncertain part flow patterns into their crisp equivalent form. In fact, the proposed fuzzy-stochastic mathematical programming model can be converted into its crisp equivalent form easily with the help of this hybrid transformation approach and many α -acceptable solutions (or distributed layout design alternatives) can be produced under different uncertainty levels and various scenarios.

To the best of our knowledge, there is no similar study in the literature on such a robust CB-DL technique under mixed fuzzy-stochastic uncertainties. In fact, this is the key contribution of the first part of this thesis to the literature on robust FLD problems.

For the UA-CBFLD problem in Chapter 4, the following contributions are made:

- A novel unequal-area capability-based facility layout design (UA-CBFLD) approach is introduced for the first time in the literature. Actually, the proposed UA-CBFLD approach integrates two sub-problems, namely the classical UA-FLD problem and a new capability-based product flow

problem, which can consider machines' capacity utilizations and the total processing requirements of the machining capabilities.

- Thereafter, a new mixed-integer non-linear programming (MINLP) model of the UA-CBFLD problem is developed based on the recent study of Baykasoğlu et al. (2022) by considering machines' processing capability overlaps as well as their unequal-area requirements. In the proposed MINLP model, there is no interconnection between the constraints of the aforementioned sub-problems. In other words, there is no interaction between the decision variables of these two sub-problems and therefore, it is not possible to formulate any logical or technical constraint for the relationship between these two sub-problems. On the other hand, these sub-problems are just connected with each other in the objective function. In detail, objective function involves multiplication of the product flows and distances between the machines. However, both of the product flows and distances between machines are the decision variables of the proposed MINLP model and this makes the objective function highly non-linear. Indeed, product flows between machines can be provided by solving the first sub-problem, i.e., capability-based product flow problem whereas the distances between machine pairs can be obtained from the solution of the conventional UA-FLD problem. Thus, the proposed UA-CBFLD problem has a decomposable structure.
- Due to the NP-hard nature and highly non-linear model structure of the proposed UA-CBFLD problem, a heuristic decomposition-based iterative mathematical programming approach is implemented as an effective and efficient solution methodology for solving the realistic size problem instances within reasonable computing times. The superiority of this solution approach is also judged by comparing it with the presently developed MINLP model as well as its linearized version. Furthermore, its best found solution is also compared with the performance of the existing cellular facility layout in a real-life manufacturing case study.

To the best of our knowledge, there is no similar research in the literature that considers machines' processing capabilities as well as their unequal area requirements.

Table 1.1 Framework of this thesis

Chapter	Chapter 3	Chapter 4
Problem Definition	Fuzzy-stochastic biased capability-based distributed layout design (FS-BCBDL) problem	Unequal-area capability-based facility layout design (UA-CBFLD) problem
Solution Methodology	Chance-constrained stochastic programming and a well-known interactive fuzzy resolution method	Mixed-integer non-linear programming (MINLP) model, a polyhedral inner-approximation method and a heuristic decomposition-based iterative mathematical programming approach
Fuzzy Parameters	Demand/process flow data	N.A.
Stochastic Parameters	Machine availability/reliability	N.A.
Objective	Generate robust capability-based distributed layout design alternatives	Generate capacity-based machine layout options with unequal area limitations

1.3 Outline of the Thesis

This thesis consists of totally 5 chapters and the first chapter is the introduction part of this thesis. In the first chapter, background and research motivations are first presented. Then, the contributions, framework, outline and produced publications based on this thesis are also given.

In Chapter 2, an extensive literature surveys on both robust facility layout design problems under uncertainty and unequal-area facility layout problems are presented, respectively. Firstly, a literature review on FLD approaches with machine unavailability risks (or breakdowns), uncertain demand and process flow data is given. Then, a literature review on different variants of the UA-FLD problems is also given since the UA-FLD problem is still a hot research topic in the literature for many years.

In Chapter 3, problem description and mathematical programming formulation of the FS-CBDL design are given. First, the proposed fuzzy-stochastic optimization model is presented. Then, the fundamentals of the proposed hybrid approach are also discussed in detail. The proposed approach is first illustrated on a basic numerical example in Section 3.2.1. In that section, comparison of the deterministic, stochastic, fuzzy and fuzzy-stochastic optimization results also presented by considering different overlapping cases based on the machine-capability information. In Section 3.2.2, performance of the proposed approach is also tested on a real-life application.

In Chapter 4, a detailed problem description and mathematical formulation of the UA-CBFLD problem are first presented. Sections 4.1.2 and 4.1.3 of this chapter also contains some linearization procedures and an inner-approximation method that are used to linearize some constraints and objective function. In Section 4.1.4, a heuristic decomposition-based iterative mathematical programming approach is presented as an efficient solution methodology. An extensive computational study is performed in Sections 4.2.1 and 4.2.2 for both a hypothetical example as well as a real-life case study in a manufacturing company.

Chapter 5 presents the summary of this thesis, concluding remarks and future research directions.

1.4 Produced Publications Based on Research in the Scope of This Thesis

The following scientific research papers have produced throughout the preparation of this thesis. These papers are either published by proceeding books of some conferences or are under review status of some international journals. These papers were also assigned to certain thesis chapters:

Chapter 3:

- Subulan, K., Varol, B., & Baykasoğlu, A. (2022). Designing robust capability-based distributed machine layouts with random machine availability and fuzzy demand/process flow information, (Submitted to *Soft Computing Journal*, under review status).

- Subulan, K., Varol, B., & Baykasoğlu, A. (2020). Capability-based distributed layout design with random machine availability and fuzzy demand/process flow data. *IES'20 International Engineering Symposium*, Izmir, Turkey.
- Subulan, K., & Varol, B. (2021) Multi-objective fuzzy-stochastic goal programming model for flexible job shop scheduling with sequence-dependent setup times. *5th International Conference on Computational Mathematics and Engineering Sciences*, Van, Turkey.
- Varol, B., Subulan, K., Taşan, A.S. (2021). A fuzzy-stochastic mathematical programming approach for an integrated lot-sizing and supplier selection problem with quantity discounts and backordering options. *5th International Conference on Computational Mathematics and Engineering Sciences*, Van, Turkey.

Chapter 4:

- Subulan K., Varol, B., & Baykasoğlu, A. (2022). Unequal-area capability-based facility layout design problem with a heuristic decomposition-based iterative mathematical programming approach, (Submitted to *Expert Systems with Applications Journal*, under review status).
- Varol, B., Subulan, K., Baykasoğlu, A. (2022). Capability-based facility layout design with unequal-area departments. *12th Annual International Conference on Industrial Engineering & Operations Management*, Istanbul, Turkey.

CHAPTER 2

LITERATURE REVIEW

2.1 Literature Review on Robust Facility Layout Design Problems under Uncertainty

In this section, a brief literature review on robust FLD problems under uncertainty is given. Actually, there are two main sources of uncertainties in FLD problems, namely (i) uncertain product demand and/or process flow data (ii) machine unavailability risks (or reliability) due to the long-term breakdowns. Although there are many studies on the robust FLD problems with fuzzy and/or stochastic demand and process flow data, there is still a lack of researches which consider the machine reliability. It should also be noted here that these uncertainties have to be handled simultaneously to generate robust layouts. In the first part of this literature survey, the robust FLD in which uncertain demand and process flow data are considered. Afterwards, the articles that focused on the long-term machine breakdowns, which correspond to machine unavailability risks (or machine reliability) are reviewed. Because, machine unavailability risk may have a considerable effect on the layout objectives. Furthermore, articles which focused on uncertainties other than demand (or process flow) and machine availability are also reviewed in the last part of this survey.

An overview of the latest articles which focused on the uncertain product demand and/or process flow data can be summarized as follows: One of the pioneering study was performed by Rosenblatt and Lee (1987) on a robustness approach for a single period plant layout problem under stochastic demand data. They targeted to generate robust layout alternatives that perform well under many different scenarios. In a similar way, Aiello and Enea (2001) also proposed a fuzzy approach to the robust FLP where the market demand was represented by triangular fuzzy numbers. By making use of a constrained arithmetic operator and a fuzzy ranking method, they aimed to obtain a robust layout that minimizes the fuzzy material handling cost. Ertay et al. (2006) introduced a robust layout framework, which is based on the integration of data envelopment analysis and analytic hierarchy process techniques. They also adopted a computer-aided layout-planning tool, i.e., VisFactory in order to facilitate alternative

layout design process and minimize total material handling costs. Furthermore, they handled demand variability by means of fuzzy linguistic variables while data gathering for this tool (especially for the activity relationships). Kulturel- Konak (2007) discussed the most recent advancements and approaches to model and analyze uncertainties in FLD problems. She has presented a detailed review and classification for all of the former researches in this field. She also highlighted that inclusion of the uncertainty in FLD models is very essential for their applicability in real-life settings. This is because nowadays, business economies become even more volatile and product life cycles shorten continuously. Therefore, configuration of robust and flexible facility layouts under such a dynamic and uncertain environment is vital for both manufacturing and service industries. Moslemipour et al. (2012) also presented a detailed review of intelligent approaches along with their advantages and disadvantages for providing dynamic and robust layouts in flexible manufacturing environments. A simulation-based approach was developed by Jithavech and Krishnan (2010) for a risk-based FLP under stochastic product demand. They first utilized simulation method for risk evaluation and prediction of the uncertainty and then, a genetic algorithm (GA) was designed to produce the layout alternatives for both forecasted demand case and the risk-based layout design. Pillai et al. (2011) proposed a simulated annealing (SA) based metaheuristic algorithm to design a robust layout for a dynamic plant layout problem under dynamic demand environment. In fact, they proposed a robust layout procedure to cope with this dynamic environment and produce layout alternatives with minimum material handling and relocation costs for an expected flow matrix. A GA-based evolutionary fuzzy approach was developed by Drira et al. (2013) for solving a dynamic layout problem with increasing uncertainty in customer demands over time. The production demand of each part were defined by using triangular fuzzy numbers. They also concluded that degradation of the available information on customer demands may significantly impact the facility layout and therefore, demand uncertainty have to be considered if a robust layout is required. A mixed-integer non-linear programming model was formulated by Izadinia et al. (2014) for a robust multi-floor layout problem (MFLP) in which material flows among the departments and storages were changed in the pre-specified intervals. Then, they also linearized that model and solved it by using GAMS optimization software to determine

the locations of each department, storage, and elevator without overlapping in the floors of a multi-floor building. Vitayasak and Pongcharoen (2015b) aimed to analyze the impact of demand variation over time periods on a robust machine layout problem in which total material handling distance was minimized by a GA-based metaheuristic approach. Comparison of the re-layout and robust layout approaches was also carried out by Vitayasak and Pongcharoen (2015a) for a non-identical machine layout problem under stochastic demand. A greedy heuristic procedure was developed by Zhao and Wallace (2015) for solving a single-product capacitated FLP problem which aims to minimize total expected material handling cost under uncertain demand. Actually, they assumed that the product demand varies according to a known continuous probability distribution from period to period. By making use of this heuristic approach, stochastic FLP was converted into a traditional quadratic assignment problem (QAP). Celik et al. (2016) proposed a dynamic programming-based heuristic procedure for solving a multi-period stochastic distributed FLP, which aims to determine relative locations of the multiple copies of the capacitated machinery as well as the material flows among them. Dynamic and stochastic natures of demand and the possibility of re-layouts are also considered while minimizing the expected total material handling cost throughout the planning horizon. An integrated permutation-based GA and robust optimization technique was introduced by Fazlelahi et al. (2016) for the dynamic facility layout problem with demand uncertainty. Actually, they aimed to produce a unique robust layout plan for all of the time periods. A position-based flexible PSO algorithm was developed by Zha et al. (2017) to solve a robust dynamic facility layout problem with unequal-area departments. They took into account the uncertain product demands, which are stated as fuzzy-random variables. Moslemipour et al. (2017) proposed a novel mathematical programming model based on the QAP formulation for designing a dynamic robust facility layout. They assumed that the periodic product demands are normally distributed with known parameters, i.e., mean, standard deviation and covariance which vary randomly from one period to another. A hybrid algorithm which integrates the SA algorithm, ant colony, clonal selection, and robust layout design approaches was employed by Moslemipour et al. (2018) to solve a stochastic dynamic FLD problem in which product demands were defined as normally distributed random variables. Peng et al. (2018) proposed a new mathematical programming

model, which also considers transport device assignment for the stochastic dynamic facility layout problem. They also developed an improved adaptive GA to solve the proposed model and optimize the total material handling and rearrangement costs. In order to generate multiple demand scenarios randomly, they utilized Monte Carlo simulation approach. A scenario-based stochastic optimization model was proposed by Şahinkoç & Bilge (2018) depending on the classical QAP formulation in which part flows between the departments were stated as uncertain. After transforming this scenario-based uncertainty into a multi-objective optimization model, they developed a multi-objective evolutionary algorithm to find Pareto optimal set regarding different robustness performance measures. A robust optimization approach was presented by Xiao et al. (2019) for an unequal-area dynamic FLP with demand and logistics uncertainties. They proposed an improved PSO algorithm to solve this model within a reasonable computing time. In order to deal with dynamic changes in product demand and mix, Pourvaziri et al. (2021) developed an approach, which decomposes the robust facility layout design problem into two sub-problems. In the first sub-problem, they constructed a robust layout using a design-of-experiment (DOE) approach and then, a branch-and-cut algorithm was applied to solve the second sub-problem to obtain the best routes of the products in each time period. A hybrid approach based on Nested Partitions (NP) and SA algorithms was designed by Khajemahalle et al. (2021) for the robust dynamic facility layout problem. They assumed that the material flows between departments and rearrangement costs are uncertain parameters, whose values varied in the symmetric range of intervals.

Although several papers are available in the literature on the FLD problems with uncertain demand and/or material flows, the number of studies focusing on the uncertainty in machine unavailability/reliability (or breakdowns) is very limited. The recent articles that considered uncertain machine availability/reliability and breakdown maintenance are summarized as follows: Vitayasak and Pongcharoen (2016) stated that machine breakdown is a stochastic or random event, which is an essential concern for designing facility layouts in manufacturing industries. Therefore, they presented a GA application for quantifying the effects of breakdown maintenance on the machine layouts. They assumed that an alternative machine can be assigned for the manufacturing operations, if a machine is broken down. This also affects the

material handling distance since the route of machine sequence is changed. Similarly, in the scope of this thesis, it is also aimed to provide a robust capability-based distributed layout by assigning alternative machines, which have the same processing capabilities (or REs) as well as not broken down. It was also proven by Vitayasak and Pongcharoen (2016) that robust layout designs may cause lower cost in breakdown maintenance conditions when compared to the re-layout design approach. That study was also extended by Vitayasak et al. (2019) so as to incorporate dynamic demand and machine maintenance planning options (i.e., preventive and corrective maintenance and both of them) for designing non-identical machine layouts. Again, they implemented GA approach to analyze the effects of these maintenance strategies under different demand distributions. They concluded that designing robust machine layouts with preventive maintenance strategy leads to the greatest reduction in material flow distances. Based on this research motivation, this thesis proposes a robust CB-DL approach which is also able to specify critical machines with the most important processing capabilities. Actually, we used the unsatisfied chance constraint sets to determine these critical machines. Thus, more reliable and robust distributed layouts can be obtained by pre-specifying these critical machines, which necessitate preventive maintenance plan.

The FLD studies focusing on the other uncertainty issues rather than the demand (or process flow) and machine reliability (or breakdowns) are summarized as follows: Meller and Gau (1996) investigated the adjacency and distance-based as well as a weighted- criteria facility layout objectives to design robust layouts. Since specifying the certain weights may not be an exact process, they employed discrete efficient frontier method with imperfect weighting-value information. An exact algorithm based on a dynamic adaptive-iterative procedure was developed by Neghabi et al. (2014) for solving a robust multi-row FLP where the size of each department is stated as uncertain. In detail, permissible intervals are determined for the length and width of each department. In other words, they are considered as bounded variables and therefore, two new parameters are introduced to apply a robust optimization approach. Nematian (2014) introduced a robust single row facility layout model with fuzzy-random variables. The cost of transmitting flows between departments, the length of departments and the minimum gap between departments were represented by LR fuzzy

numbers. After transforming that robust model into its crisp equivalent form, a new exact algorithm based on an extended branch-and-bound algorithm was applied to achieve optimal solutions for several test problems. A bi-objective MIP model was developed by Salmani et al. (2015) for solving a robust multi-row facility layout problem with dynamic and uncertain dimensions of departments where each dimension changes in a pre-determined interval. In addition to minimize the total layout area, maximizing the sum of adjacencies among the departments was also targeted. Morinaga et al. (2017) introduced a robust facility layout planning method by considering temporal efficiency and routing. For enabling robustness, they assumed that sum of the products varies over production scenario with known probabilities. Thus, an additional evaluation index has been specified to obtain facility layouts considering numerous scenarios. They applied simulation optimization approach with GA to perform production scheduling within a reasonable time since this additional index causes significant increases in the computational effort. In order to reduce this computational load, which was resulted from the consideration of numerous production scenarios, Morinaga et al. (2019) also developed a new method based on a sampling approach and GA. In that method, individuals are evaluated based on the mean fitness in some scenarios that are sampled randomly and the selection process is performed by using Welch's test. Therefore, with the help of this improved GA approach, each layout alternative was evaluated with only a limited number of production scenarios in the optimization process. Unlike the existing studies in the literature, Targhi et al. (2019) developed a mathematical programming model for a robust three-dimensional and multi-floor facility layout problem. Since the parameters of this problem can be changeable, they utilized robust planning in overlapping constraints. Hunagund et al. (2020) also developed a MILP model for designing a robust unequal-area facility layout with flexible bay structure under dynamic environment. They applied an efficient SA algorithm and conducted several numerical experiments to show suitability of the proposed robust approach.

Most of these previously discussed articles are also summarized in Table 2.1 in relation to the problem characteristics (Objectives, equal or unequal department areas, type of uncertainty: fuzziness and/or stochasticity, uncertain components/parameters) and the solution methodology. As it can be seen in Table 2.1, the majority of robust

FLD approaches concentrated on the uncertainties of product demand or process flow. However, there is still a few studies in the existing literature on the FLD approaches with random machine unavailability/reliability issues (or breakdowns). Actually, this is an important research gap in the literature on the robust FLD problems. Moreover, none of the prior studies handled the machining capabilities that are described by the REs. Fortunately, consideration of these processing capabilities may also provide additional flexibility in case of machine breakdowns. In other words, alternative machines that are not broken down (or in active status) and have the same processing capabilities can be used instead of the broken or inactive machines. However, it may not be always possible to find a feasible layout under some scenarios where some of the machines are not in active status. In this case, critical machines which necessitate the preventive maintenance actions can also be determined easily by making use of the unsatisfied scenarios in the proposed chance-constrained stochastic programming model.

Finally, it should also be noted here that although several robust FLD approaches are available in the literature, there is still a lack of papers on the robust distributed layout design techniques under uncertainty. In other words, robust distributed layouts were rarely discussed in the current literature. Furthermore, only one type of uncertainty, i.e., either fuzziness or stochasticity (randomness) was considered by the majority of the existing studies while handling nondeterministic layout parameters. However, different types of uncertainties may arise at the same time in most of the real- life facility layout problems. For that reason, these different uncertainty types should be considered together while generating robust facility layouts. Based on these facts, this thesis introduces a hybrid approach based on a chance-constrained stochastic programming and interactive fuzzy resolution method for solving a robust capability-based distributed machine layout problem with stochastic machine reliability and fuzzy demand/process flow inputs. As mentioned before, to the best of our knowledge, there is no similar research on such a robust CB-DL approach under mixed fuzzy and stochastic types of uncertainties.

Table 2.1 Summary of the literature review on the robust facility layout design problems under uncertainty

Article	Problem type	Problem characteristics							Objectives	Solution methodology
		Type of area		Uncertainty		Uncertain components				
		Equal	Unequal	Fuzzy	Stochastic	Demand	Machine Breakdowns	Other		
Rosenblatt and Lee (1987)	Single period plant layout problem	✓			✓	✓			Minimum total material handling costs	Robustness approach
Meller & Gau (1996)	Robust FLP	✓			✓			✓	Minimum total material handling costs	Robust layout method & discrete efficient frontier method
Aiello & Enea (2001)	Robust FLP	✓		✓		✓			Minimum total material handling costs	Constrained arithmetic operator & fuzzy ranking method
Ertay et al. (2006)	Robust FLP	✓		✓		✓			Minimum total material handling costs	DEA/AHP methodology with the VisFactory tool
Jithavech & Krishnan (2010)	Risk-based stochastic FLP	✓			✓	✓			Minimum maximum increased cost associated with risky flow intensities	Simulation-based approach & GA
Pillai et al. (2011)	Dynamic plant layout problem	✓			✓	✓			Minimum total material handling & relocation costs	SA based meta-heuristic
Drira et al. (2013)	Fuzzy dynamic layout problem	✓		✓		✓			Minimum sum of transportation costs	Fuzzy evolutionary algorithm
Izadinia et al. (2014)	Robust multi-floor layout problem	✓			✓	✓			Minimum total material handling costs	MILP approach with linearization technique
Neghabi et al. (2014)	Robust multi-row FLP	✓			✓			✓	Minimum total cost	A two-phase adaptive & iterative algorithm
Nematian (2014)	Fuzzy robust single row facility layout problem	✓		✓	✓			✓	Minimum cost of transmitting all flow between departments	Fuzzy stochastic chance-constrained programming based proposed exact algorithm
Salmani et al. (2015)	Robust multi-row FLP		✓		✓			✓	Maximum number of adjacencies while minimum total layout area	MIP model
Vitayasak & Pongcharoen (2015b)	Robust machine layout design problem	✓			✓	✓			Minimum total material handling distance	GA
Vitayasak & Pongcharoen (2015a)	Non-identical machine layout problem	✓			✓	✓			Minimum total material handling costs	GA
Zhao & Wallace (2015)	Single-product capacitated distributed layout problem	✓			✓	✓			Minimum total expected material handling cost	A greedy heuristic algorithm
Çelik et al. (2016)	Stochastic distributed FLP	✓			✓	✓			Minimum total expected flow distance	Dynamic programming based heuristic procedure
Vitayasak & Pongcharoen (2016)	Robust machine layout problem	✓			✓	✓	✓		Minimum total material handling distance	GA

Table 2.1 Continues

Article	Problem type	Problem characteristics							Objectives	Solution methodology
		Type of area		Uncertainty		Uncertain components				
		Equal	Unequal	Fuzzy	Stochastic	Demand	Machine Breakdowns	Other		
Fazlelahi et al. (2016)	Robust dynamic single FLP	✓			✓	✓			Minimum total cost	Integration of permutation-based GA and robust optimization techniques
Zha et. al. (2017)	Dynamic FLP with unequal area departments		✓	✓		✓			Minimum total material handling costs	Position-based flexible particle swarm optimization algorithm
Moslemipour et al. (2017)	Stochastic dynamic FLP	✓			✓	✓			Minimum total material handling costs	QAP-based mathematical programming model
Morinaga et al. (2017)	Robust FLP	✓			✓			✓	Minimum makespan for the obtained layouts	Conventional FLP method considering temporal efficiency
Moslemipour et al. (2018)	Stochastic dynamic facility layout problem	✓			✓	✓			Minimum total cost	Hybrid AC-CS-SA algorithm
Peng et al. (2018)	Robust facility layout approach for stochastic dynamic FLPs	✓			✓	✓			Minimum total material handling cost and rearrangement cost	Improved GA based robust facility layout approach & Monte Carlo simulation method
Xiao et al. (2019)	Unequal-area dynamic FLP		✓		✓	✓			Minimum total material handling cost and total cost of wasted area	Robust optimization & an improved PSO algorithm
Vitayasak et al. (2019)	Robust machine layout problem	✓			✓	✓	✓		Minimum material flow distance	GA
Targhi et al. (2019)	Robust three-dimensional and multi-floor FLP		✓		✓			✓	Minimum distances between departments	Mathematical programming approach
Hunagund et al. (2020)	Robust unequal area dynamic FLP		✓		✓	✓		✓	Minimum total material handling costs	SA algorithm
Pourvaziri et al. (2021)	Robust FLD for flexible manufacturing		✓		✓	✓			Minimum total material handling costs	Design of experiments and a hybrid genetic-tabu search metaheuristic algorithm
Khajemahalle et al. (2021)	Robust dynamic FLP	✓			✓	✓		✓	Minimum sum of the material handling costs and the rearrangement costs	Hybrid algorithm based on nested partitions and SA algorithm & heuristic algorithm
The present research	Robust capability-based distributed layout problem	✓		✓	✓	✓	✓		Minimum total product flow rate between locations	Hybrid chance-constrained program & interactive fuzzy resolution method

2.2 Literature Survey on the Unequal-Area Facility Layout Design Problems

A comprehensive literature review on the UA-FLPs between the years of 2010-2022 is presented in this section to show the motivation of this thesis. The reviewed articles are also summarized in the Table 2.2. In the literature, there are basically two types of solution methodologies, namely, the exact and approximate methods. Actually, the exact methods mainly include mathematical programming approaches whereas the approximation methods are based on heuristic and/or metaheuristic algorithms. In most of the available studies in the literature, the UA-FLPs are usually solved by metaheuristic algorithms due to their highly nonlinear structure and NP-completeness nature. However, the UA-FLPs can also be solved by an exact method by applying some linearization techniques. In addition, in the literature there are three main types of UA-FLPs available which are classical UA-FLPs, multi-objective UA-FLPs and dynamic UA-FLPs. Studies that handled these problem types and some additional studies are reviewed in this section.

An overview of the recent literature review on the classical UA-FLPs can be summarized as follows: Ant system (AS) which is the variation of ant colony optimization (ACO) was first applied to solve the UA-FLPs by Wong (2010). A slicing tree structure was employed for solution representation and they also applied heuristic information with AS. They aimed to find a better facility layout by minimizing both material flow cost between departments as well as an adaptive penalty function which is related to the number of infeasible departments that solution involves. According to the computational experiments on several test instances, AS is able to produce superior results when there is no empty space in the shop floor. A convex- optimization-based framework was designated by Jankovits et al. (2011) for solving the UA-FLP. The proposed framework integrates two mathematical programming models to provide the convex relaxation for establishing the relative position of the departments within the facility and semidefinite optimization to determine the final layout. Kulturel-Konak & Konak (2011a) implemented a hybrid particle swarm optimization (PSO) and a local search approach for solving the UA-FLP with a flexible bay structure. Kulturel-Konak & Konak (2011b) developed an ant colony optimization (ACO) approach for solving the UA-FLP with the flexible bay structure. The objective function was stated as the

minimum material transfer cost. The proposed approach was tested on various problem instances from the literature and it was proved that the proposed approach is effective in finding the best solutions in reasonable computational times. Hernandez et al. (2011) addressed the UA-FLP by considering a qualitative criterion as an objective function. Actually, they aimed to maximize the satisfaction of the decision-maker's interests. Furthermore, they presented an Interactive Genetic Algorithm (IGA) which allows the preferences of the decision-maker. According to the computational results, IGA is capable to handle decision-maker preferences and can give good quality results in reasonable computation times. Aiello et al. (2012) solved the UA-FLP by using a multi objective genetic algorithm (MOGA) based upon a slicing structure. They handled four conflicting objectives namely, minimize material transfer cost, maximize satisfaction of weighted adjacency, satisfaction of distance requests and satisfaction of aspect ratio requests via Pareto based evolutionary approach. Ulutas & Kulturel-Konak (2012) proposed an artificial immune system (AIS)-based algorithm for solving the UA-FLP with flexible bay structure. They also proposed a new encoding and a clonal selection algorithm (CSA). They proved the superiority of the proposed algorithm by testing it on various benchmark instances. The objective was to minimize the total material transfer cost as well as a penalty cost for infeasible departments. Kulturel-Konak (2012) developed a probabilistic Tabu Search (PTS) algorithm for solving the UA-FLP with relaxed-flexible bay structure (RFBS) in which empty spaces are also allowed. They also proposed a LP model to determine the locations and shapes of the departments and it was embedded in to the PTS algorithm to achieve better solutions. Bozer & Wang (2012) proposed a heuristic solution procedure, which uses graph-pair representation technique. In the search procedure of the proposed heuristic procedure, they also applied SA algorithm. An Interactive Genetic Algorithm (IGA) which can incorporate the decision maker's knowledge was presented by Garcia-Hernandez et al. (2013) to solve UA-FLP. By making use of the IGA, decision maker's knowledge could be considered and therefore, an interaction between the algorithm and decision maker was also provided. The aim of that study was to satisfy the decision maker's preferences while designing the facility layout. Kulturel-Konak & Konak (2013) also applied a hybrid GA approach, which also combines an LP model. In the proposed

approach, LP is integrated into the encoding of GA which means that solutions obtained from the LP model are used to generate new solutions by the GA.

Ulutas & Kulturel-Konak (2013) presented a clonal selection algorithm (CSA) for solving the UA-FLP with flexible bay structure. They aimed to minimize a penalized cost function. The computational results have shown that better quality solutions could be provided within less computing times for large-sized benchmark instances. Gonçalves & Resende (2015) developed a biased random-key genetic algorithm (BRKGA) for solving the UA-FLP. The aim of that study was to minimize the total flow cost while placing the rectangular departments into a rectangular facility without any overlap. Garcia-Hernandez et al. (2015) developed a new hybrid evolutionary algorithm which guides the search process based on the decision maker's preferences while solving the UA-FLP. Therefore, human assessment could be integrated into the proposed algorithm's search and selection procedures. The goal of that study was to satisfy the decision maker's preferences. A novel approach which combines zone-linear programming (zone-LP) and simulated annealing (SA) algorithm was presented by Xiao et al. (2016) for solving the UA-FLPs. The efficiency of the proposed approach was proven by performing several computational experiments on many benchmark instances. An improved optimization-based framework was proposed by Anjos & Vieira (2016) to solve the UA-FLP. They integrated two mathematical models into the proposed framework, where the first model is a nonlinear approximation for evaluating the positions of the departments. By making use of the second model, final layout was generated by an exact convex optimization formulation. The objective function of the first model was formulated as a combination of total connectivity cost among departments and the penalty term for the overlapped departments. Computational results have shown that the proposed framework is efficient even for large-scale problem instances. Paes et al. (2017) developed a hybrid GA, which integrates a decomposition strategy via partial solution deconstructions/reconstructions method for solving the UA-FLP to minimize the sum of distances among facilities weighted by material handling flows. In addition, they compared the results of the basic GA with quadrant restrictions and decomposition phases (DRQGA). According to the results of several test instances, solution

improvements can be obtained through the proposed decomposition method when compared to the basic GA.

In order to avoid the premature convergence and high computational costs, Palomo-Romero et al. (2017) developed an Island Model Genetic Algorithm (IMGA) for solving the UA-FLP for the first time in the literature. They also tested the performance of this algorithm on a well-known data sets taken from the literature. Based on their computational study, they concluded that the IMGA can find better solutions and improve formerly best found results for most of the test instances. Kang & Chae (2017) implemented a harmony search algorithm with a slicing tree layout representation in order to solve the UA-FLP in which the total material flow cost between departments was aimed to minimize. After testing the performance of this algorithm on many well-known instances in the literature, it was realized that the proposed harmony search algorithm can generate better quality solutions within less computational time. Liu & Liu (2019) developed a multi objective ant colony optimization (MOACO) algorithm to solve the UA-FLP so as to optimize the material transfer costs and closeness rating scores simultaneously. They tested the performance of the proposed algorithm by comparing it to an evolutionary algorithm with/without local search, adaptive variable neighborhood search and MOACO algorithms with flexible bay structure. According to the computational experiments, it is proven that the UA- FLP can be solved by the proposed MOACO algorithm in a more effective way. Garcia-Hernandez et al. (2019) proposed coral reefs optimization (CRO) algorithm to solve the UA-FLP with flexible bay structure. They aimed to achieve minimum material flow between the locations. The superiority of the CRO algorithm was also tested on several benchmark instances. A two-step heuristic algorithm which utilizes construction and median methods simultaneously was developed by Park & Seo (2019) to solve the UA-FLP where the input and output points of the departments were also considered. According to the computational results, better quality solutions can be achieved by means of the proposed heuristic algorithm within much shorter computation times. Garcia-Hernandez et al. (2020a) developed a Coral Reefs Optimization algorithm to solve the UA-FLP with Substrate Layers (CRO-SL). In the proposed multi-method evolutionary algorithm, various search procedures were applied simultaneously in order to prevent local optimal solutions. Garcia-Hernandez et al. (2020d) proposed a novel approach

called Island Model based on Coral Reefs Optimization (IMCRO) algorithm for solving the UA-FLPs with flexible bay structure.

The proposed approach was tested on various test instances and the computational results have shown that the IMCRO algorithm has an excellent performance on all of the instances. Garcia-Hernandez et al. (2020b) implemented a novel hybrid metaheuristic approach that combines the Coral Reefs Optimization (CRO) with Variable Neighborhood Search (VNS) algorithm to solve the UA-FLPs. They utilized a Relaxed Flexible Bay Structure (RFBS) for layout representation and aimed to minimize material flow between facilities and a penalty value resulting from infeasible solutions. The performance of the proposed approach was also tested on various benchmark instances and it could be concluded that this approach has superior performance for all instances. Garcia-Hernandez et al. (2020c) presented a novel approach based on Interactive Coral Reefs Optimization (ICRO) algorithm for solving the UA-FLP. In the proposed approach, both quantitative and qualitative characteristics of the decision-maker were handled while formulating the fitness function. Indeed, the fitness function of the problem was formulated as minimal material flows between departments with a penalty function. Moreover, decision maker's preferences were also taken into consideration as mentioned before. Ahmadi-Javid & Ardestani-Jaafari (2021) designed a novel hybrid solution approach based on memetic and simulated annealing algorithms for solving the UA-FLP whose objective function was formulated as a minimum weighted loop-based distance. In the stated problem, material handling operations were performed by automated guided vehicles (AGV), and these AGVs move along a single-loop path. The computational results of this hybrid approach have shown that the proposed hybrid approach can produce superior results. In order to obtain a common and collaborative layout design, Salas-Morera et al. (2021) addressed the UA-FLP by considering qualitative preferences of the multiple decision-makers. They developed a Coral Reef Optimization (CRO) algorithm for solving large scaled problem instances where the total material handling cost was aimed to minimize. Finally, it was proved that promising facility designs that fulfill all of the preferences of decision-makers could be obtained. A SA-based solution algorithm was presented by Hunagund et al. (2021) for solving the UA-FLP with flexible bay structure where total material transfer cost between the departments

was minimized. It was also proven that the proposed solution approach is competitive in terms of the computational time when compared to other metaheuristic algorithms.

A monarch butterfly optimization algorithm was developed by Kim & Chae (2021) to solve the UA-FLP that aims to minimize the total weighted distance among the departments. According to the comprehensive experimental study, the proposed algorithm is able to generate better results for larger sized test instances within much shorter computation times. Guo et al. (2022) developed a modified non-dominated sorting genetic algorithm (NSGA-II) for solving the UA-FLP of an air-conditioner production shop floor. In addition to minimize the material transfer cost, they aimed to optimize the closeness rating score. The computational results of the modified NSGA-II were compared with the results of two related algorithms and it was concluded that the proposed algorithm outperforms the other algorithms for large-scaled test instances.

Dynamic FLPs with unequal-area departments are also one of the main problem types of the UA-FLPs. Therefore, the studies that solved dynamic UA-FLPs are summarized as follows: McKendall Jr & Hakobyan (2010) addressed the dynamic facility layout problem (DFLP) with unequal department areas where the material flows may change during the planning horizon. They developed a metaheuristic solution procedure which combines the boundary search (BS) and Tabu Search (TS) algorithms. They targeted to minimize the sum of material transfer and rearrangement costs. Liu et al. (2017) developed a hybrid solution approach which combines Wang–Landau (WL) sampling and some heuristic algorithms for the dynamic UA-FLP. Minimizing the sum of product flow and rearrangement costs was formulated as an objective function. Furthermore, the proposed approach has been proven to be effective by solving various benchmark instances. In order to solve the dynamic UA-FLPs within reasonable computing times, Kulturel-Konak (2017) designated a matheuristic solution approach which integrates Tabu Search (TS) algorithm and a mathematical programming model. Total material handling and relay costs were stated as the main aim of the present problem. The computational experiments have displayed that the proposed matheuristic solution approach can provide promising results when solving realistic size dynamic UA-FLPs. Zha et al. (2017) also addressed

the robust UA-FLP under a fuzzy random environment where the uncertainty of product demand was considered. They also developed a position-based flexible PSO algorithm which aims to minimize the material handling costs between departments.

Hunagund et al. (2018) applied the SA algorithm to solve an unequal-area dynamic FLP with flexible bay structure. The minimization of the material transfer and fixed/variable rearrangement costs was targeted. Xiao et al. (2019) developed an improved PSO algorithm to solve the dynamic UA-FLP which considers the locations of pick-up and drop-off points. They also presented a real-life application in an assembly line to demonstrate the effectiveness of the developed algorithm. Ghadirpour et al. (2020) also handled the unequal-area stochastic dynamic facility layout problem (UA-SDFLP) by also considering the routing flexibility. They also taken into account the stochastic product demands while optimizing the expected material transfer and rearrangement costs. To do this, three mixed-integer nonlinear programming (MINLP) models were formulated for the UA-SDFLP and tested on small-sized benchmark instances. Additionally, GA was also developed for solving larger sized test instances. A hybrid solution approach depending on the PSO and SA algorithms was introduced by Zha et al. (2020) to solve the UA-DFLP where the uncertain demands were stated as fuzzy-random variables. The proposed hybrid approach was also tested on a real-life case study to show its effectiveness and efficiency. Hunagund et al. (2020) applied a SA algorithm for solving the dynamic UA-FLP with flexible bay structure whose objective is to minimize the material transfer costs. The computational results have shown that good quality solutions can be produced via the proposed algorithm when compared to the existing solution methodologies in the literature. Zolfi & Jouzdani (2022) applied a SA algorithm to solve the unequal-area multi-floor dynamic facility layout problem (MF-DFLP) with flexible bay structure. They targeted to minimize total cost which is composed of material handling cost, rearrangement fixed/variable costs, as well as the installation, rearrangement and maintenance costs of the elevators.

In the literature, the following studies handled the multi-objective UA-FLPs: Ripon et al. (2011) proposed a multi-objective genetic algorithm (MOGA) to acquire a set of Pareto-optimal solutions of a multi-objective UA-FLP that aims to optimize two objectives (i.e., minimum material transfer cost and maximum closeness rating score)

simultaneously. Ripon et al. (2013) proposed an adaptive variable neighborhood search (VNS)-based evolutionary approach for solving a multi-objective unequal-area facility layout problem (UA-FLP). Aiello et al. (2013) applied a non-dominated ranking genetic algorithm (NRGA) to find Pareto-optimal solutions of a multi objective UA-FLP with four conflicting objectives.

The objectives were to minimize material transfer cost, minimize aspect ratio, and maximize distance and adjacency, respectively. Again, they used the slicing-tree layout representation. After applying this Pareto evolutionary procedure, the optimal solution was determined by using Electre method. With the help of this multi-criteria decision making procedure, decision maker's opinions could be taken into account in the candidate solution set. Liu et al. (2018) addressed a multi-objective UA-FLP to optimize some conflicting objectives namely, minimize material transfer cost, and maximize the total adjacency value as well as utilization rate of the facility. They proposed a multi-objective particle swarm optimization algorithm with an objective space division method (MOPSO-OSD) for obtaining a set of Pareto-optimal solutions. They also compared the results of their proposed algorithm with a multi-objective particle swarm optimization algorithm including the crowding distance method (MOPSO-CD) and non-dominated sorting genetic algorithm II (NSGA- II). Moreover, they tested their proposed algorithm on three data sets from the literature and proved that their proposed algorithm is able to produce superior results. Hou et al. also (2019) developed a GA-based solution method with a slicing tree representation for solving the multi-objective UA-FLP where the goals were stated as minimizing the material handling cost and maximizing the closeness scores. The proposed solution method was tested on several problem instances and it was proved that it is able to provide superior results in shorter computation times. Liu et al. (2020a) proposed a local search heuristic procedure for solving a multi-objective UA-FLP. They also applied niche technology to find Pareto-optimal solutions in which minimum material transfer cost and maximum closeness rating are handled as objective functions. A multi-objective evolutionary approach called the configuration space evolutionary (CSE) algorithm was designed by Liu et al. (2020b) for solving a multi-objective UA-FLP with the flexible bay structure. The objectives of the problem were determined as minimum material transfer cost, maximum satisfaction of weighted closeness relationship,

maximum distance requirements, and maximum satisfaction of the aspect ratio. Bhuiyan et al. (2021) formulated a new multi-objective mixed-integer nonlinear programming model of the UA-FLP where objectives were considered as minimum material flow cost, maximum closeness rating score and minimum makespan.

They also applied a reformulation-based spatial branch-and-bound algorithm to obtain a set of Pareto-optimal solutions so that facility planners could select a proper layout in real-life cases. An adapted Genetic Algorithm with placement (GA-P) was proposed by Halawa et al. (2021) to solve an outpatient clinic design problem by formulating an UA-FLP where the objectives were to maximize natural daylight exposure and minimize total walking distance for patients. They also formulated a multi-objective non-linear programming model of the stated problem. The computational results have demonstrated that the GA-P algorithm outperforms the GA-C, PSO, and SA-P. Tang et al. (2022) addressed a multi-objective UA-FLP whose objectives were to minimize material transfer cost and maximize the adjacent value and utilization rate of the floor shop. They also applied the PSO algorithm which also combines the objective space division method (OSD). Comparative study has shown that the objective function values could be improved by using the proposed method when compared to the available methods in the literature.

Some other studies related to different variants of the UA-FLPs are also summarized as follows: Asl et al. (2016) addressed an unequal-area stochastic facility layout problem where the product demands are stated as stochastic parameters. They also developed an improved covariance matrix adaptation evolutionary strategy (ICMA-ES) to minimize the total material transfer cost. The generated results by the proposed approach gave better results when compared to some metaheuristic algorithms such as PSO and GA. Unequal-area static facility layout problem (UA-SFLP) was addressed by Derakhshan & Wong (2017) by using a modified PSO algorithm so as to minimize the material transfer cost. They also formulated two mixed-integer nonlinear programming (MINLP) models of the stated problem. The modified PSO was tested on many benchmark instances and the best found solutions were compared to the existing results in the literature. Ingole & Singh (2017) applied a firefly algorithm (FA) to solve the unequal-area fixed-shape facility layout problem

where the total material handling and transportation costs between departments were minimized. After testing the performance of FA on large-sized benchmark instances, they demonstrated that the proposed algorithm yielded the best results so far in the literature within less computational time.

To solve the UA-FLP in a continual planar site, Allahyari & Azad (2018) developed a mixed integer nonlinear programming (MINLP) model and applied the simulated annealing (SA) algorithm with a unique heuristic algorithm in its initialization phase. While constructing the problem structure, they also considered the aisle structure in the facility, block and preference constraints, usage of carrier for material handling operations, within-site boundaries and elimination of overlapping departments. Ingole & Singh (2021) proposed a biogeography-based optimization (BBO) algorithm for solving the UA-FLPs with fixed shape and flexible shape facilities. Similar to the majority of existing researches in the literature, minimizing the total material flow cost among departments was formulated as the objective function. After testing performance of the proposed algorithm on some benchmark instances, it was proven that the proposed algorithm can yield better solutions when compared to the available solution methodologies in the literature. Hunagund et al. (2022) proposed a two-stage solution approach for solving the UA-FLP with a bi-directional relaxed flexible bay structure (BRFBS). In the first stage of the solution approach, the present problem was formulated as a quadratic assignment problem (QAP) and solved with SA algorithm. Then, a heuristic method was also applied in the second stage to minimize the total material handling cost.

Although there are many papers available in the literature on different variants of the UA-FLD problem, none of the previously mentioned researches have considered machines' processing capabilities as well as their total processing requirements. Based on this motivation, this thesis presents a novel unequal-area capability-based facility layout design approach in Chapter 4.

Table 2. 2 Summary of the literature review on different variants of the unequal-area facility layout problem

Article	Problem Type	Aim of the problem	Characteristic of methodology		Solution methodology
			Exact	Heuristic/Metaheuristic	
Wong (2010)	UA-FLP	Minimum total cost and adaptive penalty function		√	AS
McKendall Jr & Hakobyan (2010)	Dynamic UA-FLP	Minimum material handling and rearrangement costs		√	TS/BSH
Jankovits et al. (2011)	UA-FLP	Minimum cost with penalty components		√	Convex-optimization-based framework
Kulturel-Konak & Konak (2011a)	UA-FLP	Minimum material handling cost		√	PSO
Ripon et al. (2011)	Multi-objective UA-FLP	Minimum material handling cost and maximum closeness rating		√	MOGA
Kulturel-Konak & Konak (2011b)	UA-FLP with the flexible bay structure	Minimum material handling cost		√	ACO
Hernandez et al. (2011)		Maximum satisfaction of the decision-maker's interests		√	ICA
Aiello et al. (2012)	UA-FLP	Minimum material handling cost, maximum satisfaction of weighted adjacency, maximum satisfaction of distance requests and maximum satisfaction of aspect ratio requests		√	MOGA
Ulutas & Kulturel-Konak (2012)	UA-FLP	Minimum total material handling cost among departments and a penalty cost for infeasible departments		√	AIS algorithm
Kulturel-Konak (2012)	UA-FLP	Minimum material flow	√	√	LP embedded PTS-RFBS approach
Bozer & Wang (2012)	UA-FLP	Minimum cost		√	Heuristic with graph-pair procedure & SA
Garcia-Hernandez et al. (2013)	UA-FLP	Satisfy decision maker's preferences		√	IGA
Ripon et al. (2013)	Multi-objective UA-FLP	Minimum material handling cost and maximum closeness rating score		√	Adaptive VNS-based evolutionary approach
Kulturel-Konak & Konak (2013)	UA-FLP	Minimum cost		√	GA/LP approach
Aiello et al. (2013)	Multi-objective UA-FLP	Minimum material handling cost, minimum aspect ratio, maximum distance and maximum adjacency		√	MOGA, NGR & Electre
Ulutas & Kulturel-Konak (2013)	UA-FLP	Minimum penalized cost function		√	CSA

Table 2.2 Continues

Article	Problem Type	Aim of the problem	Characteristic of methodology		Solution methodology
			Exact	Heuristic/Metaheuristic	
Gonçalves & Resende (2015)	UA-FLP	Minimum total cost		√	BRKGA
Garcia-Hernandez et al. (2015)	UA-FLP	Satisfy decision maker's preferences		√	Hybrid evolutionary approach
Xiao et al. (2016)	UA-FLP	Minimum total material handling cost		√	Combined zone-LP and SA algorithm
Anjos & Vieira (2016)	UA-FLP	Minimum total connectivity cost between departments, and the penalty term for overlap	√	√	An improved optimization-based framework
Paes et al. (2017)	UA-FLP	Minimum sum of distances among facilities weighted by material handling flows		√	DRQGA
Palomo-Romero et al. (2017)	UA-FLP	Minimum total material handling cost		√	IMGA
Derakhshan & Wong (2017)	UA-SFLP	Minimum material handling costs		√	Modified PSO algorithm
Kang & Chae (2017)	UA-FLP	Minimum material handling costs		√	HS-based heuristic algorithm
Ingole & Singh (2017)	Unequal-area, fixed-shape facility layout problem	Minimum total cost of material handling and transportation between departments		√	FA
Liu et al. (2017)	Dynamic UA-FLP	Minimum the sum of the material handling and rearrangement costs		√	WL sampling algorithm and heuristics
Kulturel-Konak (2017)	Dynamic UA-FLP	Minimum total of material handling and relayout costs		√	Matheuristic approach that combines TS and mathematical modeling approach
Zha et al. (2017)	Dynamic UA-FLP under fuzzy random environment	Minimum material handling cost		√	Position-based flexible PSO algorithm
Liu et al. (2018)	Multi-objective UA-FLP	Optimize the material handling cost, the total adjacency value and the utilization ratio of the shop floor		√	MOPSO-OSD
Allahyari & Azad (2018)	UA-FLP in a continual planar site	Minimum total material handling cost	√	√	MINLP and SA algorithm with an unique heuristic algorithm

Table 2.2 Continues

Article	Problem Type	Objective of the problem	Characteristic of the solution methodology		Solution methodology
			Exact	Heuristic/Metaheuristic	
Hunagund et al. (2018)	Dynamic UA-FLP	Minimum material handling cost, variable rearrangement cost, and fixed rearrangement cost		√	SA
Liu & Liu (2019)	UA-FLP	Optimum material handling cost and closeness rating score		√	MOACO algorithm
García-Hernández et al. (2019)	UA-FLP	Minimum material flow		√	CRO algorithm
Hou et al. (2019)	Multi-objective UA-FLP	Minimum material handling cost and maximum closeness score		√	GA
Xiao et al. (2019)	Dynamic UA-FLP	Minimum material handling cost		√	PSO algorithm
Park, & Seo (2019)	UA-FLP	Minimum material handling cost		√	Two-heuristic algorithm
García-Hernández et al. (2020a)	UA-FLP	Minimum material flow, number of unfeasible facilities and difference between minimum fitness value from all feasible solutions and best fitness value found overall		√	CRO-SL
García-Hernández et al. (2020d)	UA-FLP	Minimum material flow		√	IMCRO
Liu et al. (2020a)	Multi-objective UA-FLP	Minimum material handling cost and maximum closeness rating		√	Local search based on the Pareto optimization & niche technology
Liu et al. (2020b)	Multi-objective UA-FLP with the flexible bay structure	Minimum material handling costs, maximum satisfaction of weighted closeness relationship, maximum distance requirements, and maximum satisfaction of aspect ratio of facilities		√	CSE
García-Hernández et al. (2020b)	UA-FLP	Minimum material flow between facilities and a penalty function		√	CRO algorithm & VNS
García-Hernández et al. (2020c)	UA-FLP	Minimum material flow between facilities, and penalty function; the preferences of the DM		√	ICRO algorithm

Table 2.2 Continues

Article	Problem Type	Aim of the problem	Characteristic of methodology		Solution methodology
			Exact	Heuristic/Metaheuristic	
Ghadirpour et al. (2020)	UA-SDFLP	Minimum material handling and rearrangement costs	√	√	MINLP & GA
Zha et al. (2020)	Dynamic UA-FLP under fuzzy random environment	Minimum material handling and rearrangement costs		√	Hybrid PSO and SA
Hunagund et al. (2020)	Dynamic UA-FLP with FBS	Minimum material handling cost		√	SA
Ahmadi-Javid & Ardestani-Jaafari (2021)	UA-FLP	Minimum weighted loop-based distance		√	Hybrid algorithm based on Memetic and Simulated Annealing
Bhuiyan et al. (2021)	Multi-objective UA-FLP	Minimum material transportation cost, maximum closeness rating score, and minimum makespan of the facility	√	√	Reformulation-based spatial branch-and-bound algorithm
Halawa et al. (2021)	Multi-objective UA-FLP	Maximum natural daylight exposure and minimum total walking distance for patients		√	GA-P
Ingole & Singh (2021)	UA-FLP with fixed shape and flexible shape facilities	Minimum total material flow		√	BBO algorithm
Salas-Morera et al. (2021)	UA-FLP	Minimum material handling cost		√	CRO
Hunagund et al. (2021)	UA-FLP	Minimum material handling cost		√	SA
Kim & Chae (2021)	UA-FLP	Minimum total weighted distance between the departments		√	Monarch butterfly optimization algorithm
Guo et al. (2022)	UA-FLP	Minimum material handling cost and maximum closeness rating score		√	Modified NSGA-II
Zolfi & Jouzdani (2022)	Unequal-area MF-DFLP	Minimum total material handling costs, rearrangement fixed cost, rearrangement variable costs, rearrangement cost of the elevators, maintenance cost of the elevators, and installation costs of the elevators		√	SA
Tang et al. (2022)	Multi-objective UA-FLP	Minimum material handling cost, maximum adjacent value and maximum utilization rate of floor shop		√	PSO algorithm combining OSD
Hunagund et al. (2022)	UA-FLP with BRFBs	Minimum material handling cost		√	Two-staged heuristic solution approach
The present research	Unequal-area capability-based facility layout design problem	Minimum total material flows between locations	√	√	MINLP model and a heuristic decomposition-based iterative mathematical programming approach

CHAPTER 3

DESIGNING ROBUST CAPABILITY-BASED DISTRIBUTED MACHINE LAYOUTS WITH RANDOM MACHINE AVAILABILITY AND FUZZY DEMAND/PROCESS FLOW INFORMATION

3.1 Problem Description and Mathematical Formulation of the Robust FS- BCBDL Problem

The unbiased capability-based distributed layout (UBCB-DL) problem, which doesn't consider the demand and process flow data was first introduced by Baykasoğlu (2003) under deterministic environments. Actually, the main aim of the UBCB-DL is to minimize the total distances from each unoccupied factory location by a RE to the other location where the RE is accessible. In other words, each machining capability can be reached from each factory location within minimum distance. Afterwards, Baykasoğlu and Subulan (2020) extended the UBCB-DL by incorporating the product demand and process flow data of the manufactured parts and introduced a new problem, namely biased capability-based distributed layout (BCB-DL) design. Unlike the UBCB-DL problem, BCB-DL problem aims to minimize both of the total part flow rates and total distances from the unoccupied locations by the REs to the occupied locations. In other words, in addition to minimize the total distance covered, it is also targeted to achieve minimum part flow rates from the unoccupied locations by a machining capability to the locations where this capability is available. Therefore, all of the REs can be allocated over the facility floor by taking into account the transition rates between different REs. On the other hand, these transition rates, which are computed by using the demand quantities and the processing routes of parts were assumed to take deterministic values in Baykasoğlu and Subulan (2020). In fact, demand quantities may include some sort of ambiguity and therefore, uncertainty in these transition rates (or part flow rates) should be considered in order to provide a robust distributed layout. Furthermore, long-term probabilistic machine breakdowns which correspond to the machine unavailability risk are also important in designing robust facility layouts. For all of these reasons, a new robust BCB-DL problem under a mixed fuzzy-stochastic environment is introduced in this chapter.

Indeed, fuzzy demand/process flow information and random machine availability are considered simultaneously to obtain more reliable and robust distributed layouts. To do this, the original MILP formulation of the deterministic BCB-DL problem, which was formerly developed by Baykasoğlu and Subulan (2020) is first modified by incorporating the probabilistic machine availabilities and fuzzy transition rates that show the dominant part flow patterns between different REs. An overview of the proposed FS-BCBDL problem are also presented in Figure 3.1.

3.1.1 The Mixed-Integer Programming (MIP) Formulation

The modified MIP model for with the probabilistic and fuzzy constraints is formulated in Equations (3.1) – (3.10) through mathematical notation that is given in Table 3.1. The description of mathematical notation including model parameters and decision variables are also demonstrated in Figure 3.1.

Table 3.1 Mathematical nomenclature of the MIP model

<i>Indices & Sets</i>	
$(i, j) \in N$	Set of locations in the distributed layout
$k \in K$	Set of machines
$r \in R$	Set of Resource Elements (REs)
$m \in M$	Set of manufactured parts
Q	Set of parts that have transitions from RE t to r in their processing routes
<i>Parameters</i>	
d_{ij}	Rectilinear distance between locations i and j
a_{kr}	1, if machine k has processing capabilities including the RE r ; 0 otherwise
$\bar{D}_m$	Fuzzy demand quantity for part m
τ_{mtr}	Number of transitions from RE t to r in the process route of part m
$\bar{F}\bar{P}_{tr}$	Fuzzy transition rates/flow patterns between the RE t and r in the process routes of all parts
φ_k	Discrete random variable for the availability of machine k
p_k	Probability for the availability of machine k
θ	Satisfaction probability of the chance-constraint set in Equation (3.5)
β	Satisfaction probability of the chance-constraint set in Equation (3.6)
S	Scenario or sample size in the chance-constrained stochastic program
M	A huge positive number
<i>Decision variables</i>	
y_{ki}	1, if machine k is assigned to location i ; 0 otherwise
z_{irj}	1, if i is the closest location to location j for the unavailable RE r
f_{irj}	Part flow rate from location i to j for the unavailable RE r

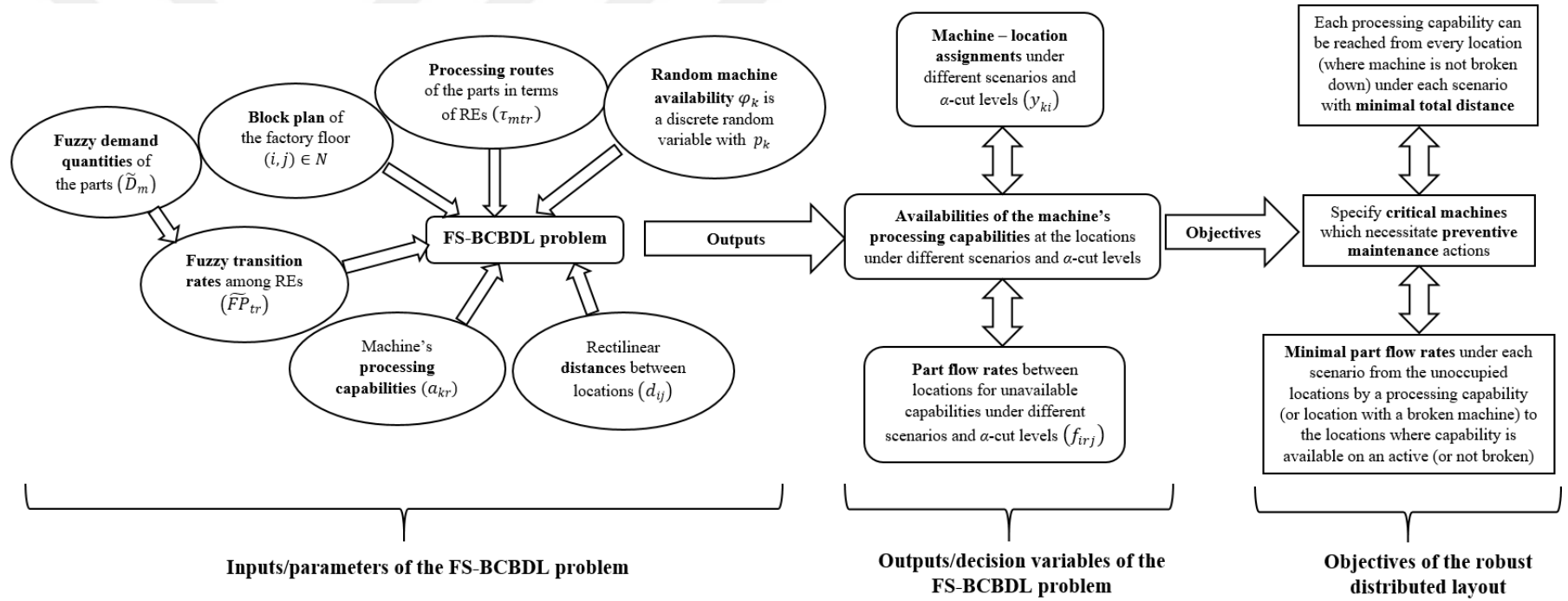


Figure 3.1 A schematic depiction of the proposed robust BCBDL problem under a mixed fuzzy-stochastic environment

As shown in Figure 3.1 and the objective function formulation in Equation (3.1), it is targeted to minimize the total part flow rates as well as total distance from the unoccupied locations by the processing capabilities (or the locations with a broken machine) to the locations where these capabilities are available on active (or not broken) machines. In detail, parts wherever in the factory floor can access to all of the needed REs from their existing locations within a minimum distance and material flow. Due to the unavailable REs and/or broken machines, total rectilinear distances and part flow rates between different locations are considered simultaneously in Equation (3.1). Therefore, all REs can be distributed appropriately over the factory floor by considering the fuzzy flow (or transition) rates among different REs and the machine unavailability risks (or machinery breakdown).

$$\text{Minimize } \sum_{i \in N} \sum_{j \in N \setminus \{i\}} \sum_{r \in R} (f_{irj} + Z_{irj}) \cdot d_{ij} \quad (3.1)$$

Subject to;

$$\sum_{i \in N} y_{ki} = 1 \quad \forall k \in K \quad (3.2)$$

$$\sum_{k \in K} y_{ki} = 1 \quad \forall i \in N \quad (3.3)$$

$$\sum_{j \in N \setminus \{i\}} Z_{irj} \leq (|N| - 1) \cdot \left(1 - \sum_{j \in N \setminus \{i\}} Z_{jri} \right) \quad \forall i \in N \setminus \{j\}, \forall r \in R \quad (3.4)$$

$$\Pr \left\{ \sum_{j \in N \setminus \{i\}} Z_{jri} \geq 1 - \sum_{k \in K} a_{kr} \cdot y_{ki} \cdot \varphi_k \right\} \geq \theta \quad \forall i \in N, \forall r \in R \quad (3.5)$$

$$\Pr \left\{ \sum_{k \in K} \sum_{t \in R \setminus \{r\}} \widetilde{F}P_{tr} \cdot a_{kt} \cdot y_{ki} \cdot \varphi_k - \sum_{j \in N \setminus \{i\}} f_{irj} \leq M \cdot (1 - z_{jri}) \right\} \geq \beta \quad (3.6)$$

$$\forall i \in N, \forall r \in R$$

$$f_{irj} \leq M \cdot Z_{jri} \quad \forall i \in N, \forall j \in N \setminus \{i\}, \forall r \in R \quad (3.7)$$

$$f_{irj} \geq 0 \text{ and } continuous \quad \forall i \in N, \forall j \in N, \forall r \in R \quad (3.8)$$

$$y_{ki}, z_{ijr} \in \{0,1\} \quad \forall i \in N, \forall j \in N, \forall k \in K, \forall r \in R \quad (3.9)$$

$$\varphi_k \text{ is a discrete random variable with probability of } p_k \quad \forall k \in K \quad (3.10)$$

Indeed, there will be part flows from a location where a RE is unavailable to any other location where it is available to achieve the required machining capability. It should also be emphasized here that if the assigned machine to this location is broken down, part flows should be realized by choosing another occupied location with an active machine including that RE. By the way, there may also be part flows from an occupied location by a RE to another occupied one because of the machine breakdown. In addition to minimize the total part flow rates and total distances as formulated in Equation (3.1), the proposed robust BCB-DL approach also aims to predetermine the critical machines which require preventive maintenance actions. Actually, these critical machines can be obtained from the stochastic optimization results, which display the scenarios with unavailable or broken machines. If one or more machines are broken down in any scenario and they cause violation of the chance constraint sets in Equations (3.5) – (3.6), one or both of these probabilistic constraints will be unsatisfied in that scenario. Therefore, unsatisfied scenarios including machinery breakdowns will exhibit these critical machines. Because, we cannot allocate machining capabilities appropriately over the factory floor in case of these critical machines are broken down. In other words, there will be no feasible distributed layout alternative for these unsatisfied scenarios. Even if some machines are broken down, there may be no unsatisfied scenario in the stochastic optimization results. In this case, there will be no critical machines and therefore, machine unavailability risks may be insignificant. Actually, we may encounter such a situation when all of the REs or processing capabilities are covered by many alternative machines. Thus, when a machine including the needed RE is broken down, alternative machines can come into play. On the other hand, if all of the alternative machines including that RE are broken down, probabilistic chance constraints are not satisfied in that scenario. In a nutshell, overlapping capabilities into the binary machine – capability matrix (a_{kr}) also play a

significant role in the stochastic optimization results, which help us to determine these critical machines. To analyze the effect of these overlapping capabilities, we considered different machine-capability overlap cases (no, low, medium, high, very high and fully overlap cases) in the subsequent illustrative example section. Let us explain the constraints of the proposed fuzzy-stochastic MIP model for the robust BCB-DL problem. According to the constraint set in Equation (3.2), every machine should only be allocated to one location on the factory floor. Additionally, when a machine is allocated to a location, then all of its capabilities are also occupied in that location. It is also supposed in this thesis that each location on the factory floor has identical size that can locate any machine. Constraint set in Equation (3.3) guarantees that only one machine can be assigned to each location on the factory floor. Actually, these assignment constraints in Equations (3.2) – (3.3) aimed to choose the optimal machine locations. According to the constraint set in Equation (3.4), if location i is not occupied by the RE r whereas this RE is available at the nearby location j , a part flow from location i to j will be carried out to reach that RE. In this situation, the binary variable, i.e., z_{jri} will be equal to “1”, which means that there is a part flow from location i to j to access RE r . Briefly, the constraint set in Equation (3.4) maintains that when a part flow for RE r is required from location i to j , then the part flows for the same RE r can't be provided to that location i from any other location j . Moreover, this constraint also maintains that the part flows can be realized to this occupied location j by the RE r from the maximum number of $|N|-1$ unoccupied locations on the factory floor. In fact, $|N|-1$ is utilized in this formula to prevent the usage of a positive Big M value (Baykasoğlu and Subulan 2020).

The chance constraint set in Equation (3.5) guarantees that if a machine k , which doesn't include the RE r is allocated to location i , then this location needs to reach RE r with minimal distance. In detail, a part flow will arise from location i to the adjacent location j where RE r is available. It should be noted here that unavailability risk (or breakdown status) of machine k is already considered by this linear probabilistic restriction with stochastic right-hand side. Indeed, random machine unavailability is formulated by this chance-constraint set. Because, the machine availability/unavailability issues cannot be known with certainty in real-life dynamic manufacturing environments but may fit some probability distributions. Hence, it is

supposed that in a long-term planning horizon, machine k may be in active statue with a pre-specified probability of p_k and therefore, it may also be in inactive statue with an unavailability probability of $(1 - p_k)$. If machine k assigned to location i is inactive, then all of its processing capabilities/REs will disappear automatically from that location i . For that reason, a robust distributed layout design is generally desired by the facility planners that considers the machine unavailability risk or long-term machine breakdowns. Thus, availability of machine k , i.e., φ_k is defined as a discrete random variable with known probability. If machine k becomes active statue (or available), then φ_k will take the value of “1” with the probability of p_k . Otherwise, if it becomes inactive statue and then, this random variable will take the value of “0” with the probability of $(1 - p_k)$. corresponds to the machine reliability in the long-term planning horizon. With the help of this random machine unavailability risk in the chance constraint set of Equation (3.5), although the machine k which is allocated to location i and includes the RE r , it may not be in active statue because of the machinery breakdown. In this case, a part flow must be realized again from location i to the nearby location j for reaching that RE r . By making use of the chance constraint set in Equation (3.6), part flow rates among various factory locations can be computed for different REs. Indeed, these part flow rates can be computed as a result of the produced distributed layout design. In other words, their values are mainly based on the machine – location allocations. For that reason, if the RE r is not included by machine k assigned to location i , there will be part flows from that location i to the nearby location j containing that RE r . Thus, the amount of part flows should be larger than the total transition rates between available REs ($t \in R$) and unavailable REs r ($r \in R$ and $r \neq t$) at this location i . In Equation (3.6), the probabilistic machine unavailability risk is again considered since the machine k with RE t may become inactive statue and therefore, this location also needs to reach the RE t as well as RE r . As a result, a part flow from location i to its nearby location j is needed for both cases to reach the RE r . However, it should be emphasized here that there is an inaccessibility risk for that RE r from the unoccupied location i because of this chance- constrained stochastic programming formulation. In fact, these probabilistic constraint sets in Equations (3.5) – (3.6) may not be satisfied 100% of the time since the machine availabilities are represented by discrete random variables with known probabilities.

Instead, it is targeted to find a robust capability-based distributed layout that will fulfill these probabilistic constraints under a pre-determined percentage, i.e., 90 or 95 percent of the time. In other words, the generated distributed layout alternatives may be feasible for 95% or 90% of all cases for the random machine availabilities. In addition to these random machine unavailability, we may not know the exact flow patterns between different REs with certainty. Because, the parts' demand quantities may have some sort of ambiguity. Therefore, transition frequencies or flow rates among different REs, i.e., $\widetilde{FP}_{tr}$ are also considered as uncertain layout parameter in Equation (3.6) and represented by triangular fuzzy numbers. In order to calculate these fuzzy flow rates and construct the transition rate matrix, we formulated Equation (3.11), which primarily uses the fuzzy demand quantities and the pre-specified deterministic processing routes of several parts. According to this formula, the number of transitions among various REs should be first provided by the manufacturing routes of all the parts. Afterwards, the fuzzy transition rate matrix can be calculated as in Equation (3.11) via the fuzzy division operator. On the other hand, it should be highlighted here that this matrix may not be constant and may vary with respect to the dynamic demand fluctuations in the future. Moreover, extra processing operations (or routing changes) of newly introduced parts or finished production of some outdated parts may also affect this transition rate matrix. For all of these reasons, construction of such a fuzzy transition rate matrix may be more convenient to generate robust layout design options in dynamic and uncertain manufacturing environments.

$$\widetilde{FP}_{tr} = \frac{\sum_{m \in Q} \widetilde{D}_m \cdot \tau_{mtr}}{\sum_{m \in M} \sum_{t \in R} \sum_{r \in R \setminus \{t\}} \widetilde{D}_m \cdot \tau_{mtr}} \quad \forall (t, r) \in R \quad (3.11)$$

Alternatively, these fuzzy flow patterns (or transition rate matrix) can also be constructed by making use of linguistic evaluations of process experts or facility planners based on their knowledge and experiences. In other words, a relationship chart with linguistic evaluations, which states the intensity of part flows among many REs can also be constructed by the facility planners. For instance, according to the processing routes of many parts in terms of the REs, if the amounts of part flows are very large between REs r and t , then the linguistic term “very high” can be used in this relationship chart. Otherwise, other linguistic terms such as “moderate” or “low” can

also be utilized to state the less amount of part flows. Therefore, corresponding triangular fuzzy numbers of these linguistic evaluations can also be used as the closeness relationship values. Finally, constraint set in Equation (3.7) guarantees that if location j is not the nearby point to location i for achieving the RE r , no part flow between these locations are needed. To satisfy this statement, big M value is utilized again to avoid the redundant part flows. In Equation (3.8), the part flow rates which are computed by Equation (3.6) are represented by continuous decision variables. Furthermore, it should be noted here that sum of all the part flow rates should be equal to “1”. The constraint set in Equation (3.9) display the binary decision variables for the machine-location allocations and the part flows for unavailable REs among different locations of the factory floor. Lastly in Equation (3.10), the probabilistic machine availability is defined as a discrete random variable for each machine with known probability.

3.1.2 The Proposed Hybrid Chance-Constrained Stochastic Program with A Fuzzy Resolution Approach

In the previous section, a hybrid fuzzy-stochastic optimization model is formulated for a robust FS-BCBDL problem in order to better reflection of the real-life layout design cases. As mentioned before, the majority of real-world facility layout applications can include distinct uncertainty types simultaneously such as randomness, fuzziness and dynamism etc. On the other hand, most of the recent studies in this field considered these uncertainty categories independently. However, these different uncertainty types should be handled and modelled simultaneously. For that reason, this section presents a hybrid chance-constrained stochastic program with a fuzzy resolution approach. Actually, we employed the chance-constrained stochastic programming approach of Charnes and Cooper (1959) simultaneously with a well-known interactive fuzzy resolution method of Jimenez et al. (2007) so as to transform the proposed fuzzy-stochastic programming model into its deterministic equivalent form. Indeed, since the probabilistic constraint set in Equation (3.6) incorporates fuzzy and stochastic parameters instantaneously, usage of such a hybrid solution approach is necessary. Baykasoğlu and Subulan (2019) have already employed a similar hybrid solution procedure while dealing with a fuzzy-stochastic intermodal fleet management

problem. Similarly, a chance-constrained program is first applied in this thesis to handle the probabilistic constraint set in Equation (3.5) in which discrete random variables are defined only for the uncertain machine availabilities with known probability values.

In this thesis, we considered two different probability distribution functions for the random machine availabilities so as to analyze the effects of these distributions on the stochastic optimization results: (i) Basic discrete random variable with known probabilities and (ii) Binomially distributed random machine availabilities with one trial only ($n = 1$) and the success probabilities of p_k . Actually, this distribution case also corresponds to Bernoulli random variables because of using only one trial. This means that any machine can be broken down only once in the planning period or the facility planner desires to take only one sample from the manufacturing environment for the random machine availabilities. If the number of trials is increased, facility planner needs to collect more data related to the machine failures and breakdown statistics. When the basic discrete random variables with known probabilities are defined for the probabilistic machine availabilities, the chance constraint set in Equation (3.5) can be transformed into its deterministic equivalent non-linear form as in Equation (3.14) by using the critical values of the random machine availabilities. To do this, these critical values, i.e., ψ_{ir} can be specified by incorporating the probability density function in Equation (3.12) and inverse cumulative distribution function (see in Equation 3.13) of this discrete random variable, i.e., φ_k (Bisschop and Roelofs 2006; Gupta et al. 2003; Subulan 2020). Let us define the cumulative distribution function of this basic discrete random variable, i.e., $F(\psi_{ir}) = \zeta_{ir} \forall i \in N, \forall r \in R$, then the deterministic approximation of this probabilistic constraint set in Equation (3.5) can be formulated as in Equations (3.12) - (3.14) by making use of its inverse cumulative (or quantile) function, i.e., $F^{-1}(\zeta_{ir}|p_k)$.

$$f_x(\varphi_k) = \begin{cases} 1 & \text{if machine } k \text{ is in active statue with probability of } p_k \\ 0 & \text{if machine } k \text{ is broken down with probability of } (1 - p_k) \end{cases} \quad (3.12)$$

$\forall k \in K$

$$\psi_{ir} = F^{-1}(\zeta_{ir}|p_k) = \begin{cases} 1 & 0 < \varphi_k \leq 1 \\ 0 & \varphi_k \leq 0 \end{cases} \quad \forall k \in K \quad (3.13)$$

$$1 - \sum_{j \in N \setminus \{i\}} Z_{jri} \leq \sum_{k \in K} a_{kr} \cdot y_{ki} \cdot \psi_{ir} \quad \forall i \in N, \forall r \in R \quad (3.14)$$

On the other hand, when the binomially distributed random variables are defined for the probabilistic machine availabilities, these critical values, i.e., ψ_{ir} should be computed as in Equation (3.15) by utilizing Binomial distribution's inverse cumulative function.

$$\psi_{ir} = F^{-1}(\zeta_{ir} | w; n, p_k) = \sum_{q=1}^{\lfloor w \rfloor} \binom{n}{q} \cdot (p_k)^q \cdot (1 - p_k)^{n-q} \quad \forall i \in N, \forall r \in R \quad (3.15)$$

where w represents the number of successes (or active statue of machine k with probability of p_k) in $n = 1$ independent Bernoulli trial. In a similar manner, the chance constraint set in Equation (3.6) must also be converted into its crisp equivalent non-linear form. However, this chance constraint set involves not only probabilistic machine availabilities (φ_k), but also includes a fuzzy parameter, i.e., $\widetilde{FP}_{tr}$ that corresponds to the uncertain transition frequencies (or part flow rates) between different machining capabilities/REs. For that reason, α -parametric approach of Jimenez et al. (2007) is first applied to transform this fuzzy constraint into its crisp equivalent form. Afterwards, the chance-constraint stochastic programming method is implemented again to cope with the random machine availabilities. The fuzzy technological coefficient, i.e., $\widetilde{FP}_{tr}$ which is located on the left-hand side of this probabilistic constraint set in Equation (3.6) is represented by a triangular fuzzy number, i.e., $(FP_{tr}^p, FP_{tr}^m, FP_{tr}^o)$. Actually, the left, core and right margins of this triangular fuzzy number display the pessimistic, most likely and optimistic values of the uncertain part flow rates, respectively. The expected interval of this fuzzy parameter can be computed as in Equation (3.16).

$$EI(\widetilde{FP}_{tr}) = \left[\frac{1}{2} \cdot (FP_{tr}^p + FP_{tr}^m), \frac{1}{2} \cdot (FP_{tr}^m + FP_{tr}^o) \right] \quad (3.16)$$

In addition to the expected interval of the fuzzy data, an α -cut based fuzzy ranking procedure of Jimenez (1996) is also employed to carry out fuzzy to crisp conversion process. After completing this fuzzy to crisp transformation process, the chance constraint set in Equation (3.6) can be reformulated as in Equation (3.17):

$$\Pr \left\{ \sum_k \sum_{t \in R \setminus \{r\}} \left(\alpha \cdot \frac{FP_{tr}^p + FP_{tr}^m}{2} + (1 - \alpha) \cdot \frac{FP_{tr}^p + FP_{tr}^o}{2} \right) \cdot a_{kt} \cdot y_{ki} \cdot \varphi_k - \sum_{j \in N \setminus \{i\}} f_{irj} \leq M \cdot (1 - z_{jri}) \right\} \geq \beta \quad \forall i \in N, \forall r \in R \quad (3.17)$$

where α is a parameter which demonstrates the feasibility degree of constraint set in Equation (3.17). When the value of this parameter is set to '0', it corresponds to an unacceptable or the most risky solution. On the other hand, one can achieve a completely acceptable or risk-free solution when its value is equal to '1'. Through this parameter whose value ranged from '0' to '1', various layout design alternatives can be produced under different uncertainty levels. In other words, many α -acceptable layout design options can be found by taking into account the conflict between satisfaction degree of layout objectives and fulfilment degree of this fuzzy constraint. After performing the fuzzy to crisp transformation procedure, the chance constraint set in Equation (3.17) should also be transformed into its deterministic equivalent non-linear form as in Equation (3.14). Similar to the previous formulations in Equations (3.12) – (3.15), critical values of the random machine availabilities, i.e., ψ_{ir} in Equation (3.18) can be calculated by making use of its inverse cumulative distribution functions, i.e., $F^{-1}(\zeta_{ir} | p_k)$ or $F^{-1}(\zeta_{ir} | w; n, p_k)$.

$$\sum_{j \in N \setminus \{i\}} f_{irj} + M \cdot (1 - z_{jri}) \geq \sum_k \sum_{t \in R \setminus \{r\}} \left(\alpha \cdot \frac{FP_{tr}^p + FP_{tr}^m}{2} + (1 - \alpha) \cdot \frac{FP_{tr}^p + FP_{tr}^o}{2} \right) \cdot a_{kt} \cdot y_{ki} \cdot \psi_{ir} \quad (3.18)$$

$$\forall i \in N, \forall r \in R$$

Unfortunately, application of the aforementioned fuzzy-random to crisp transformation process may cause a highly non-linear deterministic equivalent MIP model. Moreover, solution of this kind of mathematical programs by using the standard MIP solvers is a challenging task. Because, several auxiliary variables and additional constraints are needed to carry out these transformation operations. As a result, the required computational effort for solution of this kind of mathematical programs will increase significantly with respect to the problem dimension. In order to overcome this problematic issue, we utilized some special commands and functions such as @SPSTGRNDV, @SPDISTBINO, @SPTABLESHAPE, @SPTABLEOUTC, @SPTABLEINST, @SPTABLERNDV, @SPSAMPsize, @SPchance etc. of LINGO 19.0 optimization software to construct the chance constraint sets in Equations (3.5) - (3.6) and used its powerful stochastic programming solver to solve the stated problem (LINDO Systems Inc. 2021). Actually, the stochastic programming solver of LINGO 19.0 which is able to apply Benders Decomposition and Genetic Algorithms is used to achieve high-quality robust solutions for the current fuzzy-stochastic BCB-DL problem. Summary of the proposed hybrid chance-constrained stochastic program with a fuzzy resolution approach is also depicted in Figure 3.2.

3.2 Computational Study

This section presents an illustrative example as well as a real-life case study in order to display validity and applicability of the proposed robust FS-BCBDL approach.

3.2.1 An Illustrative Example with Comparative Results

In this section, a trivial example is given to exemplify the proposed robust FS-BCBDL problem and investigate the impacts of different machine-capability overlap cases on the fuzzy-stochastic optimization results. This example problem is composed of 6-machines and 6-locations in a 2 x 3 block plan and the machine- processing capability data related to the REs is depicted in Table 3.2. Each machine has a set of operation capabilities that are described in terms of the REs and totally 6-REs are available. The processing capability information of machines given in Table 3.2 is also depicted in Figure 3.3c, which corresponds to the medium capability overlap case. Based on this case, five different machine-RE matrices are derived for no, low, high,

very high and fully overlap cases (see Figure 3.3a-f) in order to explore the impact of distinct machine-capability overlap cases.

Table 3.2 Machining capabilities that are defined in terms of the REs

Machine-1: RE1, RE2, RE3	Machine-4: RE3, RE4, RE6
Machine-2: RE1, RE2, RE5	Machine-5: RE4, RE5, RE6
Machine-3: RE1, RE3, RE4	Machine-6: RE2, RE5, RE6

It is also intended to investigate the effects of random machine breakdowns on the fuzzy-stochastic optimization results in case of different machine-capability overlaps. To do this, different probability values (85% and 90%) for the random machine availabilities are also tested for each machine-capability overlap case. Indeed, each machine will become in active statue with a pre-specified probability of p_k in the long-run planning horizon. Otherwise, any machine may breakdown and therefore, it will be in inactive statue with the probability of $(1 - p_k)$. Actually, this probability information for the machine availabilities can be seen as the most important parameter for both discrete random variables and binomial distribution as it is formulated in Equations (3.12) and (3.15).

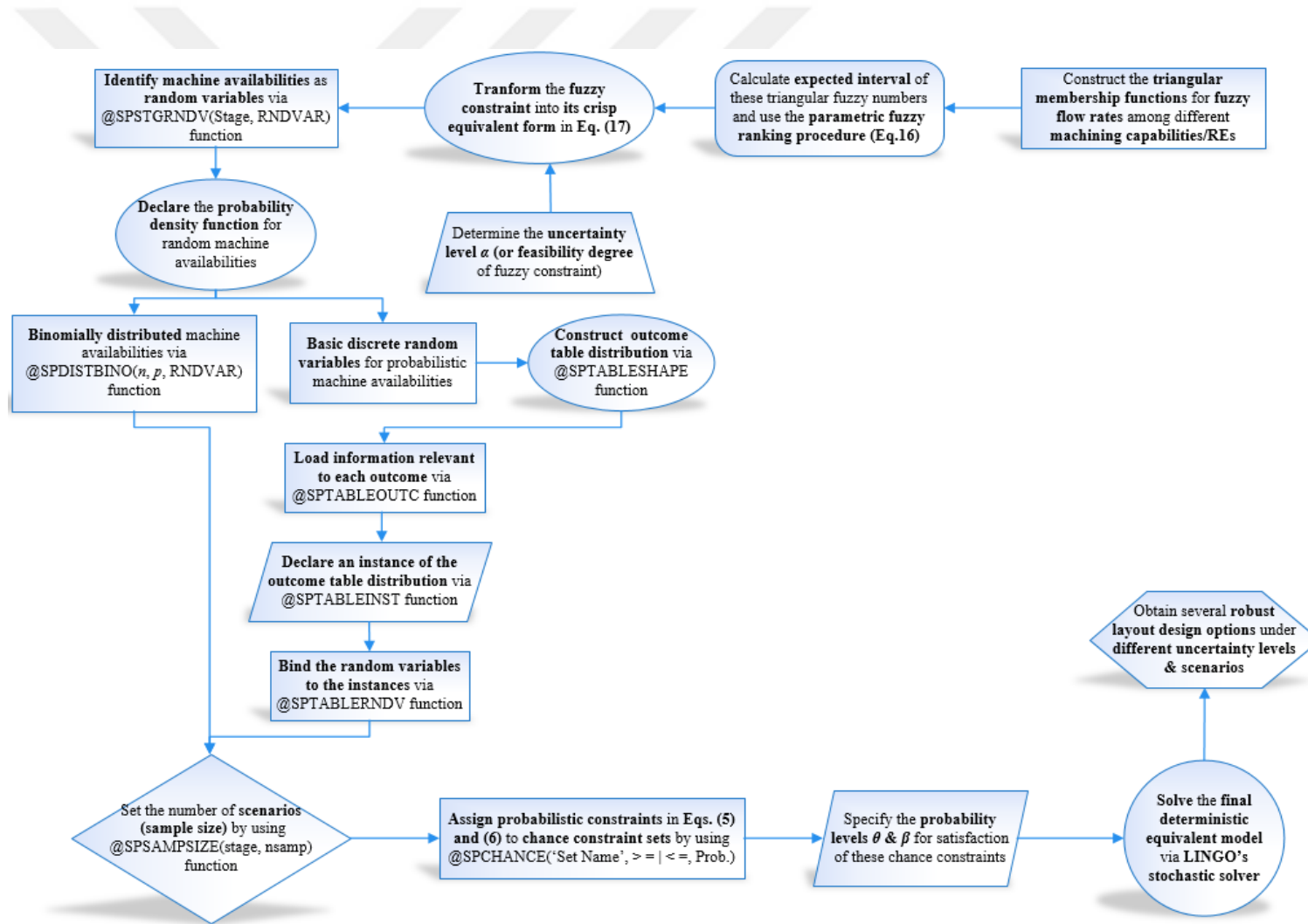


Figure 3.2 Summary of the proposed hybrid chance-constrained stochastic program with a fuzzy resolution approach for the robust FS-BCBDL problem

Machines (a)	REs (No overlap)					
	1	2	3	4	5	6
1	*					
2		*				
3			*			
4				*		
5					*	
6						*

Machines (b)	REs (Low overlap)					
	1	2	3	4	5	6
1	*	*				
2	*	*				
3			*	*		
4			*	*		
5					*	*
6					*	*

Machines (c)	REs (Medium overlap)					
	1	2	3	4	5	6
1	*	*	*			
2	*	*			*	
3	*		*	*		
4			*	*		*
5				*	*	*
6		*			*	*

Machines (d)	REs (High overlap)					
	1	2	3	4	5	6
1	*	*	*	*	*	
2	*	*		*	*	
3	*		*	*		*
4	*		*	*		*
5		*		*	*	*
6		*	*		*	*

Machines (e)	REs (Very high overlap)					
	1	2	3	4	5	6
1	*	*	*	*	*	*
2	*	*	*	*	*	*
3	*	*	*	*	*	*
4	*		*	*	*	*
5	*	*		*	*	*
6		*	*	*	*	*

Machines (f)	REs (Fully overlap)					
	1	2	3	4	5	6
1	*	*	*	*	*	*
2	*	*	*	*	*	*
3	*	*	*	*	*	*
4	*	*	*	*	*	*
5	*	*	*	*	*	*
6	*	*	*	*	*	*

(* RE is available)

Figure 3.3 a) No, b) low, c) medium, d) high, e) very high, and f) fully machine-capability overlap cases

Furthermore, fuzzy demand data with the processing routes (in terms of the REs) of 7- different parts are shown in Table 3.3. With the help of this process flow and uncertain demand data, fuzzy transition rate matrix (or fuzzy part flow rates among various REs) is computed by using Equation (3.11) and presented in Table 3.4. It should be noted here again that sum of the part flow rates in Table 3.4 is also equal to a fuzzy number, i.e., (0.918, 0.9996, 1.1233) which covers the value of “1” as an expected result. Finally, unit part flow (or material transfer) costs are computed depending on the symmetric rectilinear distances between different locations on the factory floor and given in Table 3.5.

Table 3.3 Fuzzy demand quantities and manufacturing routes of the parts in terms of REs

Parts	Demands (in units)	Process flow information in terms of REs (Processing routes)
1	(5, 10, 15)	RE1 – RE3 – RE4 – RE6 – RE1
2	(20, 30, 40)	RE2 – RE4 – RE5 – RE3
3	(5, 15, 20)	RE1 – RE2 – RE5 – RE4 – RE3 – RE6
4	(10, 20, 30)	RE1 – RE4 – RE2 – RE1 – RE5 – RE2 – RE3
5	(40, 50, 60)	RE1 – RE6 – RE3 – RE2 – RE6 – RE4 – RE1
6	(5, 10, 20)	RE3 – RE1 – RE6 – RE5 – RE1
7	(50, 70, 80)	RE3 – RE5 – RE6 – RE2

Table 3.4 Fuzzy transition rates (or uncertain part flow patterns) between different REs

From /To	RE1	RE2	RE3	RE4	RE5	RE6
RE1	(0, 0, 0)	(0.01,0.016, 0.016)	(0.01,0.011, 0.012)	(0.019,0.021, 0.024)	(0.019,0.021, 0.024)	(0.128,0.131, 0.145)
RE2	(0.019,0.021, 0.024)	(0, 0, 0)	(0.019,0.021, 0.024)	(0.032,0.032, 0.032)	(0.01,0.016, 0.016)	(0.05,0.053, 0.064)
RE3	(0.01,0.01, 0.016)	(0.05,0.053, 0.064)	(0, 0, 0)	(0.01,0.011, 0.012)	(0.07, 0.075, 0.08)	(0.01,0.016, 0.016)
RE4	(0.05,0.053, 0.064)	(0.019,0.021, 0.024)	(0.01,0.016, 0.016)	(0, 0, 0)	(0.032,0.032, 0.032)	(0.01,0.011, 0.012)
RE5	(0.01,0.011, 0.016)	(0.019,0.021, 0.024)	(0.032,0.032, 0.032)	(0.01,0.016, 0.016)	(0, 0, 0)	(0.07, 0.075, 0.08)
RE6	(0.01,0.011, 0.012)	(0.07, 0.075, 0.08)	(0.05,0.053, 0.064)	(0.05,0.053, 0.064)	(0.01,0.011, 0.016)	(0, 0, 0)

Table 3.5 Unit part flow costs among fixed locations

Locations	Locations					
	1	2	3	4	5	6
1	0	1	2	1	2	3
2	1	0	1	2	1	2
3	2	1	0	3	2	1
4	1	2	3	0	1	2
5	2	1	2	1	0	1
6	3	2	1	2	1	0

When the proposed fuzzy-stochastic programming model is run with these data through the stochastic solver of LINGO 19.0 on an Intel Corei7 2 GHz IBM PC, comparative results are provided in Tables 3.6, 3.11, 3.12 for different capability overlap cases with distinct machine availabilities and satisfaction probabilities for the chance constraint sets.

Details of the deterministic, fuzzy, stochastic and fuzzy-stochastic optimization results are first given in Table 3.6 where the sample size is set to 5, machine availabilities and the satisfaction probabilities for the chance constraint sets are equal to 90% and 80%-90%, respectively. In addition to the total number of random variables, integer variables and constraints, optimization details such as the CPU time, total solver iterations and extended solver steps are also reported in these tables. It should be emphasized here that the expected values of triangular fuzzy numbers are calculated by using Equation (3.19) and used to obtain the deterministic optimization results. It is also assumed that each machine will be in active statue (or 100% machine

availability) in the deterministic cases. Moreover, this assumption is also valid for the fuzzy optimization cases where $\alpha = 0$ and $\alpha = 1$. In the stochastic optimization cases, the expected values of triangular fuzzy numbers are utilized again as in the case of deterministic optimization (Jimenez et al. 2007, Jimenez 1996).

$$EV(\widetilde{FP}_{tr}) = \frac{FP_{tr}^p + 2.FP_{tr}^m + FP_{tr}^o}{4} \quad \forall (t, r) \in R \quad (3.19)$$

It should also be highlighted here again that the probabilistic machine availabilities are supposed to be basic discrete random variables with known probabilities in the stochastic optimization case-1. Alternatively, these machine availabilities are supposed to fit binomial distribution in the stochastic optimization case-2 to compare the results of different probability distributions. In the fuzzy-stochastic optimization cases, these different probability distributions for the random machine availabilities are also taken into account as well as the fuzzy part flow rates. In these stochastic and fuzzy-stochastic optimization cases, satisfaction probabilities for the chance constraints in Equations (3.5) – (3.6) are set to 80% and 90%, respectively (see Table 3.6). According to the comparative analysis in Table 3.6, layout objective that consists of total distances and part flow rates decreases in all of the deterministic, fuzzy, stochastic and fuzzy-stochastic cases when the machine-capability overlap is increased. For instance, total layout score of no overlap case is significantly high (i.e., 51.41448) in deterministic optimization when compared to the objective value of fully overlap case (i.e., 0). This situation is also valid for all of the other optimization cases. Thus, it is clear that the degree of machine-capability overlap has a substantial effect on the layout score of the capability-based distributed layouts. It is also obvious from Table 3.6 that the random machine availabilities has also considerable impact on the layout objective. Because, when the deterministic and stochastic optimization results are compared, it can be inferred that the layout score will increase due to the random machine breakdowns. It should also be underlined here that any feasible solution couldn't be achieved for no overlap case in all of the stochastic and fuzzy-stochastic optimization cases due to the machine unavailability risk.

Table 3.6 Comparative results for the deterministic, fuzzy, stochastic and fuzzy-stochastic optimization cases (Sample size = 5; machine availabilities = 90%; probability for breakdowns = 10%; probabilities for the chance-constraint sets = 80% and 90%, respectively)

		No overlap	Low overlap	Medium overlap	High overlap	Very high overlap	Fully overlap
Deterministic case	Overall layout score	51.41448	25.72375	20.05333	13.7145	7.010125	0
	CPU time (sec.)	8.59	1.33	15.96	7.61	16.12	0.09
	Total solver iterations	101030	15320	293150	75531	224817	0
	Extended solver steps	680	247	11746	5416	21765	0
	Total integer variables				252		
	Total constraints				481		
Stochastic case-1 (discrete random variables with known probabilities)	Total random variables				6		
	Total integer variables (Deteq.)				258		
	Total constraints (Deteq.)				771		
	Overall layout score	Infeasible (N.A)	31.99182	24.20307	18.119	12.71015	7.010125
	# of unsatisfied scenarios for CCP1	5	1	1	1	1	1
	Actual probability for CCP1	0%	80%	80%	80%	80%	80%
	# of unsatisfied scenarios for CCP2	5	0	0	0	0	0
	Actual probability for CCP2	0%	100%	100%	100%	100%	100%
	CPU time (sec.)	1.38	1.28	5.63	6.89	4.82	1.32
	Total solver iterations	0	5782	83376	93312	43281	6568
	Extended solver steps	0	14	971	1655	2809	99
	Total random variables				6		
Stochastic case-2 (Binomial distribution)	Total integer variables (Deteq.)				260		
	Total constraints (Deteq.)				771		
	Overall layout score	Infeasible (N.A)	38.25903	30.54395	23.6171	18.41923	14.02025
	# of unsatisfied scenarios for CCP1	5	1	1	1	1	1
	Actual probability for CCP1	0%	80%	80%	80%	80%	80%
	# of unsatisfied scenarios for CCP2	5	0	0	0	0	0
	Actual probability for CCP2	0%	100%	100%	100%	100%	100%
	CPU time (sec.)	1.61	2.99	11.42	23.12	13.76	4.76
	Total solver iterations	0	41380	143314	252765	151771	58376
	Extended solver steps	0	19	518	1760	1760	933
	Total random variables				6		
	Total integer variables (Deteq.)				260		
	Total constraints (Deteq.)				771		

Table 3.6 Continues

		No overlap	Low overlap	Medium overlap	High overlap	Very high overlap	Fully overlap
Fuzzy case ($\alpha = 0$)	Overall layout score	51.3365	25.6386	19.9593	13.6331	6.9588	0
	CPU time (sec.)	8.77	0.96	9.74	8.86	5.35	0.009
	Total solver iterations	10829	9954	155883	59457	32755	0
	Extended solver steps	723	132	4090	3373	2970	0
Fuzzy case ($\alpha = 1$)	Overall layout score	51.49245	25.8089	20.14735	13.7959	7.06145	0
	CPU time (sec.)	7.28	1.08	5.02	17.14	19.49	0.09
	Total solver iterations	87588	12255	99820	181584	203952	0
	Extended solver steps	503	192	3796	12315	19050	0
Fuzzy-stochastic case-1 ($\alpha = 0$)	Overall layout score	Infeasible (N.A)	31.8928	24.0983	18.0151	12.6214	6.9588
	# of unsatisfied scenarios for CCP1	5	1	1	1	1	1
	Actual probability for CCP1	0%	80%	80%	80%	80%	80%
	# of unsatisfied scenarios for CCP2	5	0	0	0	0	0
	Actual probability for CCP2	0%	100%	100%	100%	100%	100%
	CPU time (sec.)	1.43	1.32	4.0	8.2	13.53	1.29
	Total solver iterations	0	8505	53461	115247	154565	6501
	Extended solver steps	0	39	263	1376	6722	214
Fuzzy-stochastic case-2 ($\alpha = 0$)	Overall layout score	Infeasible (N.A)	38.1482	30.4241	23.4899	18.2956	13.9176
	# of unsatisfied scenarios for CCP1	5	1	1	1	1	1
	Actual probability for CCP1	0%	80%	80%	80%	80%	80%
	# of unsatisfied scenarios for CCP2	5	0	0	0	0	0
	Actual probability for CCP2	0%	100%	100%	100%	100%	100%
	CPU time (sec.)	2.44	2.79	7.91	13.76	13.98	5.97
	Total solver iterations	0	34099	108925	147976	160771	75157
	Extended solver steps	0	29	248	1061	1861	1234

Table 3.6 Continues

		No overlap	Low overlap	Medium overlap	High overlap	Very high overlap	Fully overlap
	Overall layout score	Infeasible (N.A)	32.09085	24.30785	18.2229	12.7989	7.06145
Fuzzy-stochastic case-1 ($\alpha = 1$)	# of unsatisfied scenarios for CCP1	5	1	1	1	1	1
	Actual probability for CCP1	0%	80%	80%	80%	80%	80%
	# of unsatisfied scenarios for CCP2	5	0	0	0	0	0
	Actual probability for CCP2	0%	100%	100%	100%	100%	100%
	CPU time (sec.)	1.47	1.15	8.91	6.44	6.51	0.69
	Total solver iterations	0	6665	126006	74110	81924	5247
	Extended solver steps	0	46	1350	1056	3227	149
	Overall layout score	Infeasible (N.A)	38.36985	30.6638	23.7443	18.54285	14.1229
Fuzzy-stochastic case-2 ($\alpha = 1$)	# of unsatisfied scenarios for CCP1	5	1	1	1	1	1
	Actual probability for CCP1	0%	80%	80%	80%	80%	80%
	# of unsatisfied scenarios for CCP2	5	0	0	0	0	0
	Actual probability for CCP2	0%	100%	100%	100%	100%	100%
	CPU time (sec.)	1.34	3.54	8.27	17.92	14.08	5.07
	Total solver iterations	0	42150	104781	191261	136713	43373
	Extended solver steps	0	26	464	1180	1469	476

*Deterministic equivalent form (Deteq.)

Moreover, it is clearly seen in Table 3.6 that binomially distributed random machine availability may cause higher layout scores. Furthermore, deterministic equivalent form of the binomial distribution may cause larger number of integer variables when compared to the basic discrete variables with known probabilities. By the way, both of these probability distributions have the same number of total constraints in their deterministic equivalent forms that are provided automatically by LINGO 19.0 optimization software. When the deterministic, stochastic and fuzzy optimization cases are compared to each other, it is seen in Table 3.6 that there is no substantial difference between the layout scores of the deterministic and fuzzy optimization cases. Indeed, the layout scores of deterministic optimization took values between the lowest ($\alpha = 1$) and highest ($\alpha = 0$) uncertainty levels of the fuzzy optimization cases. This means that the uncertainty in the part demands has not a crucial influence on the total layout score as much as the random machine breakdowns. Therefore, layout scores of the fuzzy optimization cases are relatively low as in the deterministic cases since the random machine breakdowns are not considered. However, the layout score will increase in these fuzzy optimization cases when the risk-free solution ($\alpha = 1$) is desired. If the most risky solution ($\alpha = 0$) is desired under the highest uncertainty level, the layout score will also reduce as an expected result. Therefore, various distributed layout alternatives can be obtained by changing the values of this parameters in this α -parametric fuzzy resolution approach. Finally, when the results of fuzzy-stochastic optimization cases which consider both of the fuzzy and stochastic layout parameters are evaluated, it can be concluded that the layout scores took fewer values in case of $\alpha = 0$ when compared to the stochastic optimization cases. Nevertheless, layout scores will take larger values than the stochastic optimization cases when risk-free layout design is needed ($\alpha = 1$). Finally, the number of unsatisfied scenarios is equal to “1” for the first chance constraint set (CCP1) in all of the stochastic and fuzzy-stochastic optimization cases. Therefore, the actual probabilities of the CCP1 is also equal to its target probability values (i.e., 80%). On the other hand, since all of the 5-scenarios are fully satisfied for the second chance constraint set, namely CCP2, its actual probability (i.e., 100%) is larger than its target probability, i.e., 90%. In these fuzzy-stochastic optimization cases-1 and 2 ($\alpha = 0$ & $\alpha = 1$), all of the satisfied and unsatisfied scenarios with the machine availabilities

are presented in Table 3.7 for the medium overlap case only. According to Table 3.7, the CCP1 is unsatisfied in scenario-2 of fuzzy-stochastic optimization case-1 where machines-1 and 2 are in inactive statue or broken down at the same time. Since this unsatisfied scenario-2 is compatible with the unavailabilities of machines-1 and 2, these machines can be called as critical machines, which necessitate preventive maintenance actions to avoid their simultaneous downtime. On the other hand, both CCP1 and CCP2 are satisfied in scenario-5 despite of the unavailability of machine-3. Thus, the generated robust layout is able to satisfy all of the probabilistic constraints in spite of the absence of this non-critical machine-3. In a similar manner, instantaneous unavailability of machines-1, 4 and 6 caused the unsatisfied scenario-1 in fuzzy-stochastic optimization case-2. For that reason, preventive maintenance actions should be taken seriously for this machine group in order to maintain continuous production of the facility. On the other hand, unavailability of non-critical machines-3 and 5 doesn't cause any constraint violation in these scenarios since the alternative machines which have similar processing capabilities are available or not broken down.

Table 3.7 Machine availabilities under different scenarios and the resulted constraint satisfactions for medium overlap case

Fuzzy-stochastic case-1 ($\alpha = 0$ & $\alpha = 1$)								
Scenario No	Machine Availabilities (φ_k)						CCP1	CCP2
	1	2	3	4	5	6		
1	A	A	A	A	A	A	Satisfied	Satisfied
2	NA	NA	A	A	A	A	Unsatisfied	Satisfied
3	A	A	A	A	A	A	Satisfied	Satisfied
4	A	A	A	A	A	A	Satisfied	Satisfied
5	A	A	NA	A	A	A	Satisfied	Satisfied
Fuzzy-stochastic case-2 ($\alpha = 0$ & $\alpha = 1$)								
1	NA	A	A	NA	A	NA	Unsatisfied	Satisfied
2	A	A	A	A	A	A	Satisfied	Satisfied
3	A	A	NA	A	A	A	Satisfied	Satisfied
4	A	A	A	A	NA	A	Satisfied	Satisfied
5	A	A	A	A	A	A	Satisfied	Satisfied

It should also be highlighted that when some of the machines are broken down (or in inactive statue), all of their processing capabilities (or REs) are disappeared from their locations. For instance, machine-location assignments for the most risky robust layout design options and its relevant RE distributions of the fuzzy-stochastic optimization case-1 are displayed in Figures 3.4 and 3.5 for the medium capability overlap case under scenario-2 ($\alpha = 0$). Similarly, the risk-free robust layout design with its RE distributions are also generated for the same fuzzy-stochastic optimization case-1 as shown in Figures 3.6 and 3.7 ($\alpha = 1$). Actually, both of these risky and risk-free robust layout designs are acquired under scenario-2 where the machines-1 and 2 are in inactive statue or broken down. According to Figures 3.4 and 3.6, the produced robust layout designs are completely different from each other under different uncertainty levels of the fuzzy part flow rates. This means that although the fuzzy part flow rates have not a significant effect on the layout score, different uncertainty levels of the fuzzy part flows may change the robust layout design completely. This is the reason why we took into account the fuzzy part flow rates by using such an α -parametric fuzzy resolution approach.

Machine locations			RE Distributions				
1 4	2 6	3 <u>1</u>	1 ■ ● ♣	2 ▲ ► ♣	3	RE1-♥	RE4-●
4 <u>2</u>	5 3	6 5	4	5 ♥ ■ ●	6 ● ► ♣	RE2-▲	RE5-►
						RE3-■	RE6-♣

Figure 3.4 Machine-location assignments & RE distributions under scenario-2 of the fuzzy-stochastic case-1 ($\alpha = 0$)

RE1 – Distribution

1	0	2	0	3	<u>0</u>
4	<u>0</u>	5	1	6	0

RE2 – Distribution

1	0	2	0	3	<u>0</u>
4	<u>0</u>	5	0	6	1

RE3 – Distribution

1	1	2	0	3	<u>0</u>
4	0	5	1	6	0

RE4 – Distribution

1	1	2	0	3	0
4	0	5	1	6	1

Distance-based objective = 7

RE5 – Distribution

1	0	2	1	3	0
4	<u>0</u>	5	0	6	1

RE6 – Distribution

1	1	2	1	3	0
4	0	5	0	6	1

Figure 3.5 RE distributions of the most risky robust layout under scenario-2 of the fuzzy-stochastic case-1 ($\alpha = 0$)

Machine locations			RE Distributions		
1 <u>2</u>	2 4	3 6	1	2 ■ ● ♣	3 ▲ ► ♣
4 3	5 5	6 <u>1</u>	4 ♥ ■ ●	5 ● ► ♣	6

Figure 3.6 Machine-location assignments & RE distributions under scenario-2 of the fuzzy-stochastic case-1 ($\alpha = 1$)

RE1 – Distribution

1	<u>0</u>	2	0	3	0
4	1	5	0	6	<u>0</u>

RE2 – Distribution

1	<u>0</u>	2	0	3	1
4	0	5	0	6	<u>0</u>

RE3 – Distribution

1	0	2	1	3	0
4	1	5	0	6	<u>0</u>

RE4 – Distribution

1	0	2	1	3	0
4	1	5	1	6	0

Distance-based objective = 9

RE5 – Distribution

1	<u>0</u>	2	0	3	1
4	0	5	1	6	0

RE6 – Distribution

1	0	2	1	3	1
4	0	5	1	6	0

Distance-based objective = 9

Figure 3.7 RE distributions of the risk-free robust layout under scenario-2 of the fuzzy-stochastic case-1 ($\alpha = 1$)

In order to compare the results of deterministic and fuzzy-stochastic optimization cases, the proposed approach is also applied without considering any type of uncertainty. Thus, machine-locations assignments and the relevant RE distributions can be depicted as in Figures 3.8 and 3.9 for the medium capability overlap case in deterministic optimization where all of the machines are available (or in active statue) in the long-run and the part flows rates are known with certainty. In this case, the distance-based objective is equal to 3 for each RE distribution as depicted in Figure 3.9 since there is only one unit distance from each unoccupied location to its nearest occupied location by the relevant REs. Therefore, total distance-based layout objective is equal to 18 (6 x 3) when all of the REs are taken into consideration.

Machine locations			RE Distributions		
1 1	2 6	3 3	1 ♥ ▲ ■	2 ▲ ► ♣	3 ♥ ■ ●
4 4	5 2	6 5	4 ■ ● ♣	5 ♥ ▲ ►	6 ● ► ♣

Figure 3.8 Machine-location assignments & RE distributions for medium overlap case in deterministic optimization

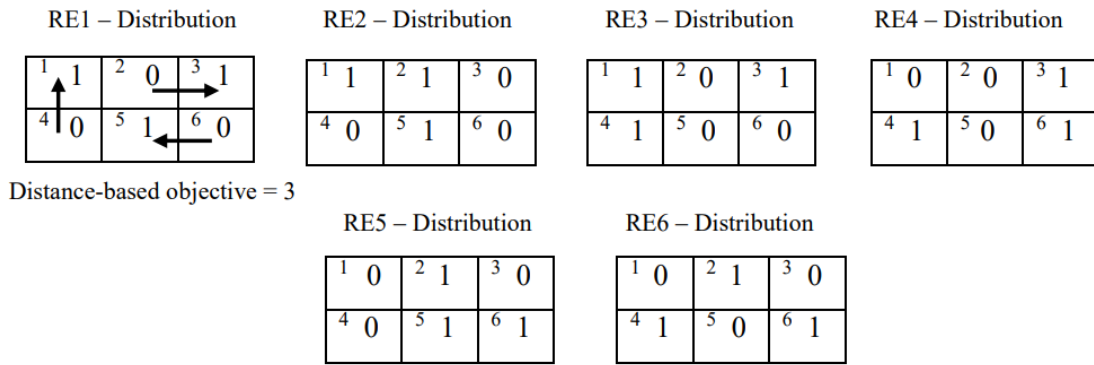


Figure 3.9 RE distributions of the generated layout for medium overlap case in deterministic optimization

Similarly, calculation of this distance-based layout score is also illustrated in Figures 3.5 and 3.7 for the fuzzy-stochastic optimization case based on RE-1. Additionally, a detailed calculation of the part flow-based layout objective is also demonstrated in Table 3.8 for the deterministic optimization case.

Table 3.8 Detailed calculation of the part flow-based layout objective in deterministic optimization

REs	Occupied locations	RE2	RE3	RE4	RE5	RE6	Score of part flow rates
RE1	Location-1	0	0	1 x 0.02125	1 x	1 x 0.13375	0.51175
	Location-3	1 x 0.0145	0	0	0.02125	1 x 0.13375	
	Location-5	0	1 x 0.011	1 x 0.02125	1 x 0.02125	1 x 0.13375	
		RE1	RE3	RE4	RE5	RE6	
RE2	Location-1	0	0	1 x 0.032275	1 x 0.0145	1 x 0.055	0.285075
	Location-2	1 x	1 x 0.02125	1 x 0.032275	0	0	
	Location-5	0.02125	1 x 0.02125	1 x 0.032275	0	1 x 0.055	
		RE1	RE2	RE4	RE5	RE6	
RE3	Location-1	0	0	1 x 0.011	1 x 0.075	1 x 0.0145	0.3865
	Location-3	0	1 x 0.055	0	1 x 0.075	1 x 0.0145	
	Location-4	1 x 0.0115	1 x 0.055	0	1 x 0.075	0	
		RE1	RE2	RE3	RE5	RE6	
RE4	Location-3	0	1 x 0.02125	0	1 x 0.0323	1 x 0.011	0.26385
	Location-4	1 x 0.055	1 x 0.02125	0	1 x 0.0323	0	
	Location-6	1 x 0.055	1 x 0.02125	1 x 0.0145	0	0	
		RE1	RE2	RE3	RE4	RE6	
RE5	Location-2	1 x 0.012	0	1 x 0.0323	1 x 0.0145	0	0.24615
	Location-5	0	0	1 x 0.0323	1 x 0.0145	1 x 0.075	
	Location-6	1 x 0.012	1 x 0.02125	1 x 0.0323	0	0	
		RE1	RE2	RE3	RE4	RE5	
RE6	Location-2	1 x 0.011	0	1 x 0.055	1 x 0.055	0	0.36
	Location-4	1 x 0.011	1 x 0.075	0	0	1 x 0.012	
	Location-6	1 x 0.011	1 x 0.075	1 x 0.055	0	0	
Total part flow-based layout objective							2.0533
Total distance-based layout objective							18.0
Total layout score							20.0533

It should be noted here that while calculating this part flow-based layout objective for the deterministic optimization case, the expected values of triangular fuzzy numbers are utilized in order to obtain crisp values for the part flow rates (see Equation 3.19). It is clearly seen in Table 3.8 that the total layout score, which is equal to sum of the total distance and part flow-based layout objectives is equal to 20.0533 which has already been found by LINGO 19.0 optimization software for the medium capability overlap case (see Table 3.6).

Furthermore, calculation of the part flow-based layout objective is presented in Table 3.9 for the previously mentioned fuzzy-stochastic optimization case-1 under scenario-2 ($\alpha = 0$) where the CCP1 is unsatisfied. As an expected result, the part flow-based layout objective is comparatively high (i.e., 33.6164) due to the simultaneous unavailability of machines-1 and 2 (see Table 3.9). On the other hand, value of the part flow-based layout score (i.e., 19.9072) will decrease considerably under scenarios-1, 3 and 4 in the same fuzzy-stochastic optimization case-1 ($\alpha = 0$) (see Table 3.10). Because, all of the machines are in active statue in these scenarios. The final layout score that is given in Table 3.6 (i.e., 24.0983) will be the expected values of these 5-scenarios with equal probabilities (i.e., 20%).

Table 3.9 Calculation of the part flow-based layout objective under scenario-2 of the fuzzy-stochastic case-1 ($\alpha = 0$)

REs	Occupied locations	RE2	RE3	RE4	RE5	RE6	Score of part flow rates
RE1	Location-5	1 x 0.013	0	0	1 x 0.02	1 x 0.1295	0.1625
		RE1	RE3	RE4	RE5	RE6	
RE2	Location-6	1 x 0.02	1 x 0.02	0	0	0	0.04
		RE1	RE2	RE4	RE5	RE6	
	Location-1	2 x 0.01	3 x 0.0515	0	1 x 0.0725	0	
RE3	Location-5	0	1 x 0.0515	0	1 x 0.0725	1 x 0.013	0.384
		RE1	RE2	RE3	RE5	RE6	
	Location-1	2 x 0.0515	3 x 0.02	0	1 x 0.0321	0	
RE4	Location-5	0	1 x 0.02	0	1 x 0.0321	1 x 0.0105	0.3222
	Location-6	1 x 0.0515	0	1 x 0.013	0	0	
		RE1	RE2	RE3	RE4	RE6	
	Location-2	1 x 0.0105	2 x 0.02	1 x 0.0321	1 x 0.013	0	0.1382
RE5	Location-6	1 x 0.0105	0	1 x 0.0321	0	0	
		RE1	RE2	RE3	RE4	RE5	
	Location-1	2 x 0.0105	3 x 0.0725	0	0	1 x 0.0105	
RE6	Location-2	1 x 0.0105	2 x 0.0725	1 x 0.0515	1 x 0.0515	0	0.5695
	Location-6	1 x 0.0105	0	1 x 0.0515	0	0	
Total part flow-based layout objective							1.6164
Total distance-based layout objective							32
Total layout score							33.6164

As it is obviously seen in Table 3.6 and 3.11-3.12 that this final layout objective took values between these minimum (i.e., 19.9072) and maximum (i.e., 33.6164) layout scores under different scenarios.

Table 3.10 Calculation of the part flow-based objective under scenarios-1, 3 and 4 of the fuzzy-stochastic case-1 ($\alpha = 0$)

REs	Occupied locations	RE2	RE3	RE4	RE5	RE6	Score of part flow rates
RE1	Location-3	0	0	1 x 0.02	1 x 0.02	1 x 0.1295	0.492
	Location-4	0	1 x 0.0105	1 x 0.02	0	1 x 0.1295	
	Location-5	1 x 0.013	0	0	1 x 0.02	1 x 0.1295	
		RE1	RE3	RE4	RE5	RE6	0.2402
RE2	Location-3	0	0	1 x 0.0321	1 x 0.013	1 x 0.0515	
	Location-4	0	1 x 0.02	1 x 0.0321	0	1 x 0.0515	
	Location-6	1 x 0.02	1 x 0.02	0	0	0	
		RE1	RE2	RE4	RE5	RE6	0.367
RE3	Location-1	1 x 0.01	1 x 0.0515	0	1 x 0.0725	0	
	Location-3	0	0	1 x 0.0105	1 x 0.0725	1 x 0.013	
	Location-5	0	1 x 0.0515	0	1 x 0.0725	1 x 0.013	
		RE1	RE2	RE3	RE5	RE6	0.2307
RE4	Location-1	1 x 0.0515	1 x 0.02	0	1 x 0.0321	0	
	Location-5	0	1 x 0.02	0	1 x 0.0321	1 x 0.0105	
	Location-6	1 x 0.0515	0	1 x 0.013	0	0	
		RE1	RE2	RE3	RE4	RE6	0.2358
RE5	Location-2	1 x 0.0105	1 x 0.02	1 x 0.0321	1 x 0.013	0	
	Location-4	0	0	1 x 0.0321	1 x 0.013	1 x 0.0725	
	Location-6	1 x 0.0105	0	1 x 0.0321	0	0	
		RE1	RE2	RE3	RE4	RE5	0.3415
RE6	Location-1	1 x 0.0105	1 x 0.0725	0	0	1 x 0.0105	
	Location-2	1 x 0.0105	1 x 0.0725	1 x 0.0515	1 x 0.0515	0	
	Location-6	1 x 0.0105	0	1 x 0.0515	0	0	
Total part flow-based layout objective							1.9072
Total distance-based layout objective							18.0
Total layout score							19.9072

Finally, in order to investigate the impacts of different machine availabilities (%) and the satisfaction probabilities for the chance constraint sets, additional computational analysis are conducted and presented in Tables 3.11 – 3.12. In Table 3.11, machine availabilities are reduced to 85% and the target probability of CCP1 is increased up to 90%. In this situation, any feasible solution couldn't found for low and medium capability overlap cases in addition to the no overlap one. However, we could achieve optimal solutions of these low and medium capability overlap cases by using the formerly defined probability values in Table 3.6. According to the Table 3.6 and 3.13, infeasible or feasible solutions with comparatively higher layout score will be provided if one decreases the machine availabilities and increases the satisfaction probabilities of the chance constraint sets.

Table 3.11 Computational analysis on the fuzzy-stochastic optimization cases (Sample size = 5; machine availabilities = 85%; probability for breakdowns = 15%; probabilities for the chance-constraint sets = 90%)

		No overlap	Low overlap	Medium overlap	High overlap	Very high overlap	Fully overlap
Fuzzy-stochastic case-1 ($\alpha = 0$)	Overall layout score	Infeasible (N.A)	Infeasible (N.A)	Infeasible (N.A)	39.5289	33.8585	27.8352
	# of unsatisfied scenarios for CCP1	5	5	5	0	0	0
	Actual probability for CCP1	0%	0%	0%	100%	100%	100%
	# of unsatisfied scenarios for CCP2	5	5	5	0	0	0
	Actual probability for CCP2	0%	0%	0%	100%	100%	100%
	CPU time (sec.)	1.17	1.52	9.0	10.23	14.36	3.23
	Total solver iterations	0	0	72876	166300	162638	50942
	Extended solver steps	0	0	2	537	1001	1400
Fuzzy-stochastic case-2 ($\alpha = 0$)	Overall layout score	Infeasible (N.A)	Infeasible (N.A)	Infeasible (N.A)	Infeasible (N.A)	Infeasible (N.A)	48.7116
	# of unsatisfied scenarios for CCP1	5	5	5	5	5	0
	Actual probability for CCP1	0%	0%	0%	0%	0%	100%
	# of unsatisfied scenarios for CCP2	5	5	5	5	5	0
	Actual probability for CCP2	0%	0%	0%	0%	0%	100%
	CPU time (sec.)	1.37	1.32	5.45	5.5	14.51	2.42
	Total solver iterations	0	0	50276	61089	174978	24083
	Extended solver steps	0	0	2	0	631	352
Fuzzy-stochastic case-1 ($\alpha = 1$)	Overall layout score	Infeasible (N.A)	Infeasible (N.A)	Infeasible (N.A)	39.91125	34.2868	28.2458
	# of unsatisfied scenarios for CCP1	5	5	5	0	0	0
	Actual probability for CCP1	0%	0%	0%	100%	100%	100%
	# of unsatisfied scenarios for CCP2	5	5	5	0	0	0
	Actual probability for CCP2	0%	0%	0%	100%	100%	100%
	CPU time (sec.)	1.86	1.44	6.21	10.34	10.14	1.9
	Total solver iterations	0	0	69228	143111	164432	25706
	Extended solver steps	0	0	2	932	734	446
Fuzzy-stochastic case-2 ($\alpha = 1$)	Overall layout score	Infeasible (N.A)	Infeasible (N.A)	Infeasible (N.A)	Infeasible (N.A)	Infeasible (N.A)	49.43015
	# of unsatisfied scenarios for CCP1	5	5	5	5	5	0
	Actual probability for CCP1	0%	0%	0%	0%	0%	100%
	# of unsatisfied scenarios for CCP2	5	5	5	5	5	0
	Actual probability for CCP2	0%	0%	0%	0%	0%	100%
	CPU time (sec.)	1.54	1.33	2.62	5.27	12.75	2.04
	Total solver iterations	0	0	7211	76568	178399	34337
	Extended solver steps	0	0	0	0	507	197

Table 3.12 Computational analysis on the fuzzy-stochastic optimization cases (Sample size = 5; machine availabilities = 85%; probability for breakdowns = 15%; probabilities for the chance-constraint sets = 60%)

		No overlap	Low overlap	Medium overlap	High overlap	Very high overlap	Fully overlap
Fuzzy-stochastic case-1 ($\alpha = 0$)	Overall layout score	51.3365	25.6386	19.9593	13.6331	6.9588	0
	# of unsatisfied scenarios for CCP1	2	2	2	2	2	2
	Actual probability for CCP1	60%	60%	60%	60%	60%	60%
	# of unsatisfied scenarios for CCP2	1	1	1	1	1	1
	Actual probability for CCP2	80%	80%	80%	80%	80%	80%
	CPU time (sec.)	8.83	1.26	9.48	6.89	7.39	0.26
	Total solver iterations	93373	12557	149735	55910	42653	0
	Extended solver steps	533	222	3723	3737	4646	0
Fuzzy-stochastic case-2 ($\alpha = 0$)	Overall layout score	Infeasible (N.A)	31.8928	24.0983	18.0151	12.6214	6.9588
	# of unsatisfied scenarios for CCP1	5	2	2	2	2	2
	Actual probability for CCP1	0%	60%	60%	60%	60%	60%
	# of unsatisfied scenarios for CCP2	5	0	0	0	0	0
	Actual probability for CCP2	0%	100%	100%	100%	100%	100%
	CPU time (sec.)	1.92	16.55	16.64	28.23	44.66	1.68
	Total solver iterations	0	171532	200260	289808	391867	10462
	Extended solver steps	0	417	1441	10255	36953	220
Fuzzy-stochastic case-1 ($\alpha = 1$)	Overall layout score	51.49245	25.8089	20.14735	13.7959	7.06145	0
	# of unsatisfied scenarios for CCP1	2	2	2	2	2	2
	Actual probability for CCP1	60%	60%	60%	60%	60%	60%
	# of unsatisfied scenarios for CCP2	1	1	1	1	1	1
	Actual probability for CCP2	80%	80%	80%	80%	80%	80%
	CPU time (sec.)	10.26	1.09	6.07	23.19	10.9	0.26
	Total solver iterations	111489	11389	114267	305812	125530	0
	Extended solver steps	867	163	4202	22346	11579	0
Fuzzy-stochastic case-2 ($\alpha = 1$)	Overall layout score	Infeasible (N.A)	32.09085	24.30785	18.2229	12.7989	7.06145
	# of unsatisfied scenarios for CCP1	5	2	2	2	2	2
	Actual probability for CCP1	0%	60%	60%	60%	60%	60%
	# of unsatisfied scenarios for CCP2	5	0	0	0	0	0
	Actual probability for CCP2	0%	100%	100%	100%	100%	100%
	CPU time (sec.)	2.91	5.38	34.23	25.49	52.96	1.53
	Total solver iterations	0	62048	364527	282836	466142	9615
	Extended solver steps	0	140	3121	9457	42259	240

It should also be noted here that the number of infeasible solutions will increase and the layout scores will also deteriorate in case of binomially distributed random machine availabilities. Fortunately, a fewer amount of infeasible solutions and comparatively better layout scores can be obtained when the random machine availabilities are described as basic discrete variables with known probability values. Similarly, computational analyses are also carried out by reducing the target probabilities of the chance constraint sets (i.e., CCP1 and CCP2) to 60% under the same machine availabilities (or probability for machinery breakdowns, i.e., 85%). In this circumstance, feasible and comparatively better layout scores are provided as in Table 3.12 for all of the capability overlap cases when compared to the former results in Table 3.6 and 3.11. However, it should be emphasized here that the higher satisfaction probabilities for these chance constraint sets are generally desired by the facility planners in real-life applications. Lastly, similar to the previously given optimization results, feasible solutions couldn't be found for no overlap case in fuzzy-stochastic optimization case-2. On the other hand, feasible solutions with poor quality layout scores could be found for no overlap case in the fuzzy-stochastic optimization case-1. Moreover, it is obviously seen in Table 3.12 that the usage of discrete random variables with known probabilities for the uncertain machine availabilities may provide better quality layout scores and feasible solutions when compared to the binomial distribution.

3.2.2 Application in a Real-Life Manufacturing System

To display the usefulness, validity and applicability of the proposed FS- BCBDL approach, a real-life implementation in a cellular manufacturing system which was previously addressed by Baykasoğlu and Gindy (2000) is also presented. The manufacturing facility receives various customer orders with comparatively low demand amounts and commonly needs layout design changes. Since the highly uncertain nature of the customer demand, they cannot estimate demand quantities accurately. For that reason, facility managers determined to describe fuzzy demand quantities for these customer orders. In the present case, the facility floor is arranged as a usual cellular manufacturing environment without considering machines' processing capabilities. The 4 x 3 block plan and the existing cellular layout of the

facility has already been displayed by Baykasoğlu and Gindy (2000) as depicted in Figure 3.10. In detail, there are 3 manufacturing cells and 7 different machines inside the factory. However, some machines have multiple copies. Therefore, it can be concluded that there are totally 12 machines and 12 available locations over the factory floor where copied machines include exactly the same REs.

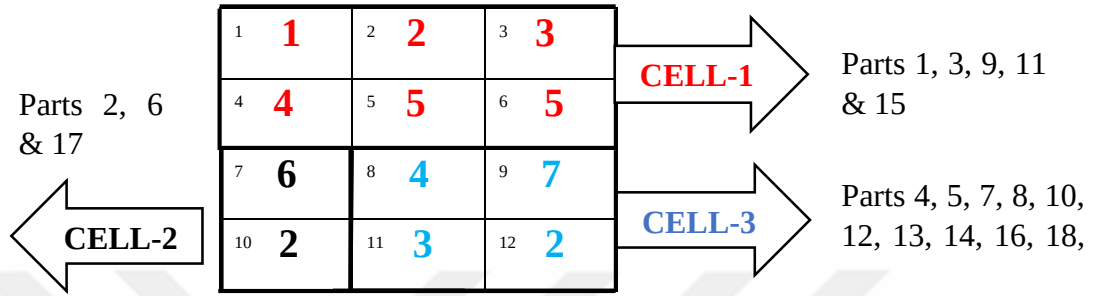


Figure 3.10 The existing cellular layout of the manufacturing facility with part assignments

The manufactured parts in each cell is also shown by Figure 3.10. Due to the extensive material transfers between different manufacturing cells and low cell capacity utilizations, the facility managers are dissatisfied with the current performance of the existing cellular layout. Indeed, non-value added material transfers between different cells are required for some parts. This may also cause extra machine requirements. For instance, one more copy of machine-2 is needed for cell-2 and two more copies of machine-2 are also needed for cell-3. Actually, this causes significant time and money consumptions, which also decreases the efficiency and profitability of the manufacturing facility.

Furthermore, capacity utilizations are calculated by including these extra copies of the required machines as 61%, 57% and 72% for each cell, respectively. Thus, facility planners seek to a better robust layout option, which improves the efficiency of the material handling operations and increase the machine capacity utilizations by taking account the machines' processing capabilities (or REs), machines' unavailability risks and uncertainty in the part flow rates. To do this, the proposed FS-BCBDL approach is considered as an alternative to achieve optimal product flows between different departments under uncertainty. Many alternative robust layouts are derived under different uncertainty levels and scenarios. Their outcomes are also compared to the present cellular layout. Before applying the proposed FS-BCBDL approach, 11

different REs are specified by the facility process planners for the machines' processing capabilities. Thereafter, uncertain demand quantities for the customer orders are also represented by triangular fuzzy numbers as presented in Table 3.13. In this table, in addition to these monthly fuzzy demand quantities, the processing requirements (or process flow information) of totally 20 manufactured parts are also identified in terms of the REs. Furthermore, the number of copies for available machines with their processing capabilities are shown in Table 3.14.

Table 3.13 Fuzzy demand quantities and process flow information of different parts in terms of REs

Parts	(Demand x 1000)	Process flow information in terms of REs (Processing routes)
1	(1, 3, 5)	RE1 – RE2 – RE4
2	(0.3, 1, 2)	RE1 – RE2 – RE3
3	(0.5, 2.5, 4)	RE5 – RE6 – RE7
4	(0.52, 1.52, 2.52)	RE8 – RE5
5	(0.48, 1.48, 2.18)	RE7 – RE4 – RE5
6	(1.2, 3.5, 5.5)	RE8 – RE6 – RE7
7	(0.5, 1, 1.5)	RE8 – RE9 – RE10
8	(0.5, 2, 3)	RE9 – RE10 – RE11
9	(1, 3, 5)	RE5 – RE1 – RE2
10	(0.8, 2, 3)	RE3 – RE4
11	(1.8, 4.5, 7)	RE5 – RE6 – RE9
12	(0.3, 1, 2)	RE10 – RE9 – RE8
13	(1, 3, 5)	RE5 – RE8 – RE10
14	(1, 2.5, 4)	RE8 – RE7 – RE5
15	(1, 2.5, 3.5)	RE1 – RE2
16	(0.5, 1.9, 2.7)	RE3 – RE4
17	(1, 2.4, 3.4)	RE6 – RE7 – RE8
18	(0.6, 1.2, 2.5)	RE8 – RE9 – RE10 – RE11
19	(0.3, 1.3, 2.3)	RE5 – RE2
20	(1, 3, 5)	RE7 – RE8 – RE9

The probabilities for the random machine availabilities which are calculated based on the previous machine breakdown statistics of the facility reports are also given in Table 3.14. By using the fuzzy demand quantities and process flow data in Table 3.13, fuzzy transition rate matrix between different REs can be calculated through Equation (3.11) as in Table 3.15.

Table 3.14 Processing capabilities in terms of the REs and the number of available copies of the machine tools (Baykasoğlu and Gindy 2000)

Machines	Copies	Resource elements (REs)											Random machine availability (%)
		1	2	3	4	5	6	7	8	9	10	11	
1-Drill Press-1 (M1)	1	*											95
2-MHP Machining Centre-1 (M2-M3-M4)	3	*	*	*				*	*	*	*	*	90
3-Colchester Lathe-1 (M5-M6)	2	*	*		*			*					85
4-MHP MT50 NC Lathe-2 (M7-M8)	2	*	*		*			*			*		85
5-CNC Grinding Machine-1 (M9-M10)	2					*	*						90
6-Jones & Shipman Cyc. Grinder (M11)	1						*						95
7-Jones * Ship. Surf. Grinder (M12)	1					*							95

It should be noted here that the sum of all these transition rates is also equal to a fuzzy number, i.e., (0.956, 1.001, 1.056) which already contains “1”. Finally, sample size or the number of scenarios is set to 5 and the target probabilities for both of the chance constraint sets (CCP1 and CCP2) are specified as 80% by the facility design managers. The facility design managers preferred to utilize discrete random variables with known probabilities instead of the binomial distribution. Because, binomial distribution may result in higher total cost values or layout scores as previously discussed in Section 3.2.1.

By using the above data for this real-life case study, the proposed fuzzy-stochastic programming model is run under different uncertainty levels (α -cut levels) and scenarios with the help of LINGO 19.0 optimization suite on an Intel Corei7 2 GHz IBM PC. The details of the fuzzy-stochastic optimization results within 3-hours runtime limit are provided as in Table 3.16. According to these results, the first probabilistic constraint set is satisfied with 80% actual probability whereas the second chance constraint set is satisfied with 100% actual probability under all uncertainty levels. This means that only one scenario may cause unsatisfactory solutions because of the random machine availabilities. In other words, the first chance constraint set is not satisfied for only one scenario. Details of these satisfied and unsatisfied scenarios with the machine statue (available or not) are also presented in Table 3.17.

Table 3. 15 Fuzzy transition rates between various REs for the real-life case study

[illegible]

Table 3. 16 Fuzzy-stochastic optimization results under different uncertainty levels

α -feasibility degree of CCP2	$\alpha = 0$	$\alpha = 0.3$	$\alpha = 0.5$	$\alpha = 0.7$	$\alpha = 1$
Total random variables			12		
Total integer variables (Deteq.)			1738		
Total constraints (Deteq.)			4383		
Total nonzeros (Deteq.)			39850		
Total solver iterations	89593646	103498008	146909359	89344547	97063615
Extended solver steps	720796	694363	968737	416112	413412
CPU time (sec.)	10800	10800	10800	10800	10800
Layout score (Obj. value)	144.866	145.177	145.3622	145.5363	145.7975
Optimality gap	12%	18%	21%	19%	16%
# of unsatisfied scenarios (CCP1)	1	1	1	1	1
Actual probability (CCP1)	80%	80%	80%	80%	80%
Unsatisfied scenarios (CCP2)	0	0	0	0	0
Actual probability (CCP2)	100%	100%	100%	100%	100%

It should be noted here that the same machine availabilities and the resulted constraint satisfactions are provided under different scenarios for both most risky ($\alpha = 0$) and risk-free ($\alpha = 1$) cases. It is also obviously seen in Table 3.16 that the layout score increases when the feasibility degree of the fuzzy-stochastic constraint set (CCP2) gets higher. Actually, this is an expected result since the risk-free layout design options may generally result in relatively high cost solutions. It should be highlighted here again that the optimal layout scores are not acquired within 3-hours runtime limit. However, the generated layout alternatives under all uncertainty levels can be stated as applicable since their optimality gaps are low enough (or within acceptable levels). When the machine availabilities under different scenarios and the resulted constraint satisfactions are examined in detail (see Table 3.17), it is noticed that MHP Machining Center-1 has critical machines which necessitate effective preventive maintenance actions. Because, simultaneous breakdowns of the machines-2 and 3 (M2 and M3 in MHP Machining Center-1) may cause unsatisfied CCP1 (scenario#2), which violates the first probabilistic constraint set (see Tables 3.18 and 3.19). Although this machining center has three copies of the same machines, the simultaneous breakdown of two of them may result in violation of the first chance constraint set. Therefore, preventive maintenance actions play an important role for these critical machines.

Table 3.17 Machine availabilities under different scenarios and the resulted constraint satisfactions
($\alpha = 0$ & $\alpha = 1$)

Scenario	Machine Availabilities (φ_k)												CCP1	CCP2
No	1	2	3	4	5	6	7	8	9	10	11	12		
1	A	A	A	A	A	A	NA	NA	A	A	A	A	Satisfied	Satisfied
2	A	NA	NA	A	A	A	A	A	A	A	A	A	Unsatisfied	Satisfied
3	A	A	A	A	A	NA	A	A	A	A	A	A	Satisfied	Satisfied
4	A	A	A	A	A	A	A	A	A	A	A	A	Satisfied	Satisfied
5	A	A	A	NA	A	A	A	A	A	A	A	A	Satisfied	Satisfied

It should also be noted here that if all of these three machines (M2, M3 and M4) in this MHP Machining Center-1 become inactive statue (or broken down), the proposed fuzzy-stochastic programming model cannot provide any feasible solution. Because, some of the REs, i.e., RE3, RE8, RE9 and RE11 are covered by this MHP Machining Center-1 only (see Table 3.14). Thus, simultaneous breakdown of three of them will result in an infeasible solution since it is not possible to reach these REs from any factory location.

According to Table 3.17, the simultaneous breakdown of non-critical machines M7 and M8 (MHP MT50 NC Lathe-2) doesn't lead to any unsatisfied scenario (see scenario#1). Because, the REs, i.e., RE1, RE2, RE4, RE7 and RE10 which are covered by these machines have already been included by other alternative machines like MHP MT50 NC Lathe-2 and MHP Machining Center-1. Finally, the generated most risky robust layout design ($\alpha = 0$) including the machine-location assignments and the relevant binary RE distributions are demonstrated in Figures 3.11 and 3.12 under scenario#4 where all of the machines are available (or not broken down). It is obviously seen in Table 3.17 that all of the probabilistic constraint sets are satisfied under this scenario.

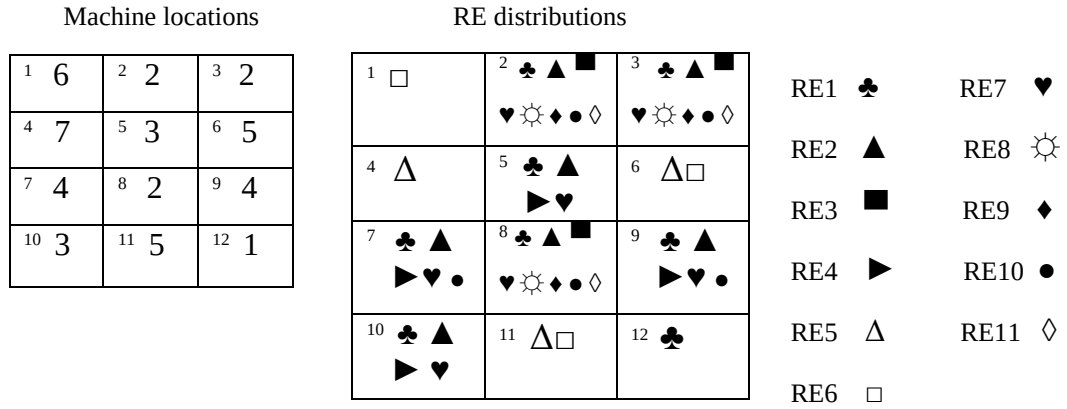


Figure 3.11 Machine-location assignments and the RE distributions for the most risky case under scenario#4 ($\alpha = 0$)

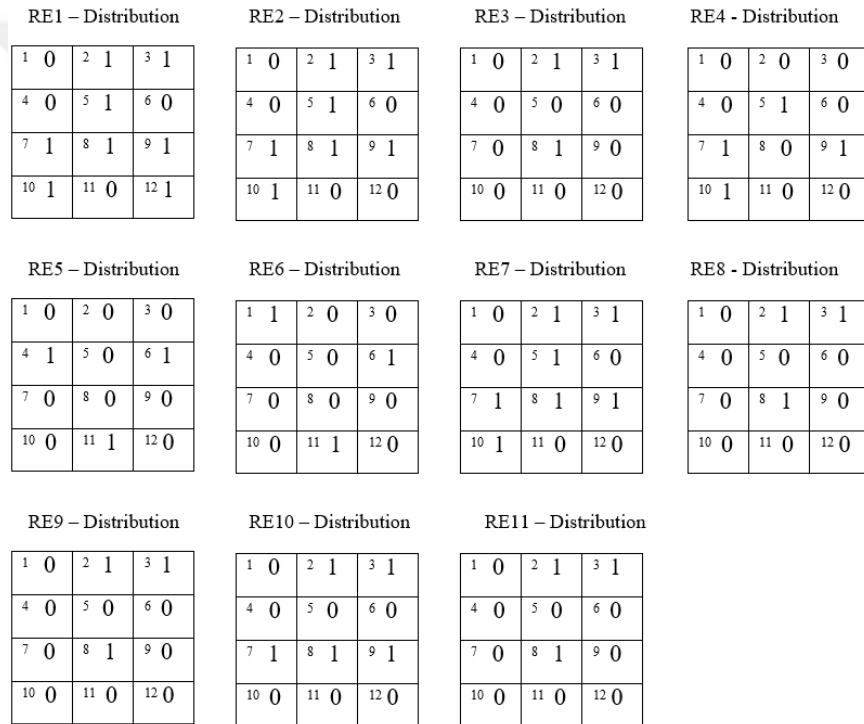


Figure 3.12 RE distribution tables of the robust layout design for the most risky case under scenario#4 ($\alpha = 0$)

As displayed in Figures 3.11-3.12, since all of the machines are in active statue under scenario#4, all of their processing capabilities are appeared in the relevant RE distribution tables (or binary matrix). In these RE distribution tables, “1” shows that the regarding factory location contains this RE. Otherwise, the factory locations that don’t cover the regarding RE are marked with “0”. In fact, these REs distributions can be provided by the machine-location assignment decisions. If a machine is allocated to any location, it’s all operation capabilities will be covered by that location.

As it is obviously seen in Figure 3.11 that some factory locations have the same machines since some of the machines have multiple copies. In a similar manner, the produced robust biased capability-based layout design with its RE distributions are also depicted in Figures 3.13 - 3.14 under scenario#2 where M2 and M3 in MHP Machining Center-1 are not in active statue (or not available).

Machine locations			RE Distributions		
1 6	2 2	3 2	1	2	3
4 7	5 3	6 5	4	5	6
7 4	8 2	9 4	7	8	9
10 3	11 5	12 1	10	11	12

Figure 3.13 Machine-location assignments & RE distributions for the most risky case under scenario#2 ($\alpha = 0$)

RE1 – Distribution	RE2 – Distribution	RE3 – Distribution	RE4 - Distribution
1 0	1 0	1 0	1 0
2 0	2 0	2 0	2 0
3 1	3 1	3 1	3 0
4 0	4 0	4 0	4 0
5 1	5 1	5 0	5 1
6 0	6 0	6 0	6 0
7 1	7 1	7 0	7 1
8 0	8 0	8 0	8 0
9 1	9 1	9 0	9 1
10 1	10 1	10 0	10 1
11 0	11 0	11 0	11 0
12 1	12 0	12 0	12 0

RE5 – Distribution	RE6 – Distribution	RE7 – Distribution	RE8 - Distribution
1 0	1 1	1 0	1 0
2 0	2 0	2 0	2 0
3 0	3 0	3 1	3 1
4 1	4 0	4 0	4 0
5 0	5 0	5 1	5 0
6 1	6 1	6 0	6 0
7 0	7 0	7 1	7 0
8 0	8 0	8 0	8 0
9 0	9 0	9 1	9 0
10 0	10 0	10 1	10 0
11 1	11 1	11 0	11 0
12 0	12 0	12 0	12 0

RE9 – Distribution	RE10 – Distribution	RE11 – Distribution
1 0	1 0	1 0
2 0	2 0	2 0
3 1	3 1	3 1
4 0	4 0	4 0
5 0	5 0	5 0
6 0	6 0	6 0
7 0	7 1	7 0
8 0	8 0	8 0
9 0	9 1	9 0
10 0	10 0	10 0
11 0	11 0	11 0
12 0	12 0	12 0

Figure 3.14 RE distribution tables of the robust layout design for the most risky case under scenario#2 ($\alpha = 0$)

Since the machines M2 and M3 in MHP Machining Center-1 are broken down or not available under scenario#2, their locations don't contain their processing capabilities as shown in Figure 3.13. Their processing capabilities or REs are also marked with "0" for these locations in the RE distribution tables (see Figure 3.14). When objective function value of the robust layout design under this scenario#2 is compared to the previous case in scenario#4, it is obvious that its layout score will get worse because of the unavailable machines lead to this unsatisfied scenario. In other words, layout score (or objective function value) increase when some of the critical machines are not in active statue. Therefore, total expected score of this robust layout design option will also increase when compared to scenario#4. Finally, the machine-location assignments and the RE distribution tables of the generated risk-free robust layout design alternative ($\alpha = 1$) are also depicted in Figures 3.15 - 3.18 under both scenarios#4 and #2, respectively.

Machine locations			RE distributions		
1 1	2 5	3 3	1 	2  	3    
4 4	5 2	6 4	4     	5        	6     
7 6	8 3	9 5	7 	8    	9  
10 7	11 2	12 2	10 	11        	12        

Figure 3.15 Machine-location assignments and the RE distributions for the risk-free case under scenario#4 ($\alpha = 1$)

RE1 – Distribution	RE2 – Distribution	RE3 – Distribution	RE4 - Distribution
1 1 2 0 3 1	1 0 2 0 3 1	1 0 2 0 3 0	1 0 2 0 3 1
4 1 5 1 6 1	4 1 5 1 6 1	4 0 5 1 6 0	4 1 5 0 6 1
7 0 8 1 9 0	7 0 8 1 9 0	7 0 8 0 9 0	7 0 8 1 9 0
10 0 11 1 12 1	10 0 11 1 12 1	10 0 11 1 12 1	10 0 11 0 12 0
RE5 – Distribution	RE6 – Distribution	RE7 – Distribution	RE8 - Distribution
1 0 2 1 3 0	1 0 2 1 3 0	1 0 2 0 3 1	1 0 2 0 3 0
4 0 5 0 6 0	4 0 5 0 6 0	4 1 5 1 6 1	4 0 5 1 6 0
7 0 8 0 9 1	7 1 8 0 9 1	7 0 8 1 9 0	7 0 8 0 9 0
10 1 11 0 12 0	10 0 11 0 12 0	10 0 11 1 12 1	10 0 11 1 12 1
RE9 – Distribution	RE10 – Distribution	RE11 – Distribution	
1 0 2 0 3 0	1 0 2 0 3 0	1 0 2 0 3 0	
4 0 5 1 6 0	4 1 5 1 6 1	4 0 5 1 6 0	
7 0 8 0 9 0	7 0 8 0 9 0	7 0 8 0 9 0	
10 0 11 1 12 1	10 0 11 1 12 1	10 0 11 1 12 1	

Figure 3.16 RE distribution tables of the robust layout design for the risk-free case under scenario#4 ($\alpha = 1$)

Machine locations	RE distributions
1 1 2 5 3 3	1 
4 4 5 2 6 4	2  
7 6 8 3 9 5	3  
10 7 11 2 12 2	4  
	5
	6  
	7 
	8  
	9  
	10 
	11
	12  

Figure 3.17 Machine-location assignments and the RE distributions for the risk-free case under scenario#2 ($\alpha = 1$)

RE1 – Distribution	RE2 – Distribution	RE3 – Distribution	RE4 - Distribution
1 1 2 0 3 1	1 0 2 0 3 1	1 0 2 0 3 0	1 0 2 0 3 1
4 1 5 <u>0</u> 6 1	4 1 5 <u>0</u> 6 1	4 0 5 <u>0</u> 6 0	4 1 5 <u>0</u> 6 1
7 0 8 1 9 0	7 0 8 1 9 0	7 0 8 0 9 0	7 0 8 1 9 0
10 0 11 <u>0</u> 12 1	10 0 11 <u>0</u> 12 1	10 0 11 <u>0</u> 12 1	10 0 11 <u>0</u> 12 0
RE5 – Distribution	RE6 – Distribution	RE7 – Distribution	RE8 - Distribution
1 0 2 1 3 0	1 0 2 1 3 0	1 0 2 0 3 1	1 0 2 0 3 0
4 0 5 <u>0</u> 6 0	4 0 5 <u>0</u> 6 0	4 1 5 <u>0</u> 6 1	4 0 5 <u>0</u> 6 0
7 0 8 0 9 1	7 1 8 0 9 1	7 0 8 1 9 0	7 0 8 0 9 0
10 1 11 <u>0</u> 12 0	10 0 11 <u>0</u> 12 0	10 0 11 <u>0</u> 12 1	10 0 11 <u>0</u> 12 1
RE9 – Distribution	RE10 – Distribution	RE11 – Distribution	
1 0 2 0 3 0	1 0 2 0 3 0	1 0 2 0 3 0	
4 0 5 <u>0</u> 6 0	4 1 5 <u>0</u> 6 1	4 0 5 <u>0</u> 6 0	
7 0 8 0 9 0	7 0 8 0 9 0	7 0 8 0 9 0	
10 0 11 <u>0</u> 12 1	10 0 11 <u>0</u> 12 1	10 0 11 <u>0</u> 12 1	

Figure 3.18 RE distribution tables of the robust layout design for the risk-free case under scenario#2 ($\alpha = 1$)

In contrast to the most risky case ($\alpha = 0$), feasibility degree of the fuzzy-stochastic constraint set in Equation (3.6) will be 100% and therefore, this robust layout design option in Figures 3.15 - 3.17 can be stated as fully acceptable, which is generally desired by risk-averse facility designers. However, there is still one unsatisfied scenario for this robust layout design option because of the random machine breakdowns under scenario#2. Fortunately, if the probability values for the machine availabilities that were formerly given in Table 3.16 are increased by performing additional preventive maintenance activities, the number of unsatisfied scenarios will decrease. So, risk-averse facility designers should focus more on accurate demand forecasting and machine failure prevention systems in order to obtain risk-free robust layout design options under various scenarios. Lastly, all of the solutions generated by the proposed FS-BCBDL method are compared to the existing cellular layout of the facility in Table 3.18 and Figure 3.19.

To make an appropriate comparison, machine-location assignments in the previously given existing cellular layout design (see Figure 3.10) are utilized while running the proposed optimization model. When the results of existing cellular layout are examined, it is recognized that the same machines, i.e., M2 and M3 in MHP Machining Center-1 are also critical for the existing cellular manufacturing system. Because, unavailability of these machines might cause violation of the first chance constraint set under the same scenario#2.

Table 3.18 Comparison of the present cellular layout with the proposed FS-BCBDL design method

	Total layout score (Objective value)	
	The existing cellular layout	The proposed robust BCBDL design approach
The most risky case ($\alpha = 0$)	180.554	144.866
The risk-free case ($\alpha = 1$)	181.508	145.7975

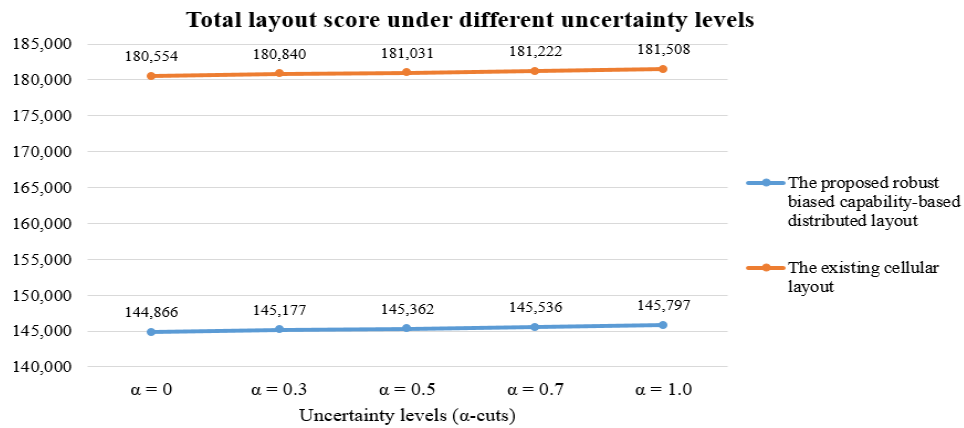


Figure 3.19 Comparison of the layout scores under different uncertainty levels

According to the comparative results in Table 3.18 and Figure 3.19, considerable improvements on the expected layout scores can be achieved through the proposed FS-BCBDL approach. Indeed, the proposed approach can provide considerable decreases in the total part flows for each uncertainty level. Moreover, each machining capability can be achieved from each site of the factory with minimum total distance by considering random machine breakdowns as well. Therefore, more operative and efficient material transfers can be performed by means of the proposed robust BCBDL method in case of uncertain demand data and random machine availabilities.

CHAPTER 4

UNEQUAL-AREA CAPABILITY-BASED FACILITY LAYOUT DESIGN

PROBLEM WITH A HEURISTIC DECOMPOSITION-BASED

ITERATIVE MATHEMATICAL PROGRAMMING APPROACH

4.1 Problem Formulation and Mathematical Formulation of the UA-CBFLD

In most of the real-life applications, manufacturing departments with machinery may have multiple processing capabilities. Besides, some of these processing capabilities are available on more than one machine that are located in different departments. In this circumstance, processing capability overlaps will arise. In a similar way, departments may also include more than one processing capability according to their machinery pool. In this situation, any manufacturing operation in the products' routes, which needs this capability or RE can be realized in alternate departments. For all of these reasons, facility layouts have to be planned by taking into account these processing capability overlaps among different departments. On the other hand, if each department has a machinery pool with a single processing capability to accomplish the production operations, capability-based layout design methods are not required. In that case, classical layout design approaches may be adequate. Because, each department acts like a different processing capability in such a situation.

As in the studies of Baykasoğlu and Subulan (2020) and Baykasoğlu et al. (2022), these processing capabilities are described by the Resource Elements (REs) in this thesis. In fact, each RE displays a group of elementary machining operations and is such that the exclusive and the shared capability boundaries among all the existing machines included in a production facility can be individually described. For more details on the REs, please consult to the studies of Gindy et al. (1996 and 1997) and Baykasoğlu et al. (2001). As it is also obviously seen in Figure 4.1 that if the machines located in different departments have the same REs, these departments will have capability overlaps. For instance, departments-1 and 2 have processing capability overlap because of the RE-1. Similarly, departments-2 and 3 have capability overlap because of the RE-2.

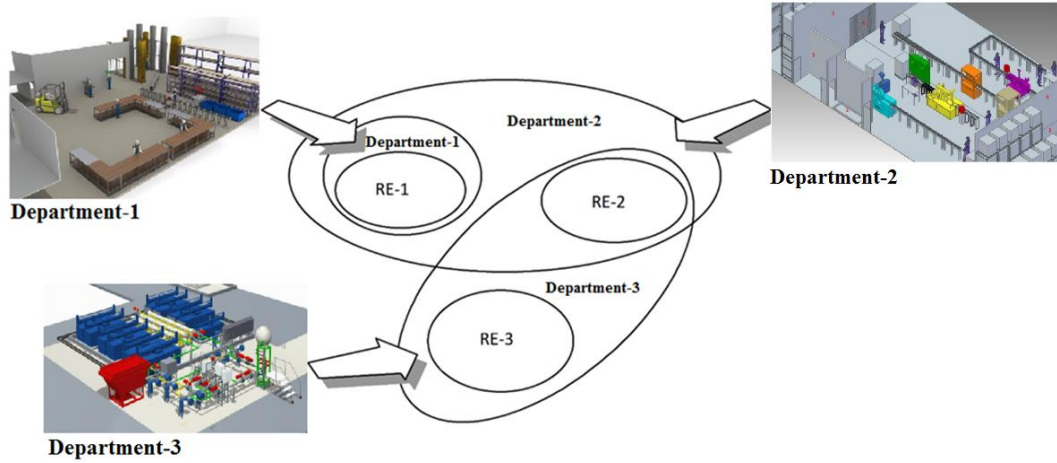


Figure 4.1 Depiction of the processing capability overlaps among different departments

However, RE-3 is available on department-3 only. Thus, there is no capability overlap on this RE. The manufacturing departments may also cover more than one RE. For example, department-3 includes some machines with two abilities such as RE-2 and RE-3. However, department-1 has a machinery with a unique processing capability which includes RE-1 only.

Thus far, there is no available study in the literature that focuses on the capability-based facility layout design with unequal department areas. As mentioned before, all of the previous researches on the capability-based layout design techniques (Baykasoğlu, 2003; Baykasoğlu and Göçken, 2010; Baykasoğlu and Subulan, 2020; Baykasoğlu et al., 2022) have considered equal-area departments or machines, which are assigned to the pre-specified facility locations. For that reason, a new problem namely the UA-CBFLD is introduced in this chapter which can be considered as an extension of the recently proposed capability-based machine layout (CB-ML) problem with unequal area departments by Baykasoğlu et al. (2022). It should also be noted here that the processing capabilities of machines located in the departments were not taken into consideration by the previous researches on the classical UA-FLD problems (Tate and Smith, 1995; Kochhar et al., 1998; Sherali et al., 2003; Kulturel-Konak, 2012; Kulturel-Konak and Konak, 2013; Xie et al., 2016; Paes et al., 2017). When the processing capabilities of machines located in the departments are incorporated into the classical UA-FLD problem, RE-based part flows have to be defined as a parameter instead of the machine or department-based material flows, which is the case in the

traditional UA-FLD problems. In other words, products' process plans are described by using the REs instead of the classical machine or department-based processing routes. Since the manufacturing routes of the products are described by the REs, product flows among different REs should be considered in the proposed UA-CBFLD problem. Therefore, the product flows between different departments, i.e., f_{ij} should be defined as a model variable for the proposed UA-CBFLD problem, instead of a parameter (or input data) which is the case in traditional UA-FLD problem. Because, product flows between different departments can only be specified after locating the departments over the facility floor. In other words, dispersal of the machining capabilities over the facility floor cannot be known before organizing the layout or determining the department locations. Indeed, the amounts of product flows between several REs can be calculated by using the RE-based process plan and demand quantities of the products through Equation (4.1). Similarly, total processing requirement of every RE can also be computed as other input through the processing times of the REs, manufacturing route and amount of demands for the products as formulated in Equation (4.2). In reality, RE's processing requirements are utilized in order to check the capacity usage or load of the machinery located in the departments.

$$\lambda_{qr} = \sum_{s=1}^S \tau_{sqr} \cdot D_s \quad \forall (q, r) \in R \ q \neq r \quad (4.1)$$

$$\delta_r = \sum_{s=1}^S \phi_{sr} \cdot D_s \cdot p_{sr} \quad \forall r \in R \quad (4.2)$$

Besides the machining capabilities in the departments, REs' total processing requirements, processing times and amounts of demands as well as the product flows between various REs constitute the key input data of the proposed UA-CBFLD problem. Moreover, maximal machine capacity usages of the departments is also identified as another model parameter. Conversely, the proposed UA-CBFLD problem desires to locate rectangular departments with unequal areas on a rectangular facility floor with dimensions (i.e., B^x, B^y) without any overlap. Each department, i.e., $(i, j) \in N$ is identified by its half of the side lengths along the horizontal and vertical axis (l_i^x

and l_i^y) and its area is equal to $A_i = 4 \cdot l_i^x \cdot l_i^y$. Furthermore, the aspect ratio, i.e., AR_i , between the side lengths of each department is also defined due to the concrete reasons that executes the maximum allowable ratio between its longest and shortest side lengths along the x -and y -axes. A capability-based facility layout is described by the coordinates of the departments' centroids (i.e., c_i^x and c_i^y), the horizontal (i.e., l_i^x) and vertical (i.e., l_i^y) dimensions as well as the allocations of the usage of REs to these departments. Additionally, it is also targeted to decide the product flows between these departments according to the facility layout design (or locating the departments on the facility floor) and to compute their capacity utilizations by using the RE allocations to each department. Based on all these additional problem characteristics, the proposed UA-CBFLD problem is much more complicated than the classical UA-FLD problem as well as the other capability-based facility layout design methods in the available literature (Baykasoğlu, 2003; Baykasoğlu and Göçken, 2010; Baykasoğlu and Subulan, 2020; Baykasoğlu et al., 2022). Lastly, primary components of the proposed UA-CBFLD problem are shown in Figure 4.2.

4.1.1 The Proposed MINLP Model Formulation for the UA-CBFLD Problem

In the light of these facts, we proposed a mixed integer non-linear programming (MINLP) model of the UA-CBFLD problem for the first time in the literature, which is formulated through Equations (4.3) – (4.29) by using the mathematical nomenclature given in Table 4.1. In fact, mathematical formulations in Equations (4.3) – (4.15) are belong to the classical UA-FLD problem in the existing literature (Sherali et al., 2003; Kulturel-Konak and Konak, 2013; Xie et al., 2016) whereas the constraint sets in Equations (4.16) – (4.26) are mainly related to the capability-based product flow part of the proposed UA-CBFLD problem.

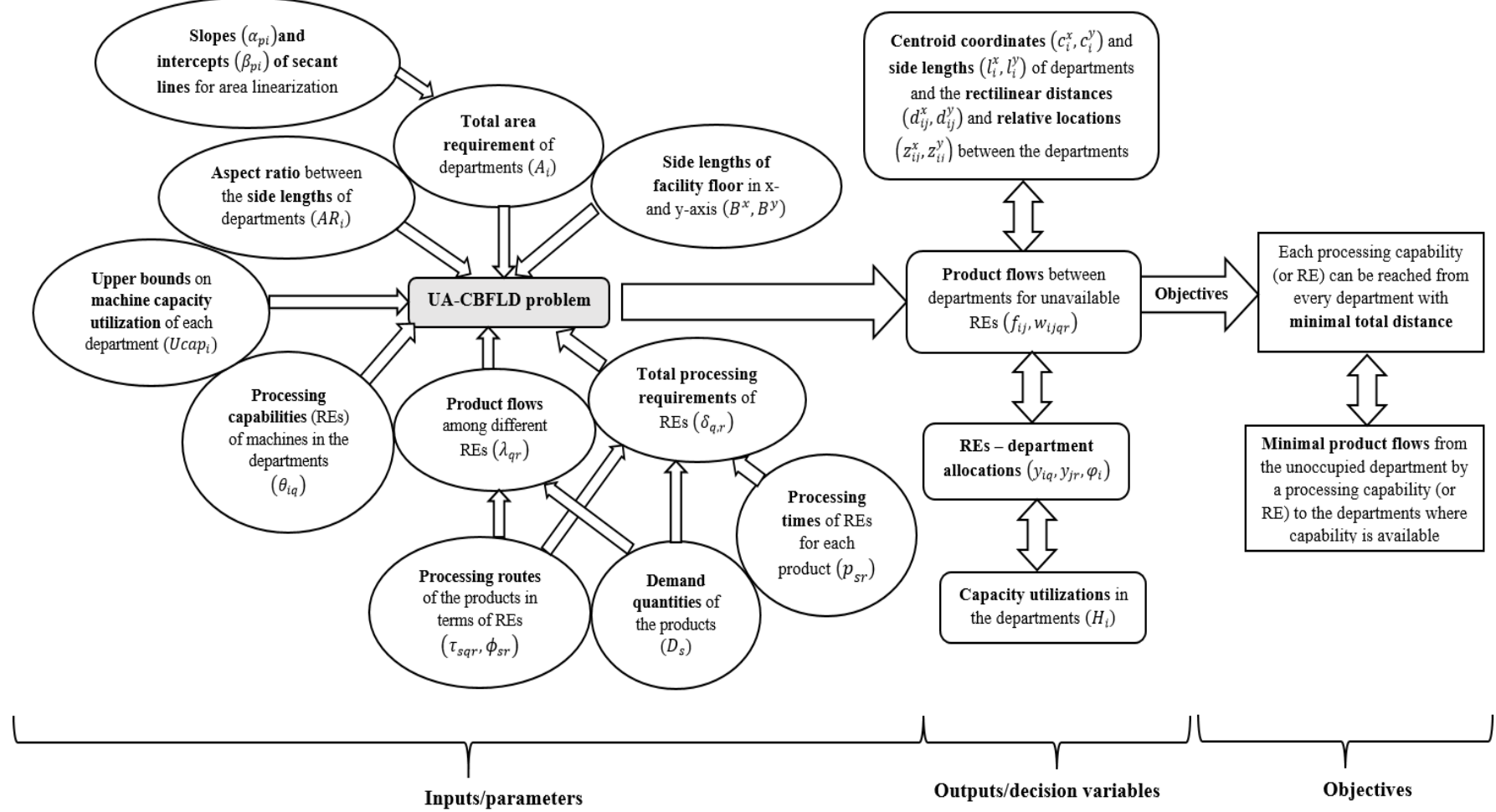


Figure 4.2 The key components of the proposed UA-CBFLD problem

Table 4.1 Mathematical nomenclature for the MINLP model of the UA-CBFLD problem

<i>Indices & Sets</i>	
$(i, j) \in N$	Set of departments in the facility layout
$p \in n_i$	Set of secant lines used in the inner approximation method for department i
$(q, r) \in R$	Set of Resource Elements (REs)
$s \in S$	Set of manufactured products
<i>Parameters</i>	
B^x	Side length in the x -axis direction of the facility floor
B^y	Side length in the y -axis direction of the facility floor
A_i	Total area requirement of the department i
AR_i	Aspect ratio between the side lengths of department i
D_s	Demand quantity for product s
τ_{sqr}	Number of transitions from the RE q to r in the process route of product s
ϕ_{sr}	Number of availabilities of the RE r in the process route of product s
λ_{qr}	Product flow quantities between RE q and RE r
θ_{iq}	1, if department i has processing capabilities including the RE q ; 0 otherwise
p_{sr}	Processing time of RE r for product s
$\delta_{q,r}$	Total processing requirement (in time units) of RE q or r
$Ucap_i$	Upper bound on the machine capacity utilization of department i
$Lcap_i$	Lower bound on the machine capacity utilization of department i
M	An arbitrarily chosen big positive number
<i>Additional auxiliary parameters for linearization of department area constraint</i>	
α_{pi}	The slope of secant line p for department i
β_{pi}	The constant/intercept of secant line p for department i
<i>Decision variables</i>	
f_{ij}	Product flow between departments i and j
c_i^x	Centroid coordinate of department i in the x -axis direction
c_i^y	Centroid coordinate of department i in the y -axis direction
l_i^x	Half of the side length of department i in the x -axis direction
l_i^y	Half of the side length of department i in the y -axis direction
d_{ij}^x	Rectilinear distance between the centroids of departments i and j in the x -axis direction
d_{ij}^y	Rectilinear distance between the centroids of departments i and j in the y -axis direction
z_{ij}^x	Relative locations of departments i and j in the x -axis direction
z_{ij}^y	Relative locations of departments i and j in the y -axis direction
y_{iq}	1, if the RE q is used by department i ; 0 otherwise
y_{jr}	1, if the RE r is used by department j ; 0 otherwise
w_{ijqr}	Product flow from department i to j for the RE q of department i and the RE r of department j
H_i	Total capacity usage (in time units) of department i
φ_i	1, if department i uses any RE; 0 otherwise
<i>Additional auxiliary variables for constraint linearization</i>	
z_{ijqr}	1, if there is a product flow from department i to j between the REs q and r ; 0 otherwise
Δ_{iq}	Capacity utilization (in time units) of department i for the RE q

By using the objective function in Equation (4.3), total cost of product transfer operations that is frequently formulated as a function of the distance and the amount of product flows between departments is minimized.

$$\text{Minimize } \sum_{i \in N} \sum_{j \in N \setminus \{i\}} f_{ij} \cdot (d_{ij}^x + d_{ij}^y) \quad (4.3)$$

Subject to;

$$c_j^x - l_j^x \geq c_i^x + l_i^x - B^x \cdot (1 - z_{ij}^x) \quad \forall (i, j) \in N, \quad i \neq j \quad (4.4)$$

$$c_j^y - l_j^y \geq c_i^y + l_i^y - B^y \cdot (1 - z_{ij}^y) \quad \forall (i, j) \in N, \quad i \neq j \quad (4.5)$$

$$z_{ij}^x + z_{ji}^x + z_{ij}^y + z_{ji}^y = 1 \quad \forall (i, j) \in N, \quad i \neq j \quad (4.6)$$

$$d_{ij}^x \geq |c_i^x - c_j^x| \quad \forall (i, j) \in N, \quad i \neq j \quad (4.7)$$

$$d_{ij}^y \geq |c_i^y - c_j^y| \quad \forall (i, j) \in N, \quad i \neq j \quad (4.8)$$

$$l_i^x \leq AR_i \cdot l_i^y \quad \forall i \in N \quad (4.9)$$

$$l_i^y \leq AR_i \cdot l_i^x \quad \forall i \in N \quad (4.10)$$

$$c_i^x + l_i^x \leq B^x \quad \forall i \in N \quad (4.11)$$

$$c_i^y + l_i^y \leq B^y \quad \forall i \in N \quad (4.12)$$

$$c_i^x - l_i^x \geq 0 \quad \forall i \in N \quad (4.13)$$

$$c_i^y - l_i^y \geq 0 \quad \forall i \in N \quad (4.14)$$

$$A_i = 4 \cdot l_i^x \cdot l_i^y \quad \forall i \in N \quad (4.15)$$

$$\sum_{i \in N} y_{iq} \cdot \theta_{iq} \geq 1 \quad \forall q \in R \quad (4.16)$$

$$y_{iq} \leq \theta_{iq} \quad \forall i \in N, \quad \forall q \in R \quad (4.17)$$

$$w_{ijqr} \leq \lambda_{qr} \cdot y_{iq} \cdot y_{jr} \quad \forall (i, j) \in N, \quad \forall (q, r) \in R, \quad q \neq r \quad (4.18)$$

$$\sum_{i \in N} \sum_{j \in N} \sum_{r \in R} \sum_{q \in R \setminus \{r\}} w_{ijqr} = \sum_{r \in R} \sum_{q \in R \setminus \{r\}} \lambda_{qr} \quad (4.19)$$

$$f_{ij} = \sum_{r \in R} \sum_{q \in R \setminus \{r\}} w_{ijqr} \quad \forall (i, j) \in N \quad (4.20)$$

$$\sum_{i \in N} \sum_{j \in N} w_{ijqr} = \lambda_{qr} \quad \forall (q, r) \in R, \quad q \neq r \quad (4.21)$$

$$\sum_{i \in N} H_i \geq \sum_{r \in R} \delta_r \quad (4.22)$$

$$\varphi_i \geq y_{iq} \quad \forall i \in N, \quad \forall q \in R \quad (4.23)$$

$$\sum_{q \in R} y_{iq} \cdot \delta_q \leq H_i \quad \forall i \in N \quad (4.24)$$

$$H_i \leq Ucap_i \cdot \varphi_i \quad \forall i \in N \quad (4.25)$$

$$\sum_{i \in N} y_{iq} \cdot H_i \geq \delta_q \quad \forall q \in R \quad (4.26)$$

$$c_i^x, c_i^y, l_i^x, l_i^y, d_{ij}^x, d_{ij}^y \geq 0 \quad \forall (i, j) \in N \quad (4.27)$$

$$z_{ij}^x, z_{ij}^y, y_{iq}, y_{jr}, \varphi_i \in \{0, 1\} \quad \forall (i, j) \in N, \quad i \neq j, \quad \forall (q, r) \in R \quad (4.28)$$

$$f_{ij}, H_i, w_{ijqr} \geq 0 \text{ and integer} \quad \forall (i, j) \in N, \quad \forall (q, r) \in R \quad (4.29)$$

Actually, it is a distance-based objective by using the centroid of department pairs. In contrast to the classical UA-FLD problem, this objective function in Equation (4.3) has a non-linear form since both of the product flows and rectilinear distances between departments are defined as decision variables. However, product flows between departments are usually taken as input data in the classical UA-FLD problems as mentioned before. The usage of this additional integer variable, i.e., f_{ij} , causes difficulties in the solution of the proposed UA-CBFLD problem because of the non-linear and complex structure of the objective function in Equation (4.3). On the other hand, by formulating such an objective function, processing necessities of many products can be fulfilled with minimal product transfers by allocating the machining capabilities suitably over the facility floor.

We need to emphasize here that the machines located in the departments are pre-determined by the facility layout designers in advance. Therefore, if a machine is located in a department, then all of its processing capabilities in terms of the REs will occupy in that department automatically.

Constraint sets in Equations (4.4) – (4.6) guarantee that departments i and j are separated in at least one direction (in the x -and y -axis directions). Additionally, the binary decision variables, i.e., z_{ij}^x, z_{ij}^y in Equation (4.6) prevent overlapping of departments in both x -and y -axis directions. In other words, constraint sets in Equations (4.4) – (4.6) aim to locate departments within the facility floor in a consistent way and also ensure that no two departments overlap by imposing a separation at least in the East-West or North-South direction. Constraint sets in Equations (4.7) – (4.8) bound the rectilinear distance between the centroids of each department pair $(i, j) \in N$ in both x -and y -axis directions with a positive product flow. According to the constraint sets in Equations (4.9) – (4.10), side lengths of each department must satisfy the maximum allowable ratio between its longest and shortest side. Constraint sets in Equations (4.11) – (4.14) maintain that the departments are positioned within the borders of the facility floor. In other words, they aim to determine the department centroids and side lengths properly so that each department is enclosed within the facility. The area requirement of each department is expressed through the non-linear constraint set in Equation (4.15). It should be emphasized here that total area requirement of department i , i.e., A_i is calculated by summing the machinery, stock and personnel work areas needed by all the machines that are located in department i . The remaining constraints in Equations (4.16) – (4.26) have already expressed in detail by Baykasoğlu et al. (2022). For more explanations about the meanings of these constraints, please refer to Baykasoğlu et al. (2022). In Equation (4.27), the centroid and side lengths of each department as well as the rectilinear distances between department pairs are defined as positive continuous variables. Moreover, constraint set in Equation (4.28) represents the binary variables for the relative locations of the departments, utilization of the REs by departments and also usage of the departments. Finally, product flows between numerous departments and their capacity usages are defined as integer variables in Equation (4.29).

4.1.2 Linearization of the Non-Linear Constraints and Objective Function

Some of the non-linear constraints as in Equations (4.7) – (4.8), (4.18) and (4.26) can be easily linearized and therefore, this section presents their linearized mathematical formulations in order to reduce the computational complexity of the proposed problem. For instance, since the Equations (4.7) – (4.8) include absolute value operator for calculating the rectilinear distances between department centroids, it has a non-linear form. Because, this absolute value function determines the positive deviation between departments' centroids which are predefined as decision variables of the proposed MINLP model. For that reason, Equations (4.7) – (4.8) are linearized as in the following Equations (4.30) – (4.31) and (4.32) – (4.33) (Mangasarian, 2007 and 2013).

$$d_{ij}^x \geq c_i^x - c_j^x \quad \forall (i,j) \in N, \quad i \neq j \quad (4.30)$$

$$d_{ij}^x \geq c_j^x - c_i^x \quad \forall (i,j) \in N, \quad i \neq j \quad (4.31)$$

$$d_{ij}^y \geq c_i^y - c_j^y \quad \forall (i,j) \in N, \quad i \neq j \quad (4.32)$$

$$d_{ij}^y \geq c_j^y - c_i^y \quad \forall (i,j) \in N, \quad i \neq j \quad (4.33)$$

Because of the constraint set in Equation (4.18) contains the product of two binary variables, it may also be linearized by using a supplementary binary variable, i.e., $z_{ijqr} = y_{iq} \cdot y_{jr}$ as in Equations (4.34) – (4.37) (Oral and Kettani, 1992).

$$w_{ijqr} \leq \lambda_{qr} \cdot z_{ijqr} \quad \forall (i,j) \in N, \quad (q,r) \in R, \quad q \neq r \quad (4.34)$$

$$z_{ijqr} \leq y_{iq} \quad \forall (i,j) \in N, \quad (q,r) \in R, \quad q \neq r \quad (4.35)$$

$$z_{ijqr} \leq y_{jr} \quad \forall (i,j) \in N, \quad (q,r) \in R, \quad q \neq r \quad (4.36)$$

$$z_{ijqr} \geq y_{iq} + y_{jr} - 1 \quad \forall (i,j) \in N, \quad (q,r) \in R, \quad q \neq r \quad (4.37)$$

Similarly, since the constraint set in Equation (4.26) consists of multiplication of a binary and general integer variable, it can also be converted into its linear form by defining an extra integer variable, i.e., $\Delta_{iq} = y_{iq} \cdot H_i$ as well as a huge positive number (big M), as in Equations (4.38) – (4.41) (Coelho, 2013).

$$\Delta_{iq} \leq y_{iq} \cdot M \quad \forall i \in M, \quad \forall q \in R \quad (4.38)$$

$$\Delta_{iq} \leq H_i \quad \forall i \in M, \quad \forall q \in R \quad (4.39)$$

$$\Delta_{iq} \geq H_i - (1 - y_{iq}) \cdot M \quad \forall i \in M, \quad \forall q \in R \quad (4.40)$$

$$\sum_{i \in N} \Delta_{iq} \geq \delta_q \quad \forall q \in R \quad (4.41)$$

We need to mention here again that the objective function formula given in Equation (4.3) includes product of an integer variable, i.e., f_{ij} with summation of the two continuous variables, i.e., $(d_{ij}^x + d_{ij}^y)$ and therefore, it has also a non-linear structure. As mentioned by Rubin (2019), if one can bound f_{ij} by a positive integer threshold value (not too large), this objective function can also be linearized theoretically as in Equations (4.42) – (4.48). However, this linearization may not be practical for the proposed UA-CBFLD problem since the product flows can take considerably high values. Thus, this linearization process may be impractical due to the significant increases in model dimension which necessitates a large number of decision variables and constraints during the linearization process. For this reason, instead of applying such an impractical linearization procedure, we proposed a heuristic decomposition-based iterative mathematical programming approach in the subsequent section to overcome this non-linear objective function.

$$\zeta_{ij} = f_{ij} \cdot (d_{ij}^x + d_{ij}^y) \quad \forall (i, j) \in N \quad (4.42)$$

$$\text{Minimize} \quad \sum_{i \in N} \sum_{j \in N \setminus \{i\}} \zeta_{ij} \quad (4.43)$$

Subject to;

$$\sum_{k=0}^K \omega_{ijk} = 1 \quad \forall (i, j) \in N \quad (4.44)$$

$$f_{ij} = \sum_{k=0}^K k \cdot \omega_{ijk} \quad \forall (i, j) \in N \quad (4.45)$$

$$\zeta_{ij} = \sum_{k=0}^K k \cdot \omega_{ijk} \cdot (d_{ij}^x + d_{ij}^y) \quad \forall (i, j) \in N \quad (4.46)$$

$$\omega_{ijk} \in \{0,1\} \quad \forall (i, j) \in N, \quad k \in K \quad (4.47)$$

$$\zeta_{ij} \geq 0 \quad \forall (i, j) \in N \quad (4.48)$$

where ζ_{ij} and ω_{ijk} represent auxiliary continuous and binary decision variables, respectively. Note that f_{ij} is bounded by a positive integer value (not too large), i.e., K , so that $f_{ij} \in \{0,1,\dots,K\}$ and then, we can introduce additional auxiliary binary variables, i.e., ω_{ijk} . However, since the resulting constraint set in Equation (4.46) contains product of a binary variable and summation of the two continuous variables, i.e., $\sigma_{ijk} = \omega_{ijk} \cdot (d_{ij}^x + d_{ij}^y)$, it has still a non-linear form and therefore, it should also be linearized as in the following Equations (4.49) – (4.53):

$$\zeta_{ij} = \sum_{k=0}^K k \cdot \sigma_{ijk} \quad \forall (i, j) \in N \quad (4.49)$$

$$\sigma_{ijk} \leq M \cdot \omega_{ijk} \quad \forall (i, j) \in N, \quad k \in K \quad (4.50)$$

$$\sigma_{ijk} \leq (d_{ij}^x + d_{ij}^y) \quad \forall (i, j) \in N, \quad k \in K \quad (4.51)$$

$$\sigma_{ijk} \geq (d_{ij}^x + d_{ij}^y) - M \cdot (1 - \omega_{ijk}) \quad \forall (i, j) \in N, \quad k \in K \quad (4.52)$$

$$\sigma_{ijk} \geq 0 \quad \forall (i, j) \in N, \quad k \in K \quad (4.53)$$

As mentioned previously, although this linearization procedure is theoretically correct, it is not generally applicable for solving large-sized UA-CBFLD problem instances in which high amounts of product flows arise between departments. In such a situation, model dimension will increase significantly because of the huge number constraints and decision variables formulated in Equations (4.42) – (4.48) and (4.49) – (4.53). Therefore, optimal solution cannot be provided within reasonable computational time by any optimization solver like CPLEX, Gurobi optimizer etc.

4.1.3 Polyhedral Inner-Approximation Method for Area Constraint Linearization

The non-linear department area constraint in Equation (4.15) makes the proposed UA-CBFLD problem difficult to solve optimally, because it comprises the product of two continuous decision variables, i.e., $l_i^x \cdot l_i^y$. Actually, this kind of non-linear constraints cannot be linearized exactly or directly without any approximation method due to the non-convex area calculation. For that reason, several linearization methods were developed in some previous researches to approximate this non-linear area constraint (Montreuil, 1990; Lacksonen, 1997; Meller et al., 1999; Sherali et al., 2003; Castillo and Westerlund, 2005; Konak et al., 2006). However, all of these methods mainly depend on a polyhedral outer-approximation, which generally causes smaller department areas than the needed actual areas (Xiao et al., 2017). Furthermore, these former methods cannot check the area approximation of each department under a pre-specified maximum deviation error. For that reason, a new polyhedral inner-approximation method was proposed by Xie et al. (2016) and Xiao et al. (2017) to formulate the department area constraint, which ensures that the actual area is bigger than or equal to the needed one at all times. In this thesis, we have also implemented this inner-approximation method since a little larger department area may frequently be a safer design option in many real-life manufacturing settings.

As depicted in Figure 4.3, this method employs some secant lines (instead of using tangential lines), which start from the point (x_1, y_1) representing the minimum allowable width and the maximum allowable height of the department. In fact, the given curve in Figure 4.3 is cut until the point (x_i^{max}, y_i^{min}) has been enclosed by the last secant line. Therefore, the ending point of a secant line can be computed by using

a pre-specified error ratio, $ER_i \geq \frac{(A_{actual}-A_i)}{A_i}$, which is able to control the maximum deviation error of the area approximation by finding the locations of starting and ending points of the secant lines. Moreover, it should also be noted here that this inner approximation method can utilize different number of secant lines for different department areas whereas the aforementioned traditional tangent approximation methods generally use the same number of tangential lines for all the department areas. Therefore, different values can be defined for the maximum area approximation deviation of each department in the inner approximation method.

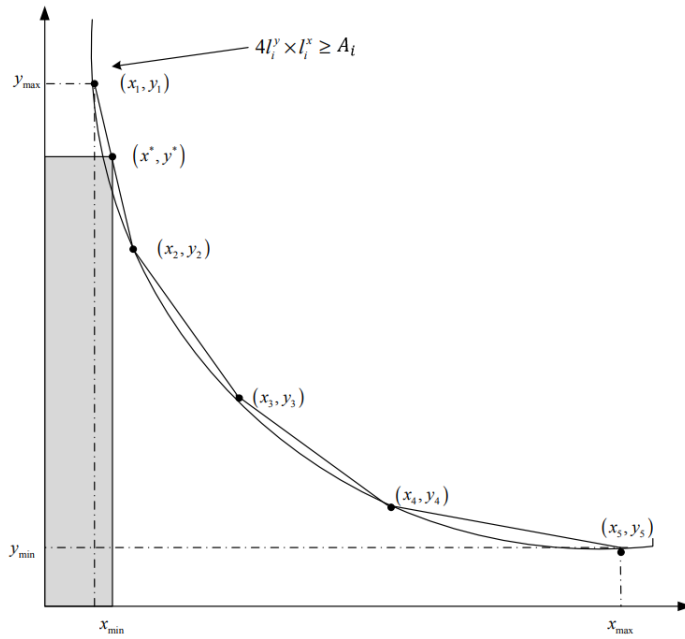


Figure 4.3 Secant lines of the inner-approximation technique for the non-convex area linearization (Xie et al., 2016)

After determining these starting/ending points, slopes and intercepts of the secant lines should also be calculated within a maximum allowed approximation error ratio (ER_i). To do this, minimum and maximum side lengths of the departments are first calculated for x and y -axis as formulated in Equations (4.54) – (4.57). Actually, these minimum/maximum side lengths are computed based on side lengths of the facility floor (B^x, B^y), aspect ratio (AR_i) and total area requirements of the departments (A_i).

$$x_{min} = \min \{B^x, \sqrt{A_i/AR_i}\} \quad (4.54)$$

$$y_{min} = \min\{B^y, \sqrt{A_i/AR_i}\} \quad (4.55)$$

$$x_{max} = \min\{B^x, \sqrt{A_i \cdot AR_i}\} \quad (4.56)$$

$$y_{max} = \min\{B^y, \sqrt{A_i \cdot AR_i}\} \quad (4.57)$$

Afterward, the minimum number of secant lines, i.e., n_i that guarantees the maximal allowable approximation error ratio (ER_i) is calculated for each department as in Equation (4.58). Thus, the maximum approximation deviation from the actual area of each department cannot be greater than the pre-specified error ratio, ER_i .

$$n_i \geq \frac{\ln(x_i^{max}) - \ln(x_i^{min})}{\ln\left(1 + 2 \cdot ER_i + 2 \cdot \sqrt{ER_i + ER_i^2}\right)} + 1 \quad \forall i \in N \quad (4.58)$$

In the next step, the points of secant lines, i.e., $(x_1, y_1), (x_2, y_2), \dots, (x_p, y_p)$ are computed recursively through Equation (4.59) until $x_p \geq x_{max}$ is satisfied. In other words, the point (x_i^{max}, y_i^{min}) has been covered by the last secant line.

$$\begin{cases} x_p = x_{min} & \forall p = 1 \\ x_p = \left(1 + 2 \cdot ER_i + 2 \cdot \sqrt{ER_i + ER_i^2}\right) \cdot x_{p-1} & \forall p = 2, 3, \dots, n_i + 1 \\ y_p = A_i/x_p & \forall p = 1, 2, \dots, n_i + 1 \end{cases} \quad (4.59)$$

After computing the points of a set of secant lines, i.e., $y = \alpha_{pi} \cdot x + \theta_{pi} \quad \forall p \in n_i, \forall i \in N$, the slopes and constants of these secant lines, i.e., α_{pi} and β_{pi} , are calculated via Equations (4.60) – (4.61), respectively.

$$\alpha_{pi} = -\frac{A_i}{x_p \cdot x_{p+1}} \quad \forall p \in n_i, \forall i \in N \quad (4.60)$$

$$\beta_{pi} = \frac{A_i}{x_p} + \frac{A_i}{x_{p+1}} \quad \forall p \in n_i, \forall i \in N \quad (4.61)$$

Finally, constraint set in Equation (4.62) correspond to the linearization of the department area requirement in Equation (4.15) by using a set of secant lines whose slopes and intercepts are determined based on the Equations (4.60) – (4.61).

$$2 \cdot l_i^y \geq 2 \cdot \alpha_{pi} \cdot l_i^x + \beta_{pi} \quad \forall p \in n_i, \forall i \in N \quad (4.62)$$

4.1.4 Heuristic Decomposition-Based Iterative Mathematical Programming Approach

In order to cope with the highly non-linear structure of the objective function of the proposed MINLP model and also provide good quality solutions within reasonable times, we implemented a heuristic decomposition-based iterative mathematical programming approach. As mentioned before in Sections 4.1.1 and 4.1.2, all of the non-linear constraint functions are linearized properly in order to obtain fully linearized constraint functions. However, the linearization process of the objective function is still a problematic issue because of the large amounts of product flows between departments. Because, linearization of the objective function results in significant increases in the model dimension as highlighted previously.

Furthermore, it is also proven in the computational study (see the subsequent Section 4.2) that the lower bound of the proposed MINLP model doesn't change for a long time and therefore, optimization process cannot terminate under reasonable computing time. Therefore, one can achieve just a best found solution under a runtime limit via standard non-linear programming solvers. In other words, global or local optimal solution cannot be provided by solving the proposed MINLP model due to this lower bound problem and never ending optimization process. For this reason, instead of linearizing the objective function, we preferred to decompose the proposed UA-CBFLD problem into two sub-problems namely, (i) classical UA-FLD problem and (ii) Capability-based product flow problem. Fortunately, the developed MINLP model in Section 4.1.1 can be easily decomposed into two sub-models for the aforementioned sub-problems. Because, there is no interconnection between the constraints of these sub-models. In other words, there is no interaction between the decision variables of these two sub-models and therefore, it is not possible to formulate any logical or technical constraint for the relationship between these two sub-problems. Indeed,

constraints of the submodel-1 (see in Equations 4.4 – 4.15) are formulated for the traditional UA-FLD problem only, whereas the constraints of the submodel-2 (see in Equations 4.16 – 4.26) are just related to the capability-based product flow problem. Actually, these sub-models are connected with each other in the objective function only. In this non-linear objective function, the product flows between departments, i.e., f_{ij} can be provided by solving the submodel-2 of the capability-based product flow problem, which can consider the machining capabilities covered by each department. On the other hand, other terms in this non-linear objective function, i.e., distances between department pairs (d_{ij}^x, d_{ij}^y) , can be obtained from solution of the submodel-1 of the UA-FLD problem. Thus, the proposed UA-CBFLD problem which is a new extension of the classical UA-FLD problem has a decomposable structure.

Submodel-1 (Equations 4.3 and 4.4 – 4.15, 4.27 - 4.28): Unequal-area facility layout design model with inner approximation method.

Submodel-2 (Equations 4.3 and 4.16 – 4.26, 4.28 – 4.29): Capability-based product flow model.

After the decomposition phase of the proposed approach, a feasible initial flow matrix, which shows the product flows between departments is provided by solving the submodel-2 first. It should be noted here that since there is no initial facility layout beforehand, the distances between departments' centroids, i.e., $(d_{ij}^x + d_{ij}^y)$ are excluded from the objective function in this second step.

In the third step, an iterative optimization process is initiated by solving the submodel-1 with this initial product flow matrix, which is obtained from the previous step-2. Then, an initial facility layout design can be generated by using this capability-based product flow matrix. After that, submodel-2 is solved again by using this initial layout design parameters (i.e., distances between departments' centroids) as input data to update the product flow matrix so that the departments' location information is considered. Then, submodel-1 is solved again by using this updated product flow matrix to improve the previously generated facility layout design so as to take into account the processing capabilities of machines located in the departments.

Finally, the proposed approach will be terminated if the objective function value of submodel-1 doesn't change or improve yet. In other words, the proposed iterative mathematical programming approach will continue as long as the objective value of submodel-2 converges to a better value. We need to mention here that the number of required iterations to provide the final facility layout design option may vary from one problem instance to another because of defining such a stopping criterion. The main steps of the proposed approach are summarized in Table 4.2. Besides, it is also outlined step by step in Figure 4.4.

Table 4.2 Main steps of the heuristic decomposition-based iterative mathematical programming approach

<i>Step-1:</i>	Decomposition phase: Submodel-1 (UA-FLD problem) & Submodel-2 (Capability-based product flow problem)
<i>Step-2:</i>	Find a feasible initial product flow matrix. First solve submodel-2 by excluding the total distance between departments' centroids, i.e., $(d_{ij}^x + d_{ij}^y)$ and then, obtain an initial part flow matrix, i.e., f_{ij} without departments' location information.
<i>Step-3:</i>	Perform iterative optimization. Solve submodel-1 by using the part flow matrix as input data, which is obtained from step-2 and then, provide a feasible facility layout design by considering capability-based product flows.
<i>Step-4:</i>	Solve submodel-2 again by using the distances between departments' centroids as input data, which is obtained from step-3 and then, update part flow matrix with departments' location information.
<i>Step-5:</i>	Solve submodel-1 again by using the updated part flow matrix as input data, which is obtained from step-4 and then, provide a facility layout design, which considers machining capabilities in the departments.
<i>Step-6:</i>	Perform steps 3 – 5 until objective function value of submodels-1 doesn't change or improve yet.

4.2 Computational Study on UA-CBFLD

In this section, a hypothetical example as well as a real-life application study are given in order to illustrate the proposed UA-CBFLD problem and verify the results of heuristic decomposition-based iterative mathematical programming approach.

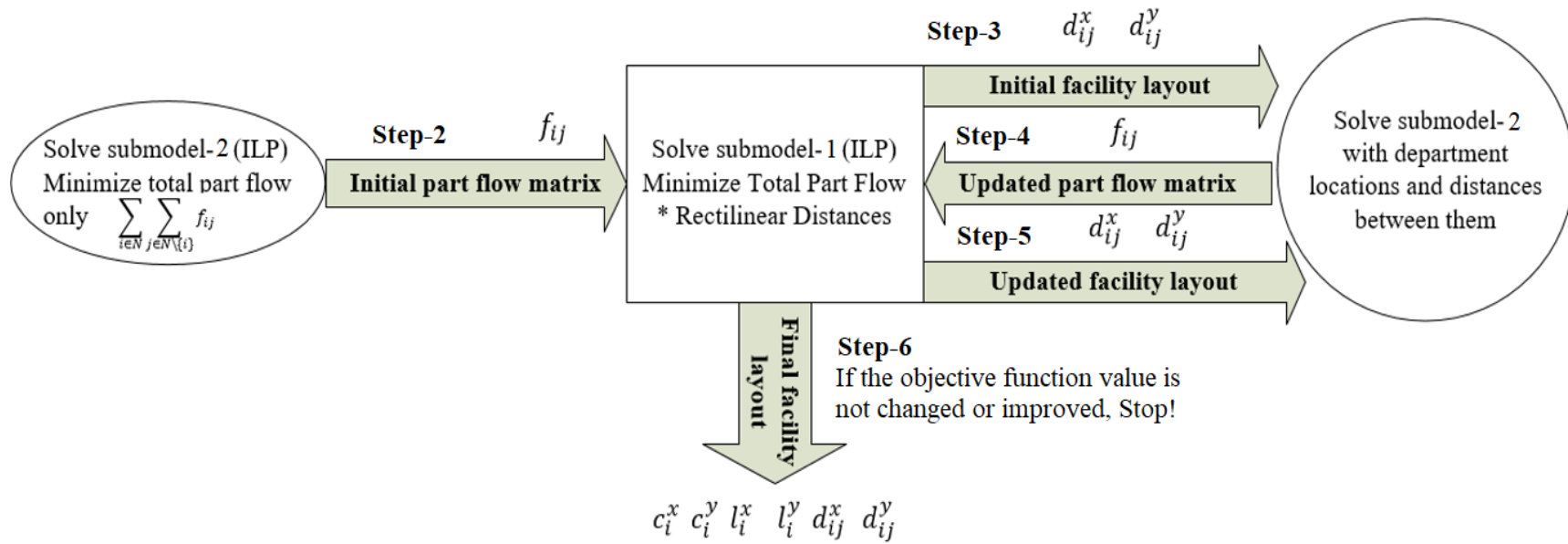


Figure 4.4 Schematic depiction of the heuristic decomposition-based iterative mathematical programming approach

4.2.1 An Illustrative Example with Diverse Capability Overlap Cases

An illustrative numerical example is first presented to analyse the impacts of various machining capability overlap cases on the optimization outcomes. In fact, this numerical example is composed of totally 8-departments in a facility floor whose side lengths, i.e., B^x and B^y are fixed to 360 and 200 meters in x-and y-axis directions, respectively. The area requirements of these departments are also known in advance and set to 12000, 8000, 6000, 12000, 8000, 12000, 12000 and 2000 square feet, respectively. It is also assumed that the machines located in these departments have 8 different operation capabilities that are described by the REs. Therefore, each department has also a set of processing capabilities (or REs) according to the machines located within its boundaries. As mentioned before, since the capability overlaps of the machines located in each department may influence the facility layout design, five different cases namely no, low, medium, high and very high overlap are considered to examine the impact on the amounts of product flows between departments and total cost of material handling operations. Thus, the department-capability data related to these REs is shown for each case in Figure 4.5.

Actually, stars (*) in Figure 4.5 demonstrates which department have which processing capabilities (or REs). So, if a star is available for a department-RE pair, this department has the machines including the relevant capability. For instance, machines in each department has just only one capability in no overlap case. Hence, each capability is available in only one department. Thus, this situation corresponds to no overlap case. Instead, each capability or RE is enclosed by more than one department in the low, medium, high and very high overlap situations. Likewise, one department has more than one capability in these situations. For that reason, if one or more capabilities are included by more than one department, capability overlap situation will exist. In a nutshell, if the number of stars rises in these department-RE matrices, the level of department-capability overlap will also be larger. The amounts of product demands, process plan (by means of the REs) and the processing times of 8-different products are depicted in Table 4.3.

		<i>REs (No overlap)</i>							
<i>Departments</i>		1	2	3	4	5	6	7	8
1		*							
2			*						
3				*					
4					*				
5						*			
6							*		
7								*	
8									*

		<i>REs (Low overlap)</i>							
<i>Departments</i>		1	2	3	4	5	6	7	8
1		*		*					
2			*			*			
3		*		*					
4					*		*		
5						*		*	
6							*		*
7			*					*	
8					*				*

		<i>REs (Medium overlap)</i>							
<i>Departments</i>		1	2	3	4	5	6	7	8
1		*		*		*			
2		*	*			*			
3		*		*			*		
4					*		*	*	
5						*		*	*
6				*			*		*
7			*		*			*	
8			*		*				*

		<i>REs (High overlap)</i>							
<i>Departments</i>		1	2	3	4	5	6	7	8
1		*	*	*		*			
2		*	*	*		*			
3		*		*	*		*		
4					*	*	*	*	
5						*		*	*
6				*			*	*	*
7			*		*			*	*
8		*	*		*				*

		<i>REs (Very high overlap)</i>							
<i>Departments</i>		1	2	3	4	5	6	7	8
1		*	*	*	*	*			
2		*	*	*		*	*		
3		*		*	*	*	*		
4					*	*	*	*	*
5		*				*	*	*	*
6			*	*			*	*	*
7			*	*	*			*	*
8		*	*		*			*	*

Figure 4.5 Several capability overlap situations for the machines located in the departments

Based on this information, product flows among several REs and total processing requirement of every RE are calculated by Equations (4.1) – (4.2) and given in Table 4.4. Lastly, we supposed that all departments have the same maximal limitation on their machine capacity usages (in time units) and its value is taken as 14500 minutes.

In order to apply the inner approximation method, minimum/maximum side lengths of each department are first calculated as given in Table 4.5 based on the side lengths of the facility floor, aspect ratio and total area requirements (see Equations 4.54 – 4.57). It is assumed that the aspect ratio, i.e., AR_i is equal to “2.0” for every department. Moreover, the value of maximal allowable approximation error ratio, i.e., ER_i is determined as 1.0% for each department. Therefore, the needed minimum number of secant lines, i.e., n_i according to these maximum allowable approximation error and aspect ratios were set to “6” for each department as well.

Table 4.3 Manufacturing routes, processing times and demand quantities of the products

Products	Demand quantity	RE-based process plans (Processing times in minutes)
1	50	RE1(2)-RE5(3)-RE8(1)-RE4(3)-RE6(5)-RE2(3)-RE7(6)-RE5(1)
2	100	RE2(3)-RE1(5)-RE3(3)-RE8(2)-RE6(4)-RE4(1)-RE2(5)-RE1(6)
3	200	RE3(1)-RE1(2)-RE7(3)-RE8(4)-RE3(5)-RE5(2)-RE2(1)-RE4(3)-RE1(2)
4	150	RE1(3)-RE8(2)-RE6(5)-RE7(3)-RE2(4)-RE6(1)-RE3(3)-RE7(1)-RE4(1)
5	80	RE1(2)-RE2(1)-RE5(4)-RE3(3)-RE5(2)-RE4(3)-RE8(1)-RE7(2)
6	120	RE3(1)-RE2(4)-RE8(1)-RE1(3)-RE6(4)-RE3(2)-RE4(1)-RE7(2)
7	70	RE2(3)-RE3(1)-RE6(2)-RE1(5)-RE4(3)-RE5(2)-RE2(1)-RE8(3)
8	180	RE4(2)-RE7(5)-RE1(2)-RE3(2)-RE5(4)-RE6(2)-RE8(5)-RE5(3)-RE1(3)

Table 4.4 Product flows between different REs and their total processing requirements (in min)

REs	Resource Elements (REs)							
	1	2	3	4	5	6	7	8
1	-	80	280	70	50	120	200	150
2	100	-	70	200	80	150	50	190
3	200	120	-	120	380	70	150	100
4	200	100	0	-	70	50	120	80
5	180	70	80	80	-	180	0	50
6	70	50	150	100	0	-	150	180
7	180	150	0	0	50	0	-	200
8	120	0	200	50	180	100	80	-
Total processing requirements (min.)	4220	2590	2980	1930	2480	2530	2800	2660

Table 4.5 Minimum/maximum side lengths of the departments

Departments	x_{min}/y_{min}	x_{max}/y_{max}
1	77.45966692	154.9193338
2	63.2455532	126.4911064
3	54.77225575	109.5445115
4	77.45966692	154.9193338
5	63.2455532	126.4911064
6	77.45966692	154.9193338
7	77.45966692	154.9193338
8	31.6227766	63.2455532

Finally, after computing the points of these secant lines, their slopes and intercepts are computed as in Table 4.6 so as to utilize into the constraint set in Equation (4.62). It is obviously realized in Table 4.6 that the slopes of each secant line is equal to each other for every department since the same value of aspect ratio is defined. On the other hand, the intercepts of these secant lines varied from one department to another.

Table 4.6 The slopes and intercepts of the secant lines for the illustrative example

Secant lines, $p \in \forall n_i$	Slopes (α_{pi})	Intercepts (β_{pi})							
		Dept.1	Dept.2	Dept.3	Dept.4	Dept.5	Dept.6	Dept.7	Dept.8
1	-1.63800497	281.8	230.09	199.26	281.8	230.09	281.8	281.8	115.04
2	-1.09872	230.79	188.44	163.20	230.79	188.44	230.79	230.79	94.22
3	-0.73698	189.02	154.33	133.66	189.02	154.33	189.02	189.02	77.17
4	-0.49434	154.81	126.40	109.47	154.81	126.40	154.81	154.81	63.20
5	-0.33159	126.79	103.52	89.65	126.79	103.52	126.79	126.79	51.76
6	-0.22242	103.84	84.79	73.43	103.84	84.79	103.84	103.84	42.39

Based on all of these information, the proposed MINLP model is first solved with the previously given data by the non-linear programming solver of Gurobi 9.1 in Python 3.8.8 modelling and development environment on an Intel Corei7 2 GHz IBM PC. Then, the proposed heuristic decomposition-based iterative mathematical programming approach is also implemented by using Gurobi MIP solver. The optimization results of the proposed MINLP model are obtained for different capability overlapping cases under one hour runtime limit as shown in Tables 4.7 - 4.8.

Table 4.7 Comparative results provided by the proposed MINLP model and the heuristic decomposition-based approach

Capability overlap cases	The proposed MINLP model				The proposed heuristic decomposition approach			
	Best found objective	Objective bound	Opt. gap	CPU Time	Best found objective	Objective bound	Opt. gap	CPU Time
No overlap	1.008.928,29	1.008.928,29	0%	1521.26	1.008.928,29	1.008.928,29	0%	1061.71
Low overlap	578.872,55	0	100%	3600	528.284,46	528.284,46	0%	474.26
Medium overlap	165.718,04	0	100%	3600	146.434,20	146.434,21	0%	552.94
High overlap	43.871,32	0	100%	3600	43.871,32	43.871,32	0%	1701.37
Very High overlap	0	0	0%	70.96	0	0	0%	2.03

According to Table 4.7, optimal solutions can be obtained for just no overlap and very high overlap cases by making use of the proposed MINLP model. Unfortunately, just feasible solutions can be acquired for the other capability overlap cases under one hour runtime limit since the lower bound of the proposed MINLP model doesn't change for a long time and therefore, optimization process cannot terminate. In other words, the lower bound of the objective value is always equal to "0" throughout the optimization process and therefore, optimality gaps are equal to 100% all the time.

This problematic issue forces us to apply a heuristic decomposition-based iterative mathematical programming approach as mentioned before in the previous section. Fortunately, as it is obviously seen in Tables 4.7 - 4.8 that superior results can be produced via the heuristic decomposition-based iterative mathematical programming approach under reasonable computing times. In other words, although the proposed MINLP model is run for a relatively long time limit (e.g., one hour runtime limit) for the low, medium and high overlap situations, it can just find local optimal solutions whose objective values are inferior to the results of the heuristic decomposition-based iterative mathematical programming approach.

In a nutshell, better quality solutions are generated by using the heuristic decomposition-based solution approach with less computational times (see Tables 4.7 and 4.8). Optimization details of this heuristic decomposition-based solution approach including the solutions of submodel-1 and submodel-2 are presented in Table 4.8. In addition, solution details are also reported in Tables 4.9 - 4.11. The facility layouts generated by the heuristic decomposition-based solution approach under distinct capability overlap situations are displayed in Figure 4.6. According to Figure 4.6, different layout design alternatives can be produced under various capability overlap situations.

The resulted product flows among distinct departments are also presented in Table 4.9. The machines' total capacity usages in each department are also involved in Table 4.9. This table shows that there may be a huge amount of product flows among several departments in case of no overlap situation. Actually, product flows among departments are equal to the product flow quantities between distinct REs, which is formerly presented in Table 4.4. This is because, each department has only one RE and every RE is covered by a single department in no overlap situation. Therefore, product flows between distinct REs converted into the product flows among departments automatically. Please also note that the machines in each department will be actively used with the maximal capacity usage in this no overlap case. On the other hand, product flow quantities among distinct departments will reduce when the machining capability overlaps are augmented.

Table 4.8 Optimization details of the proposed heuristic decomposition-based solution approach

Capability Overlap Cases	Iteration No.	Optimal solution of submodel-2	Optimal solution of submodel-1	CPU Time
No overlap	1	6200	-	0.38
	2	-	1.008.928,29	1061.04
	3	1.008.928,29	-	0.29
Low overlap	1	4760	-	0.33
	2	-	633.715,11	256.69
	3	530.191,19	-	0.31
	4	-	528.284,46	135.64
	5	499.882,73	-	0.34
	6	-	499.877,38	80.66
	7	499.877,38	-	0.29
Medium overlap	1	1910	-	0.35
	2	-	185.285,35	14.07
	3	146.447,54	-	0.3
	4	-	146.434,20	486.67
	5	146.434,20	-	0.35
High overlap	1	580	-	0.38
	2	-	62.152,43	1242.86
	3	46.583,22	-	0.39
	4	-	46.480,95	304.65
	5	43.871,55	-	0.34
	6	-	43.871,32	166.03
	7	43.871,32	-	0.34
Very high overlap	1	0	-	0.55
	2	-	0	1.48

As depicted in Table 4.9, product flow quantities among distinct departments are decreased steadily in low, medium, high and very high capability overlap situations. Conversely, the machines' worst-case capacity usages will increase for the higher capability overlap situations. Actually, these capacity usages are strictly related to the product flow quantities among the departments. In other words, if a large quantity of products are transferred to any department from other departments, its machine capacity utilizations will increase. In the very high overlap situation, there is no product flow between different departments because of very high machining capability overlaps, i.e. the machines located in each department can fulfil all the allocated processing requirements with their existing capacity. Thus, there is no need to transfer any product between different departments and therefore, the layout score for very high overlap case will be equal to "0" (see in Tables 4.7 and 4.8).

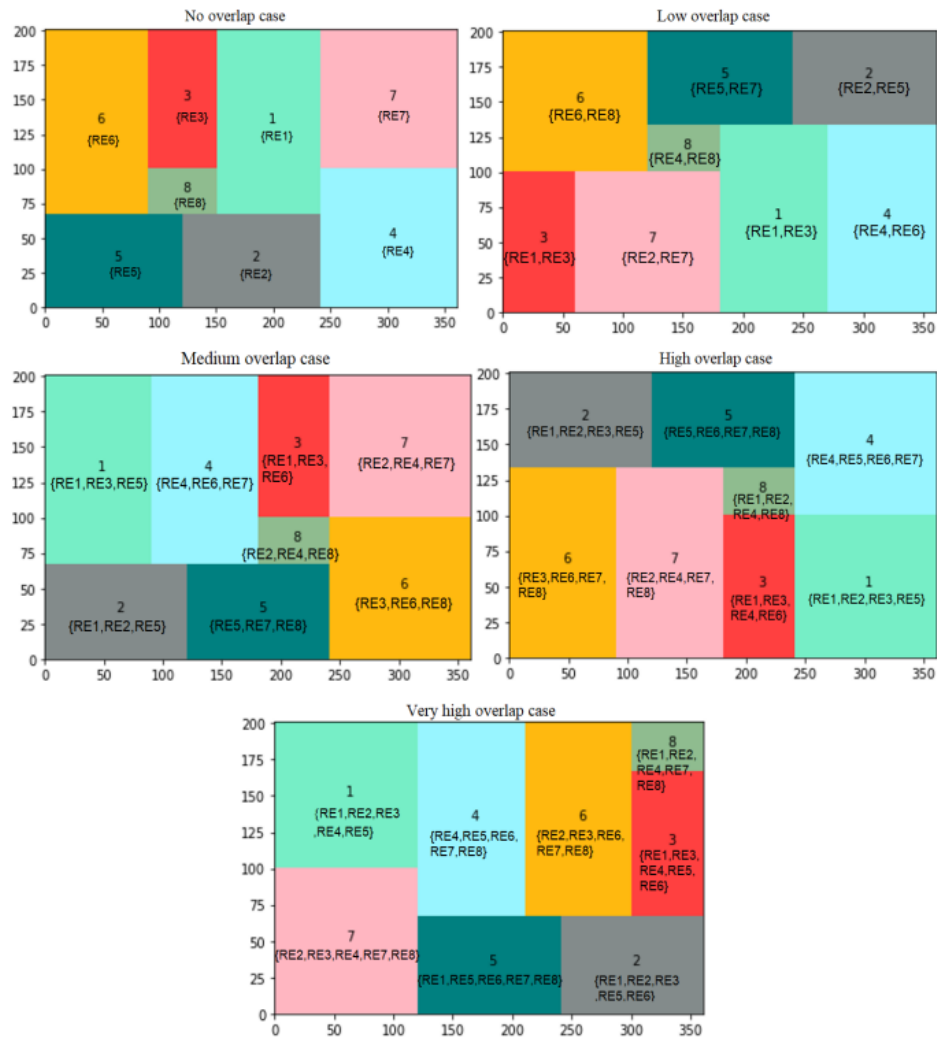


Figure 4.6 Unequal-area facility layout designs under distinct capability overlap situations

It is also recognized that all the machines should become in active statue in the production operations, excluding the high and very high overlap situations. In fact, machines in department-2 will not be utilized under the high overlap situation. Similarly, machines located in departments-5 and 7 are also inactive or unused. Therefore, they have no capacity utilization or their worst case capacity usages are equal to “0” as seen in Table 4.9. Due to the unused machines in departments-5 and 7, capacity usages of the other actively used machines will be comparatively higher in high and very high capability overlap situations.

Table 4.9 Product flows among various departments and their capacity usages under distinct overlap situations

Departments	Departments (No overlap case)								Worst case capacity utilization
	1	2	3	4	5	6	7	8	
1	-	80	280	70	50	120	200	150	4220
2	100	-	70	200	80	150	50	190	2590
3	200	120	-	120	380	70	150	100	2980
4	200	100	0	-	70	50	120	80	1930
5	180	70	80	80	-	180	0	50	2480
6	70	50	150	100	0	-	150	180	2530
7	180	150	0	0	50	0	-	200	2800
8	120	0	200	50	180	100	80	-	2660

Departments	Departments (Low overlap case)								Worst case capacity utilization
	1	2	3	4	5	6	7	8	
1				380	430			250	14500
2		150		330					5070
3			480				550		7200
4	420	50		150					4460
5	260				50			330	5280
6					150	280			5190
7			350				200	390	5390
8	320				450		100	130	4590

Departments	Departments (Medium overlap case)								Worst case capacity utilization
	1	2	3	4	5	6	7	8	
1	1170			530					14500
2		330							9290
3			190					510	9730
4	180			420					7260
5					560			80	7940
6						800			8170
7							200		5390
8			540		70			620	7180

Departments	Departments (High overlap case)								Worst case capacity utilization
	1	2	3	4	5	6	7	8	
1	1690	-							14500
2	-	-	-	-	-	-	-	-	0
3		-	950					50	11660
4		-		296					7210
5		-			1144			180	10470
6		-				450			9340
7		-					500		7580
8		-	150		200			590	11400

Departments	Departments (Very high overlap case)								Worst case capacity utilization
	1	2	3	4	5	6	7	8	
1	1780				-		-		14200
2		590			-		-		14500
3			150		-		-		4460
4				710	-		-		10470
5	-	-	-	-	-	-	-	-	0
6					-	650	-		13560
7	-	-	-	-	-	-	-	-	0
8					-			232	14200
								0	

For more details on the product flows between different departments, Table 4.10 is organized for the medium overlap situation only. With the help of this table, details of the product flows among distinct departments are depicted by incorporating the

processing capability information (or REs) as well. We should mentioned here that the product flows inside the departments' themselves are not given in Table 4.10. For example, totally 330 units (from RE-1 to RE-2: 80 units; from RE-2 to RE-1: 100 units; from RE-2 to RE-5: 80 units; from RE-5 to RE-2: 70 units) of products are transferred within the department-2 itself as it is also seen in Table 4.9. Table 4.10 shows the detailed product flows between just different departments including the RE information. For example, huge amount of product flows will arise from department-1 to department-4, because department-1 does not involve REs 6 and 7. Since these REs are not covered by department-1, totally 530 parts are shipped from its existing REs 1, 3 and 5 to department-4. Besides these available REs 1, 3 and 5, if department-1 had also included REs 6 and 7, no product transfer is required from machine-1 to machine-4.

Table 4.10 Details of product flows between departments under medium capability overlap case

From department	From RE	To department	To RE	Amounts of product flows	Total product flow
1	1	4	7	200	
1	3	4	7	150	
1	5	4	6	180	530
3	1	8	4	70	
3	1	8	8	150	
3	3	8	2	120	
3	3	8	4	120	
3	6	8	2	50	510
4	7	1	1	180	180
5	5	8	4	80	80
8	2	3	3	70	
8	2	3	6	150	
8	4	3	1	200	
8	8	3	1	120	540
8	4	5	5	70	70

Finally, the REs utilized by every department are also given for the high capability overlap situation in Table 4.11. Since department-2 was not active, it cannot utilize any RE as an expected result. Although departments-4, 6 and 7 have already four different REs (see in Figure 4.5), they only used three of them during the production process as demonstrated in Table 4.11. Oppositely, departments-1, 3, 5 and 8 have utilized all of their existing REs.

Table 4.11 REs utilized by the departments under high capability overlap situation

Departments	Utilized REs
1	RE1, RE2, RE3, RE5
2	-
3	RE1, RE3, RE4, RE6
4	RE4, RE5, RE7
5	RE5, RE6, RE7, RE8
6	RE3, RE7, RE8
7	RE2, RE4, RE7
8	RE1, RE2, RE4, RE8

4.2.2 Application Study in a Real-Life Manufacturing System

Since the proposed UA-CBFLD problem has not been handled previously in the existing literature, we adapted a real-life cellular manufacturing system which was formerly introduced by Baykasoğlu and Gindy (2000) to show its validity and also practicality of the heuristic decomposition-based iterative mathematical programming approach. Indeed, the present shop floor was organized by facility designers as a cellular layout without considering the machines' area requirements in each cell.

Actually, they assumed that each machine in a cell has the same area requirement as in the traditional equal-area facility layout design problems. The existing cellular layout of the facility has already shown by Baykasoğlu and Gindy (2000) as in Figure 4.7. The product assignments to each manufacturing cell were also displayed in the same figure. As it is obviously seen from Figure 4.7 that there are 7-different machines inside the facility but some machines have multiple copies. Thus, there are totally 12 machines available over the shop floor where copied machines include exactly the same processing capabilities or REs.

In this section, it is aimed to redesign this shop floor as a functional layout by considering machines' processing capabilities (or REs) as well as their unequal area requirements. Because, the facility designers are not fully fulfilled with the performance of this current cellular layout due to the large amount of material transfers between distinct cells and low cell capacity usages. Therefore, they seek to a better layout option, which is also able to handle the machines' unequal area requirements.

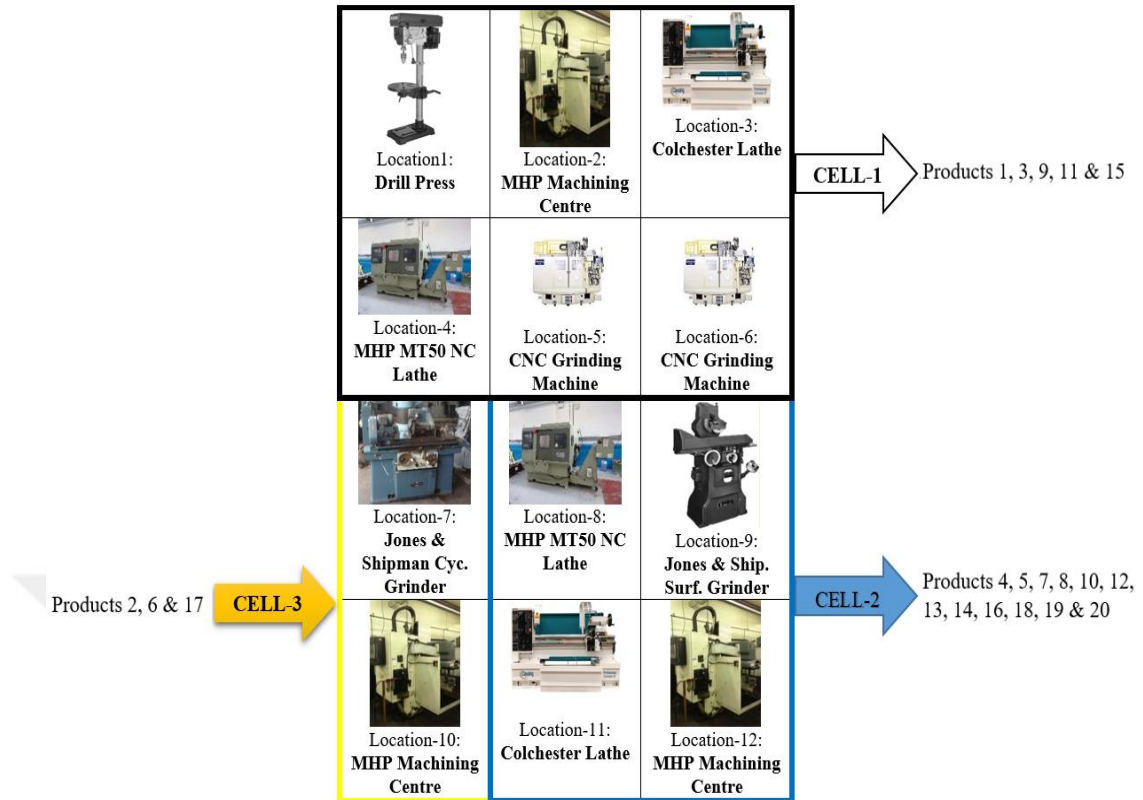


Figure 4.7 The existing equal-area cellular layout of the facility with product-cell assignments

Besides, it is targeted to increase the effectiveness of material transfer actions and increase the machines' capacity usages by taking into account their operation capabilities as well as the total processing necessities (in time units) of each machining capability. Thus, 11-different REs are identified by the facility designers for the machines' processing capabilities. The number of available machine copies and the machining capability data associated with the REs are demonstrated in Table 4.12. Total area requirements including the machines' installation and operation areas, work-in-process inventory spaces, operator working areas and the areas needed for aisles for the material handling and transfer operations are also presented in Table 4.12. It should be noted here that total net areas are calculated by summing the aforementioned space requirements for all of the machine copies. For instance, just 10.5007 m² is needed for one copy of the MHP Machining Centre. Based on these space requirement calculations for each machine, total area requirement of this shop floor can be computed as 147.62312 m². According to the block plan of this facility, its side lengths in the x and y-axis directions are equal to 15 and 10 meters, respectively.

Table 4.12 The number of machine copies with their capabilities in terms of the REs (Baykasoğlu and Gindy, 2000)

Machines	Copies	Resource elements (REs)											Picture	Total area (m ²)
		1	2	3	4	5	6	7	8	9	10	11		
1-Drill Press (M1)	1	*												10.10202
2-MHP Machining Centre (M2-M3-M4)	3	*	*	*				*	*	*	*	*		31.5021
3-Colchester Lathe (M5-M6)	2	*	*		*			*						30.2966
4-MHP MT50 NC Lathe (M7-M8)	2	*	*		*			*			*			29.694
5-CNC Grinding Machine (M9-M10)	2					*	*							20.903
6-Jones & Shipman Cyc. Grinder (M11)	1						*							13.7625
7-Jones & Ship. Surf. Grinder (M12)	1				*									11.3629

Since the total available area of the shop floor (i.e., 150 m²) is larger than the total area requirements of all the machines, there will be a few empty spaces after completing the facility design. After that, monthly demand quantities and process

flows information (or manufacturing routes) of totally 20-different products are also specified by the process planners in terms of the REs as given in Table 4.13. By means of these demand quantities and process flow information, product flow quantities among distinct REs are calculated as in Table 4.14 by using Equation (4.1). Furthermore, REs' total processing necessities (in min.) are also calculated by Equation (4.2) based on this process plan, demand information as well as the REs' processing times for each kind of product (see in Table 4.14).

Table 4.13 Demands quantities, process plan and processing times of the products

Products	Demand quantities (units)	Process flow information (Processing times)
1	3000	RE1 (3) – RE2 (4) – RE4 (2)
2	1000	RE1 (5) – RE2 (1) – RE3 (3)
3	2500	RE5 (1) – RE6 (2) – RE7 (4)
4	1520	RE8 (2) – RE5 (1)
5	1480	RE7 (3) – RE4 (5) – RE5 (2)
6	3500	RE8 (1) – RE6 (1) – RE7 (1)
7	1000	RE8 (2) – RE9 (3) – RE10 (4)
8	2000	RE9 (1) – RE10 (3) – RE11 (5)
9	3000	RE5 (3) – RE1 (4) – RE2 (2)
10	2000	RE3 (2) – RE4 (3)
11	4500	RE5 (1) – RE6 (4) – RE9 (1)
12	1000	RE10 (3) – RE9 (2) – RE8 (3)
13	3000	RE5 (1) – RE8 (4) – RE10 (2)
14	2500	RE8 (2) – RE7 (1) – RE5 (4)
15	2500	RE1 (2) – RE2 (5)
16	1900	RE3 (1) – RE4 (2)
17	2400	RE6 (3) – RE7 (2) – RE8 (4)
18	1200	RE8 (2) – RE9 (1) – RE10 (5) – RE11 (3)
19	1300	RE5 (3) – RE2 (1)
20	3000	RE7 (5) – RE8 (3) – RE9 (4)

Table 4. 14 Product flows among distinct REs with their total processing necessities

From/To	RE-1	RE-2	RE-3	RE-4	RE-5	RE-6	RE-7	RE-8	RE-9	RE-10	RE-11
RE-1		9500									
RE-2			1000	3000							
RE-3				3900							
RE-4					1480						
RE-5	3000	1300				7000		3000			
RE-6							8400		4500		
RE-7				1480	2500			5400			
RE-8					1520	3500	2500		5200	3000	
RE-9								1000		4200	
RE-10									1000		3200
RE-11											
Total proc. req. (min.)	20200	32800	8900	23200	37380	33700	40240	49540	24700	25000	13600

According to the working policy of this manufacturing company, there is only one shift period and the average number of working days is equal to 20 days per month. Additionally, the maximum number of products that can be processed by these machines per hour are calculated based on the theoretical machine capacities. Therefore, the upper bounds on the machine capacity utilizations (in min.) are also specified as 40000 for Drill Pres and one copy of MHP Machining Centre. Similarly, the upper bounds on the machine capacity utilizations are calculated as 50000 minutes for all of the other machines.

In addition to these limitations on the maximum capacity utilizations, process planners of the company desired to define lower bounds on the machine capacity utilizations in order to decrease the total idle time of the machines. For that reason, a new constraint given in Equation (4.63) was also added to the previously formulated MINLP model to satisfy their expectation. The limitation on the minimal capacity usage (i.e., $Lcap_i$) of each machine was set to 20000 units. It should be noted here that if machine i utilizes any RE (or its statue will be active), then its capacity utilization must be greater than this lower bound. If it doesn't use any RE or (i.e., inactive machine statue), its capacity utilization will also be equal to zero.

$$Lcap_i \cdot \varphi_i \leq H_i \quad \forall i \in N \quad (4.63)$$

Finally, the aspect ratio is set to 2.0 for all the departments where these machines except two copies of Colchester Lathe (M5 and M6) are located. Due to the large size difference of this machine type, i.e., Colchester Lathe, their aspect ratio is determined as 3.0 by the facility designers. Additionally, they also specified the value of maximum allowable approximation error ratio as 1.6% for each department where these machines are installed. Thus, the minimum number of secant lines is computed as “6” for each department with respect to their aspect ratios and the pre-determined error ratio.

According to the polyhedral inner-approximation method, the minimum/maximum side lengths of each machine location are first calculated as given in Table 4.15 based on the side lengths of the shop floor, aspect ratios and total area requirements. Thereafter, their slopes and intercepts are computed as in Tables 4.16 and 4.17 after calculating these min/max points of the secant lines.

It is seen in Table 4.16 that the slopes of each secant line is equal to each other for every machine location (except the two copies of Colchester Lathe) since the same value of aspect ratio ($AR_i = 2.0$) is identified. However, the slopes of secant lines for these two copies of Colchester Lathe are completely different from the other machines since their aspect ratios are set to $AR_i = 3.0$. Besides, all the intercepts of these secant lines varied from one machine location to another as can be seen in Table 4.17.

Table 4.15 Minimum/maximum side lengths of departments where the machines are located

Machines	x_{min}	y_{min}	x_{max}/y_{max}
Drill Press (M1)	2.247	2.247	4.495
MHP Machining Centre (M2)	2.291	2.291	4.583
MHP Machining Centre (M3)	2.291	2.291	4.583
MHP Machining Centre (M4)	2.291	2.291	4.583
Colchester Lathe (M5)	2.247	2.247	6.741
Colchester Lathe (M6)	2.247	2.247	6.741
MHP MT50 NC Lathe (M7)	2.724	2.725	5.449
MHP MT50 NC Lathe (M8)	2.724	2.725	5.449
CNC Grinding Machine (M9)	2.286	2.286	4.572
CNC Grinding Machine (M10)	2.286	2.286	4.572
Jones & Shipman Cyc. Grinder (M11)	2.623	2.623	5.246
Jones & Ship. Surf. Grinder (M12)	2.383	2.386	4.767

By making use of the above data given in Tables 4.12 - 4.17, the proposed MINLP model in Equations (4.3) – (4.29) and (4.38) is first solved without any linearization process by the non-linear programming solver of Gurobi 9.1 in Python 3.8.8 modelling and development environment on an Intel Corei7 2 GHz IBM PC. The runtime limit is set to 10800-seconds (3-hours) for this original MINLP model. The best found objective value (or layout score) by the proposed MINLP is equal to 378.428,81 (Total flow * distances) under this runtime limit.

After that, only the non-linear constraints of the proposed MINLP model (i.e., Equations 4.7, 4.8, 4.15, 4.18 and 4.26) are linearized as explained in Sections 4.1.2 and 4.1.3. However, the objective function in Equation (4.3) remains still non-linear due to the product of an integer variable, i.e., f_{ij} with summation of the two continuous variables, i.e., $(d_{ij}^x + d_{ij}^y)$. As mentioned previously, since the linearization of this objective function causes significant increases in model dimension, it is certainly not applicable for solving such a realistic size problem instance.

Table 4.16 The slopes of the secant lines for the real-life case study

Secant lines	Slopes (α_{pi})											
	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12
1	-1.554	-1.554	-1.554	-1.554	-2.331	-2.331	-1.554	-1.554	-1.554	-1.554	-1.554	-1.554
2	-0.938	-0.938	-0.938	-0.938	-1.407	-1.407	-0.938	-0.938	-0.938	-0.938	-0.938	-0.938
3	-0.566	-0.566	-0.566	-0.566	-0.85	-0.85	-0.566	-0.566	-0.566	-0.566	-0.566	-0.566
4	-0.342	-0.342	-0.342	-0.342	-0.513	-0.513	-0.342	-0.342	-0.342	-0.342	-0.342	-0.342
5	-0.206	-0.206	-0.206	-0.206	-0.31	-0.31	-0.206	-0.206	-0.206	-0.206	-0.206	-0.206
6	-0.125	-0.125	-0.125	-0.125	-0.187	-0.187	-0.125	-0.125	-0.125	-0.125	-0.125	-0.125

Table 4.17 The intercepts of the secant lines for the real-life case study

Secant lines	Intercepts (β_{pi})											
	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12
1	7.987	8.144	8.144	8.144	11.79	11.979	9.683	9.683	8.124	8.124	9.323	8.471
2	6.206	6.328	6.328	6.328	9.308	9.308	7.524	7.524	6.313	6.313	7.244	6.582
3	4.822	4.917	4.917	4.917	7.232	7.232	5.846	5.846	4.905	4.905	5.629	5.114
4	3.747	3.82	3.82	3.82	5.619	5.619	4.542	4.542	3.811	3.811	4.373	3.974
5	2.911	2.968	2.968	2.968	4.366	4.366	3.529	3.529	2.961	2.961	3.398	3.088
6	2.262	2.306	2.306	2.306	3.393	3.393	2.742	2.742	2.301	2.301	2.64	2.399

For that reason, constraint linearization procedures are implemented only and the resulting MINLP model was resolved under 3-hours runtime limit. The best found objective value by the resulting MINLP with linear constraints is equal to 378.927,95. The generated facility layout design options via the proposed MINLP model with or without constraint linearization are also depicted in Figure 4.8. It is clear in Figure 4.8 that similar objective function values are provided despite the application of constraint linearization procedures.

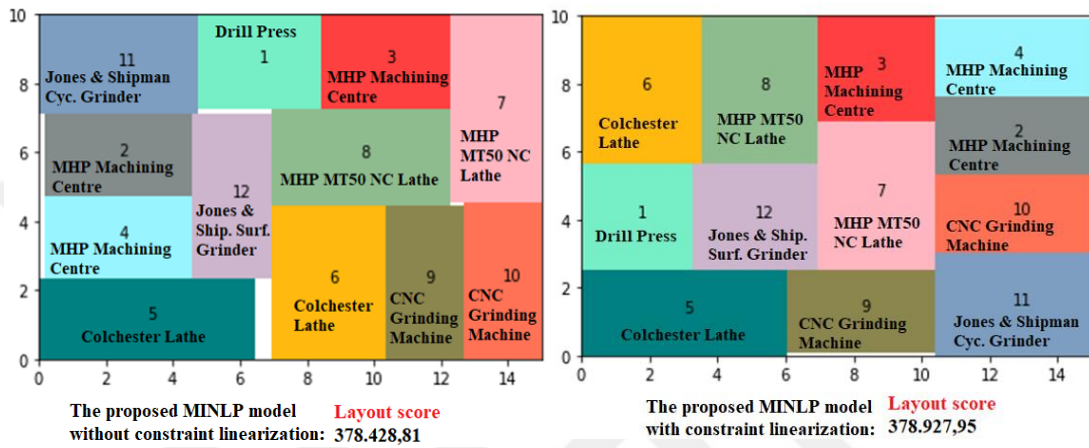


Figure 4.8 The best capability-based UA-FLD options by the proposed MINLP model with or without constraint linearization

This means that even if all the non-linear constraints are linearized properly, the performance of the proposed MINLP doesn't improve because of the existence of extremely non-linear objective. Consequently, linearization of the constraint functions is not enough for solving the proposed UA-CBFLD problem under reasonable computation times. Hence, application of such a heuristic decomposition-based iterative mathematical programming approach is inevitable for solving the large-sized UA-CBFLD problem instances.

Consequently, the heuristic decomposition-based iterative mathematical programming approach is also applied to solve the submodels-1 and 2 by using Gurobi MIP solver. The detailed optimization results of these sub-models are presented in Table 4.18 at each iteration. According to Table 4.18, capability-based product flow problem (sub-model-2) can be solved within very shorter computing times.

However, solution of the UA-FLD problem (sub-model-1) took relatively longer time because of its NP-hard and complex nature. It should be highlighted here that the runtime limit is set to 1-hour for each sub-model. Fortunately, superior objective value (or lower layout score), i.e., 283.976,05 is obtained by the heuristic decomposition-based iterative mathematical programming approach under relatively shorter computing time (i.e., totally 4985.09 seconds) when compared to the proposed MINLP model with/without constraint linearization (see Table 4.19).

Table 4.18 Optimization results of the proposed heuristic decomposition-based solution approach for the real-life case study

Iteration No.	Best solution of submodel-2	Optimality gap (%)	Best solution of submodel-1	Lower bound	Optimality gap (%)	CPU Time
1	75380	0%	-	-	-	2.03
2	-	-	315.856,06	304.016,37	3.74%	3600*
3	293.234,99	0%	-	-	-	2.89
4	-	-	283.978,07	283.976,05	0%	1377.32
5	283.978,07	0%	-	-	-	2.85
Best objective value (Layout Score)			283.978,07	Total CPU time		4985.09

Table 4.19 Summary of the comparative results for the existing cellular layout and the proposed layout design options

	MINLP without any linearization	MINLP with constraint linearization	Heuristic decomposition-based iterative approach	Existing cellular layout with equal machine areas	Existing cellular layout with unequal machine areas
Best found objective	378.428,81	378.927,95	283.978,07	367.400	382.652,14
Computation time (sec.)	10800	10800	4985.09	-	-

In order to compare the outcomes of the proposed UA-CBFLD approach with the current cellular layout of the facility, actual values of the centroids for the machine locations and their side lengths in the current manufacturing system were put into the proposed MINLP model as input data and then, the layout score of the existing cellular layout with equal machine areas is provided as 367.400 by running the proposed model. It should be noted here again that the existing cellular layout cannot consider the machines' processing capabilities (or REs) and also assumes that each machine has equal area requirement.

However, this unrealistic assumption related to the equal machine areas cannot provide an accurate comparison with the existing cellular layout of the manufacturing company. For this reason, the proposed MINLP model was also resolved for the existing cellular layout by considering the unequal machine areas and then, the layout score of the existing cellular layout with unequal machine areas is yielded as 382.652,14. The existing cellular layout with equal and unequal machine areas is also visualized in Figure 4.9. It is clear that the layout score will increase or deteriorate when the realistic unequal machine areas are taken into account.

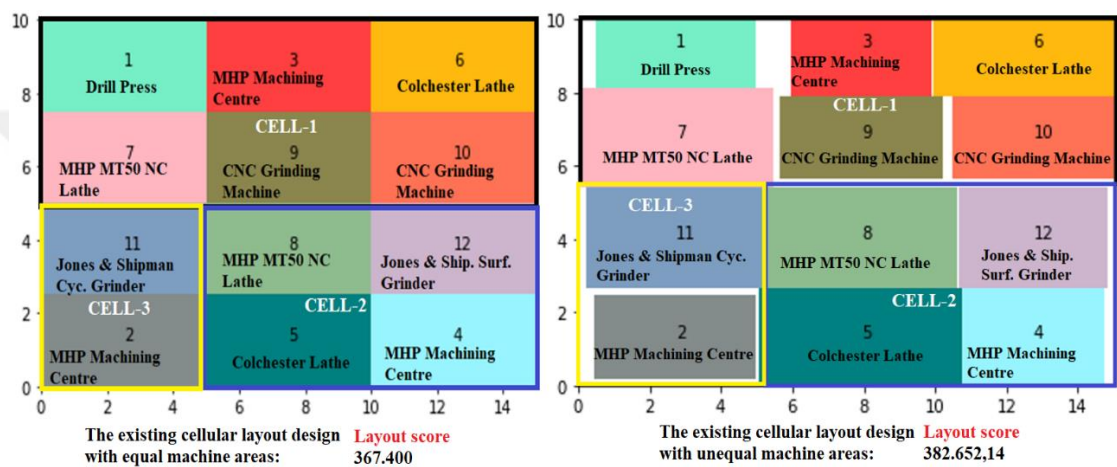


Figure 4.9 The existing cellular facility layout of the company with equal or unequal machine areas

As it is obviously seen in Table 4.19 and Figures 4.9 - 4.10, when compared to the existing cellular layout designs with equal/unequal machine areas, superior layout score, i.e., 283.978,07, is generated by using the proposed UA-CBFLD approach via the heuristic decomposition-based iterative mathematical programming approach. Fortunately, unequal machine areas and different processing capabilities of the machines can be easily handled by the proposed UA-CBFLD approach. Therefore, more effective and efficient material handling operations can be provided via the proposed approach. The best capability-based unequal-area facility layout design generated by the heuristic decomposition-based approach is also visualized in Figure 4.10. It is clear in Figures 4.9 and 4.10 that total empty spaces in the best capability-based unequal-area facility layout design is less than the existing cellular layout because of the usage of inner-approximation method for handling the non-linear area constraints.

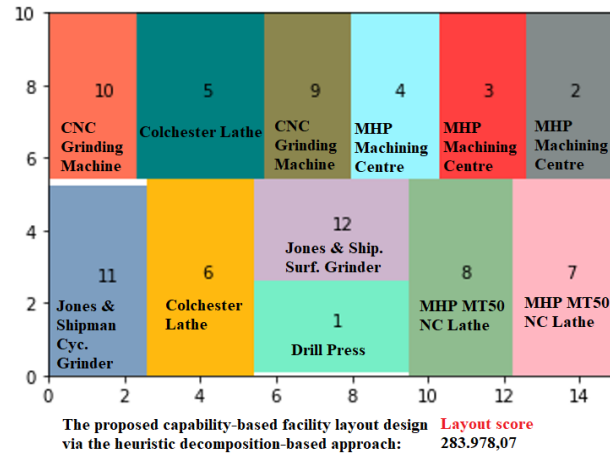


Figure 4.10 The proposed best capability-based UA-FLD via the heuristic decomposition-based approach

In other words, machines' actual areas in the proposed solution is too little bigger than or equal to the needed machine areas since this method guarantees a given maximal deviation error. Comparison of these actual and required machine areas with the error rates are also given in Table 4.20 for the proposed best capability-based UA-FLD through the heuristic-decomposition based solution approach. According to Table 4.20, error rate of each machine area is less than the pre-specified maximum allowable approximation error rate, i.e., 1.6%. The total average approximation error is equal to 0.98% which can also be reduced by using more secant lines in the inner-approximation method. However, it should be noted here that when the number of secant lines are increased to decrease this approximation error, the number of constraints (or model dimension) will also increase. The detailed optimization results of the best found capability-based unequal-area facility layout design are also presented in Tables 4.21 - 4.23.

The resulted product flows among distinct machines are first presented in Table 4.21. As mentioned previously, flows among distinct machines are the foremost outputs of the proposed approach in contradiction of the classical UA-FLDPs in which these flows are just stated as input data. According to Table 4.21, there is no product flow from or to the CNC Grinding Machine (i.e., M10) and Jones & Shipman Cyc. Grinder (i.e., M11). The reason is that these machines (i.e., M10 and M11) doesn't use any RE as can also be seen in Table 4.22.

Table 4.20 Comparison of the required and actual machine areas with error rates for the inner-approximation method

Machines	Required area (m ²)	Actual area	Error rate (%)
Drill Press (M1)	10.10202	10.24346	1.4%
MHP Machining Centre (M2)	10.5007	10.62560	1.18%
MHP Machining Centre (M3)	10.5007	10.66416	1.55%
MHP Machining Centre (M4)	10.5007	10.62560	1.18%
Colchester Lathe (M5)	15.1483	15.38735	1.57%
Colchester Lathe (M6)	15.1483	15.25621	0.71%
MHP MT50 NC Lathe (M7)	14.847	14.87019	0.156%
MHP MT50 NC Lathe (M8)	14.847	14.87019	0.156%
CNC Grinding Machine (M9)	10.4515	10.61455	1.56%
CNC Grinding Machine (M10)	10.4515	10.57378	1.17%
Jones & Shipman Cyc. Grinder (M11)	13.7625	13.90785	1.05%
Jones & Ship. Surf. Grinder (M12)	11.3629	11.43229	0.61%
Total areas	147.62312	149.0712485	0.98%

Therefore, they can be stated as inactive or unused machines and they have no capacity utilization (or their worst case capacity utilizations are equal to “0”). On the other hand, other machines except M10 and M11 have utilized at least one RE as given in Table 4.22. Therefore, there are some product flows from/to these active machines. The RE used by these machines with their capacity utilizations are also presented in Table 4.22.

Table 4.21 The resulting product flows (units) between different machines for the real-life case study

Machines	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12
M1										-	-	
M2							3900			-	-	
M3		3200	5200	1000						-	-	
M4			8200		2500				3500	-	-	1520
M5				5400		1480				-	-	2500
M6										-	-	1480
M7								9500		-	-	
M8		1000					3000			-	-	
M9			4500		8400					-	-	
M10	-	-	-	-	-	-	-	-	-	-	-	-
M11	-	-	-	-	-	-	-	-	-	-	-	-
M12	3000			3000				1300	7000	-	-	

Table 4.22 The REs used by different machines and their worst-case capacity utilizations

Machines	Utilized REs	Worst case capacity utilization (min.)
Drill Press (M1)	RE1	20200
MHP Machining Centre (M2)	RE3 – RE11	40000
MHP Machining Centre (M3)	RE9 – RE10	49700
MHP Machining Centre (M4)	RE8	50000
Colchester Lathe (M5)	RE7	40240
Colchester Lathe (M6)	RE4	50000
MHP MT50 NC Lathe (M7)	RE1 – RE4	50000
MHP MT50 NC Lathe (M8)	RE2	32800
CNC Grinding Machine (M9)	RE6	50000
CNC Grinding Machine (M10)	-	0
Jones & Shipman Cyc. Grinder (M11)	-	0
Jones & Ship. Surf. Grinder (M12)	RE5	50000

For instance, although the MHP Machining Centres (i.e., M2 and M3) have already eight different REs (see in Table 4.12), they utilized only two of them during the manufacturing process. Lastly, details of the product flows with the processing capability information (or REs) are also presented in Table 4.23.

Table 4.23 Details of the product flows among distinct machines for the real-life case study

From machine	From RE	To machine	To RE	Amounts of product flows	Total product flow
M2	RE-3	M7	RE-4	3900	3900
M3	RE-10	M2	RE-11	3200	3200
M3	RE-9	M3	RE-10	4200	
M3	RE-10	M3	RE-9	1000	5200
M3	RE-9	M4	RE-8	1000	1000
M4	RE-8	M3	RE-9	5200	
M4	RE-8	M3	RE-10	3000	8200
M4	RE-8	M5	RE-7	2500	2500
M4	RE-8	M9	RE-6	3500	3500
M4	RE-8	M12	RE-5	1520	1520
M5	RE-7	M4	RE-8	5400	5400
M5	RE-7	M6	RE-4	1480	1480
M5	RE-7	M12	RE-5	2500	2500
M6	RE-4	M12	RE-5	1480	1480
M7	RE-1	M8	RE-2	9500	9500
M8	RE-2	M2	RE-3	1000	1000
M8	RE-2	M7	RE-4	3000	3000
M9	RE-6	M3	RE-9	4500	4500
M9	RE-6	M5	RE-7	8400	8400
M12	RE-5	M1	RE-1	3000	3000
M12	RE-5	M4	RE-8	3000	3000
M12	RE-5	M8	RE-2	1300	1300
M12	RE5	M9	RE-6	7000	7000

Table 4.23 have shown that huge amount of product flows, i.e., 8400 units, from CNC Grinding Machine (i.e., M9) to Colchester Lathe (i.e., M5) since the CNC Grinding Machine doesn't include the RE-7. In other words, since the RE-7 is unavailable in the processing capability set of CNC Grinding Machine, totally 8400 units of products are transported from its available RE-6 to the Colchester Lathe. In addition to available RE-5 and RE-6, if the CNC Grinding Machine had also contained RE-7, there will be no need to any product flow from CNC Grinding Machine to Colchester Lathe. The product flows inside the machines' themselves are also given in Table 4.23. For instance, totally 5200 units of products (from RE-9 to RE-10: 4200 units; from RE-10 to RE-9: 1000 units) are transferred within the MHP Machining Centre (M3) itself as shown in Tables 4.21 - 4.23.

CHAPTER 5

CONCLUSIONS AND FUTURE RESEARCHES

5.1 Summary of the Thesis and Concluding Remarks

In the scope of this thesis, two advanced and new generation facility layout design approaches are addressed based on extensions of the previous researches of Baykasoğlu and Subulan (2020) and Baykasoğlu et al. (2022).

Firstly, a robust capability-based distributed layout design problem with fuzzy demand data and random machine availabilities is presented in Chapter 3. Indeed, a fuzzy-stochastic programming model of this problem is first developed to obtain robust layout alternatives under different uncertainty levels and scenarios. Then, a hybrid solution technique based on chance-constrained stochastic programming and an interactive fuzzy resolution method is proposed to transform the fuzzy-stochastic optimization model into its deterministic equivalent form. The proposed approach is first tested on a numerical example with extensive computational analysis. Actually, several computational analyses are conducted on this numerical example by considering different machine-capability overlap cases, probabilities for machine breakdowns and target probabilities for the chance constraint sets. Furthermore, comparative results are also presented for the deterministic, fuzzy, stochastic and fuzzy-stochastic optimization cases to gain managerial insights. Lastly, a real-world implementation is offered for a cellular manufacturing system of a manufacturing facility to demonstrate the rationality and applicability of the proposed approach under mixed fuzzy-stochastic environments. Computational experiments have displayed that the proposed approach can produce better outcomes in terms of the expected layout scores when compared to the current cellular layout of the facility. Moreover, many risky and/or risk-free robust layout design alternatives can be produced through the proposed approach according to the facility designers' risk attitude. For instance, risk-free robust layout design options in which “the feasibility degree of fuzzy-stochastic constraints is high, whereas the number of unsatisfied scenarios for the chance constraints is low” can be generated for the risk-averse facility designers. Furthermore, critical machines, which covers the vital processing capabilities and therefore,

necessitate additional preventive maintenance actions can also be determined easily by using the proposed approach. Consequently, manufacturing facilities may be aware of these critical machines and aim to increase their reliability rates or decrease their breakdown probabilities. Thus, more efficient and reliable robust layout design alternatives can be derived via the proposed approach under different scenarios and uncertainty levels.

In Chapter 4, a novel unequal-area capability-based facility layout design (UA-CBFLD) problem is presented for the first time in the literature. The MINLP model of this problem is first formulated by extending the recent INLP model in Baykasoğlu et al. (2022) for the capability-based machine layout (CB-ML) problem where the machines are allocated to pre-specified factory locations with equal areas. In the scope of this thesis, machines' operation capabilities which are described by the REs as well as their unequal area requirements are taken into consideration. In contrast to the classical unequal facility layout design approaches in which the product flows among the machines are generally defined as input data (or model parameter), product flows are identified based on the machining capabilities and therefore, they are also described as decision variables that makes the objective function of the proposed MINLP model highly non-linear. Furthermore, in order to cope with the nonlinear unequal area constraints, a polyhedral inner-approximation method which has been recently proposed by Xie et al. (2016) & Xiao et al. (2017) is implemented. Actually, this area linearization method is chosen to handle the highly non-linear area constraints since it guarantees that the actual areas is too little bigger than or equal to the needed areas within a pre-specified maximal deviation error. Due to the NP-hard nature and highly non-linear structure of the proposed UA-CBFLD problem, a heuristic decomposition-based iterative mathematical programming approach is applied to obtain superior solutions for the large-sized problem instances within realistic run times. In order to illustrate the proposed UA-CBFLD approach, a basic numerical example with different machine-capability overlap cases is first generated to perform extensive computational analysis. Thereafter, an application study in a real-life manufacturing system is also presented to display the advantages of the heuristic decomposition-based approach and compare its solutions with the outcomes of the developed MINLP model as well as the current cellular facility layout of the

manufacturing company. The computational results have shown that superior layout scores can be obtained by the heuristic decomposition-based approach under shorter computing times when compared to the proposed MINLP model with/without constraint linearization procedures. Besides, the proposed capability-based unequal-area layout design alternative has also significantly better objective value (or layout score) than the existing cellular layout design of the company.

5.2 Future Research Directions

For the robust FS-BCBDL problem which is addressed in Chapter 3, a metaheuristic solution approach which combines a metaheuristic algorithm with the proposed fuzzy-stochastic programming model can be developed in the future to solve the larger sized FS-BCBDL problem instances under shorter computing times. Moreover, capability-based distributed layout design approach is only studied with equal machine areas so far. Hence, development of an optimization model for an unequal-area capability-based distributed layout design problem may be planned as a future work. It should be noted here that the classical UA-FLD problem is extended by considering unequal machine areas in the second part of this thesis. The allocation of the machining capabilities over the factory floor is not intended. In other words, distributed layout concept with unequal machine areas has not considered yet.

For the UA-CBFLD problem which is discussed in Chapter 4, uncertainty related to the machine reliability/availability can be handled by using stochastic programming methods to produce more robust layout design alternatives. Additionally, fuzzy sets and systems can be used to deal with the uncertainties in product flows among different REs. Finally, development of a dynamic UA-CBFLD problem in which product flows among several REs may vary in each time period can also be handled in the future works.

REFERENCES

- Ahmadi-Javid, A., & Ardestani-Jaafari, A. (2021). The unequal area facility layout problem with shortest single-loop AGV path: how material handling method matters. *International Journal of Production Research*, 59(8), 2352 – 2374.
- Aiello, G., & Enea M. (2001). Fuzzy approach to the robust facility layout in uncertain production environments. *International Journal of Production Research* 39(18), 4089-4101.
- Baykasoğlu, A., & Gindy, NN. (2000). MOCACEF 1.0: Multiple objective capability based approach to form part-machine groups for cellular manufacturing applications. *International Journal of Production Research* 38(5), 1133-1161.
- Baykasoğlu, A., & Göçken, M. (2010). Capability-based distributed layout and its simulation based analyses. *Journal of Intelligent Manufacturing* 21(4), 471-485.
- Baykasoğlu, A., & Subulan, K. (2019). A fuzzy-stochastic optimization model for the intermodal fleet management problem of an international transportation company. *Transportation Planning and Technology* 42(8), 777-824.
- Baykasoğlu, A., & Subulan, K. (2020). Capability-based distributed layout formation with or without demand and process flow information. *Applied Soft Computing* 94, 1-16.
- Baykasoğlu, A. (2003). Capability-based distributed layout approach for virtual manufacturing cells. *International Journal of Production Research* 41(11), 2597-2618.
- Baykasoğlu, A. (2003). Capability-based distributed layout approach for virtual manufacturing cells. *International Journal of Production Research*, 41(11), 2597-2618.
- Baykasoğlu, A., & Gindy, N.N.Z. (2000). MOCACEF 1.0: Multiple objective capability based approach to form part-machine groups for cellular manufacturing applications. *International Journal of Production Research*, 38(5), 1133-1161.

- Baykasoğlu, A., & Göçken, M. (2010). Capability-based distributed layout and its simulation based analyses. *Journal of Intelligent Manufacturing*, 21(4), 471-485.
- Baykasoğlu, A., & Subulan, K. (2020). Capability-based distributed layout formation with or without demand and process flow information. *Applied Soft Computing*, 94, Article 106469.
- Baykasoğlu, A., Gindy, N.N.Z., & Cobb, R.C. (2001). Capability based formulation and solution of multiple objective cell formation problems using simulated annealing. *Integrated Manufacturing Systems*, 12(4), 258-274.
- Baykasoğlu, A., Subulan, K., & Hamzadayı, A., (2022). Capability-based machine layout with a matheuristic-based approach. *Expert Systems with Applications*, 198, Article 116900.
- Benjaafar, S., Heragu, SS., & Irani, SA. (2002). Next generation factory layouts: Research challenges and recent progress. *Interfaces* 32(6), 58–76.
- Benjaafar, S., & Sheikhzadeh, M. (2000). Design of flexible plant layouts. *IIE Transactions* 32, 309-322.
- Bisschop, J., & Roelofs, M. (2006). AIMMS – The User’s Guide. The Netherlands: Paragon Decision Technology.
- Castillo, I., & Westerlund, T. (2005). An ε -accurate model for optimal unequal area facility layout. *Computers & Operations Research*, 32, 429–447.
- Celik, M., Efeoglu, B., & Sural, H. (12-16 June 2016). A performance measurement framework and solution approach for the integrated facility layout problem with uncertain demand. *14th International Material Handling Research Colloquium Proceedings*. Karlsruhe, Baden-Württemberg, Germany.
- Charnes, A., & Cooper, WW. (1959). Chance-constrained programming. *Management Science* 6(1), 73-79.

- Coelho, L.C. (2013). Linearization of the product of two variables. *Working papers of Canada Research Chair in Integrated Logistics*. <https://www.leandro-coelho.com/linearization-product-variables/> [accessed 17 April 2022].
- Drira, A., Pierreval, H., & Hajri-Gabouj, S. (2013). Design of a robust layout with information uncertainty increasing over time: A fuzzy evolutionary approach. *Engineering Applications of Artificial Intelligence* 26(3), 1052-1060.
- Ertay, T., Ruan, D., & Tuzkaya, UR. (2006). Integrating data envelopment analysis and analytic hierarchy for the facility layout design in manufacturing systems. *Information Sciences* 176(3), 237-262.
- Fazlelahi, FZ., Pournader, M., Gharakhani, M., & Sadjadi, SJ. (2016). A robust approach to design a single facility layout plan in dynamic manufacturing environments using a permutation-based genetic algorithm. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* 230(12), 2264-2274.
- Friedrich, C., Klausnitzer, A., & Lasch, R. (2018). Integrated slicing tree approach for solving the facility layout problem with input and output locations based on contour distance. *European Journal of Operational Research*, 270(3), 837 – 851.
- Garcia-Hernandez, L., Garcia-Hernandez, J.A., Salas-Morera, L., Carmona-Muñoz, C., Alghamdi, N.S., Oliveira, J.V., & Salcedo-Sanz, S. (2020). Addressing unequal area facility layout problems with the coral reef optimization algorithm with substrate layers. *Engineering Applications of Artificial Intelligence*, 93, Article 103697.
- Garcia-Hernandez, L., Salas-Morera, L., Garcia-Hernandez, J.A., Salcedo-Sanz, S., & Oliveira, J.V. (2019). Applying the coral reefs optimization algorithm for solving unequal area facility layout problems. *Expert Systems with Applications*, 138, Article 112819.

- Gindy, N.N.Z., & Ratchev, T.M. (1997). Cellular decomposition of manufacturing facilities using resource elements. *Integrated Manufacturing Systems*, 8(4), 215-222.
- Gindy, N.N.Z., Ratchev, T.M., & Case, K. (1996). Component grouping for cell formation using resource elements. *International Journal of Production Research*, 34(3), 727–752.
- Guan, C., Zhang, Z., Liu, S., & Gong, J. (2019). Multi-objective particle swarm optimization for multi-workshop facility layout problem. *Journal of Manufacturing Systems*, 53, 32 – 48.
- Gupta, N., Ali, I., & Bari, A. (2003). A compromise solution for multi-objective chance constraint capacitated transportation problem. *ProbStat Forum* 6, 60-67.
- Halawa, F., Madathil, S.C., & Khasawneh, M.T. (2021). Multi-objective unequal area pod-structured healthcare facility layout problem with daylight requirements. *Computers & Industrial Engineering*, 162, Article 107722.
- Hamed, M., Esmaeilian, G., Ismail, N., & Ariffin, M.K.A. (2012a). Capability-based virtual cellular manufacturing systems formation in dual-resource constrained settings using tabu search. *Computers & Industrial Engineering* 62(4), 953-971.
- Hamed, M., Esmaeilian, G., & Ismail, N.B. (2012b). Developing capability-based virtual cellular manufacturing systems and comparison with capability-based classical cellular manufacturing systems. *Applied Mechanics and Materials* 110, 3938-3946.
- Hamed, M., & Esmaeilian, G. (3-5 March 2015). Functional and distributed layouts and their effectiveness on capability-based virtual cellular manufacturing systems performance. *2015 International Conference on Industrial Engineering and Operations Management*. IEEE, pp.1-8. Dubai, United Arab Emirates.
- Hamed, M., Esmaeilian, G.R., Ismail, N., & Ariffin, M.K.A. (2012). Capability-based virtual cellular manufacturing systems formation in dual-resource constrained settings using Tabu search. *Computers & Industrial Engineering*, 62(4), 953–971.

- Hamed, M., Ismail, N., Esmaeilian, G.R., & Ariffin, M.K.A. (2012). Developing a method to generate semi-distributed layouts by genetic algorithm. *International Journal of Production Research*, 50(4), 953–975.
- Hunagund, I.B., Pillai, V.M., & Kempaiah, U.N. (2020). Design of robust layout for unequal area dynamic facility layout problems with flexible bays structure. *Journal of Facilities Management* 18(4), 361-392.
- Ingole, S., & Singh, D. (2021). Fixed and flexible shape facility layout problems using biogeography-based optimisation algorithm. *International Journal of Industrial and Systems Engineering*, 37(1), 84 – 118.
- Izadinia, N., Eshghi, K., & Salmani, M.H. (2014). A robust model for multi-floor layout problem. *Computers & Industrial Engineering* 78, 127-134.
- Jimenez, M., Arenas, M., Bilbao, A., & Rodriguez, M.V. (2007). Linear programming with fuzzy parameters: an interactive method resolution. *European Journal of Operational Research* 177, 1599–609.
- Jimenez, M. (1996). Ranking fuzzy numbers through the comparison of its expected intervals, *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 4(4), 379–388.
- Jingfa, L., Dawen, W., Kun, H., & Yu, X. (2017). Combining Wang–Landau sampling algorithm and heuristics for solving the unequal-area dynamic facility layout problem. *European Journal of Operational Research*, 262(3), 1052 – 1063.
- Jithavech, I., & Krishnan, K.K. (2010). A simulation-based approach for risk assessment of facility layout designs under stochastic product demands. *The International Journal of Advanced Manufacturing Technology* 49, 27-40.
- Khajemahalle, L., Emami, S., & Keshteli, R.N. (2021). A hybrid nested partitions and simulated annealing algorithm for dynamic facility layout problem: a robust optimization approach. *INFOR: Information Systems and Operational Research* 59(1), 74-101.

- Kim, M., & Chae, J. (2021). A monarch butterfly optimization for an unequal area facility layout problem. *Soft Computing*, 25, 14933 – 14953.
- Kochhar, J.S., Foster, B.T., & Heragu, S.S. (1998). HOPE: A genetic algorithm for the unequal area facility layout problem. *Computers & Operations Research*, 25(7-8), 583-594.
- Konak, A., Kulturel-Konak, S., Norman, B.A., & Smith, A.E. (2006). A new mixed integer programming formulation for facility layout design using flexible bays. *Operations Research Letters*, 34, 660–672.
- Kulturel-Konak S. (2007). Approaches to uncertainties in facility layout problems: perspectives at the beginning of the 21st century. *Journal of Intelligent Manufacturing* 18(2), 273-284.
- Kulturel-Konak, S. (2012). A linear programming embedded probabilistic tabu search for the unequal-area facility layout problem with flexible bays. *European Journal of Operational Research*, 223(3), 614-625.
- Kulturel-Konak, S., & Konak, A. (2013). Linear programming based genetic algorithm for the unequal area facility layout problem. *International Journal of Production Research*, 51(14), 4302 – 4324.
- Lacksonen, T.A. (1997). Preprocessing for static and dynamic facility layout problems. *International Journal of Production Research*, 35(4), 1095–1106.
- LINDO Systems Inc. LINGO: The Modelling Language and Optimizer <https://www.lindo.com/downloads/PDF/LINGO.pdf> (Accessed 17 December 2021).
- Liu, J., Liu, J., Yan, X., & Peng, B. (2020). A heuristic algorithm combining Pareto optimization and niche technology for multi-objective unequal area facility layout problem. *Engineering Applications of Artificial Intelligence*, 89, Article 103453.

- Liu, J., Liu, S., Liu, Z., & Li, B. (2020). Configuration space evolutionary algorithm for multi-objective unequal-area facility layout problems with flexible bays. *Applied Soft Computing*, 89, Article 106052.
- Mangasarian, O.L. (2007). Absolute value equation solution via concave minimization. *Optimization Letters*, 1, 3–8.
- Mangasarian, O.L. (2013). Absolute value equation solution via dual complementarity. *Optimization Letters*, 7, 625–630.
- Meller, R.D., & Gau, K.Y. (1996). Facility layout objective functions and robust layouts. *International Journal of Production Research* 34(10), 2727-2742.
- Meller, R.D., Narayanan, V., & Vance, P.H. (1999). Optimal facility layout design. *Operations Research Letters*, 23, 117–127.
- Montreuil, B. (1990). A modelling framework for integrating layout design and flow network design. In: *Proceedings of Materials Handling Research Colloq.* Hebron, KY, pp. 43–58.
- Morinaga, E., Iwasaki, K., Wakamatsu, H., & Arai, E. (1-5 September 2019). Reduction of computational load in robust facility layout planning considering temporal production efficiency. *IFIP International Conference on Advances in Production Management Systems*. Springer, Cham, pp. 189-195. Austin, TX, USA.
- Morinaga, E., Iwasaki, K., Wakamatsu, H., & Arai, E. (3-7 September 2017). A robust facility layout planning method considering temporal efficiency. *IFIP International Conference on Advances in Production Management Systems*. Springer, Cham, pp. 168-175. Hamburg, Germany.
- Moslemipour, G., Lee, T.S., & Loong, Y.T. (2018). Solving stochastic dynamic facility layout problems using proposed hybrid ac-cs-sa meta-heuristic algorithm. *International Journal of Industrial and Systems Engineering* 28(1), 1-31.

- Moslemipour, G., Lee, T.S., & Rilling, D. (2012). A review of intelligent approaches for designing dynamic and robust layouts in flexible manufacturing systems. *The International Journal of Advanced Manufacturing Technology* 60, 11-27.
- Moslemipour, G., Lee, T.S., & Loong, Y.T. (2017). Performance analysis of intelligent robust facility layout design. *Chinese Journal of Mechanical Engineering* 30(2), 407-418.
- Neghabi, H., Eshghi, K., & Salmani, M.H. (2014). A new model for robust facility layout problem. *Information Sciences* 278, 498-509.
- Nematian, J. (2014). A robust single row facility layout problem with fuzzy random variables. *The International Journal of Advanced Manufacturing Technology* 72, 255-267.
- Oral, M., & Kettani, O. (1992). A linearization procedure for quadratic and cubic mixed-integer problems. *Operations Research*, 40, 109-116.
- Paes, F.G., Pessoa, A.A., & Vidal, T. (2017). A hybrid genetic algorithm with decomposition phases for the unequal area facility layout problem. *European Journal of Operational Research*, 256(3), 742 – 756.
- Peng, Y., Zeng, T., Fan, L., Han, Y., & Xia, B. (2018). An improved genetic algorithm based robust approach for stochastic dynamic facility layout problem. *Discrete Dynamics in Nature and Society* 1529058.
- Pillai, V.M., Hunagund, L.B., & Krishnan, K.K. (2011). Design of robust layout for dynamic plant layout problems. *Computers & Industrial Engineering* 61(3), 813-823.
- Pourvaziri, H., Salimpour, S., Akhavan Niaki, S.T., & Azab, A. (2021). Robust facility layout design for flexible manufacturing: a doe-based heuristic. *International Journal of Production Research* DOI: 10.1080/00207543.2021.1967500

- Pourvaziri, H., Salimpour, S., Niaki, S.T.A., & Azab, A. (2021). Robust facility layout design for flexible manufacturing: a doe-based heuristic. *International Journal of Production Research*, <https://doi.org/10.1080/00207543.2021.1967500>
- Rosenblatt, M.J., & Lee, H.L. (1987). A robustness approach to facilities design. *International Journal of Production Research* 25(4), 479–486.
- Şahinkoç, M., & Bilge, Ü. (2018). Facility layout problem with qap formulation under scenario-based uncertainty. *INFOR: Information Systems and Operational Research* 56(4), 406-427.
- Salmani, M.H., Eshghi, K., & Neghabi, H. (2015). A bi-objective MIP model for facility layout problem in uncertain environment. *The International Journal of Advanced Manufacturing Technology* 81(9), 1563-1575.
- Shafigh, F., Defersha, F.M., & Moussa, S.E. (2015). A mathematical model for the design of distributed layout by considering production planning and system reconfiguration over multiple time periods. *Journal of Industrial Engineering International* 11(3), 283-295.
- Shafigh, F., Defersha, F.M., & Moussa, S.E. (2017). A linear programming embedded simulated annealing in the design of distributed layout with production planning and systems reconfiguration. *The International Journal of Advanced Manufacturing Technology* 88, 1119-1140.
- Shafigh, F., Defersha, F.M., & Moussa, S.E. (2015). A mathematical model for the design of distributed layout by considering production planning and system reconfiguration over multiple time periods. *Journal of Industrial Engineering International*, 11(3), 283–295.
- Shafigh, F., Defersha, F.M., & Moussa, S.E. (2017). A linear programming embedded simulated annealing in the design of distributed layout with production planning and systems reconfiguration. *International Journal of Advanced Manufacturing Technology*, 88(1–4), 1119–1140.

- Sherali, H.D., Fraticelli, B.M. P., & Meller, R.D. (2003). Enhanced model formulations for optimal facility layout. *Operations Research*, 51(4), 509 – 679.
- Subulan K. (2020). An interval-stochastic programming based approach for a fully uncertain multi-objective and multi-mode resource investment project scheduling problem with an application to ERP project implementation. *Expert Systems with Applications* 149, 113189.
- Targhi, N.S., Sabzehparvar, M., & Ebrahimnezhad, S. (2019). A mathematical model for robust facility layout problem in 3D space with possibility of floors. *2019 15th Iran International Industrial Engineering Conference*. IEEE, pp. 158-164. Yazd, Iran.
- Tate, D.M., & Smith, A.E. (1995). Unequal-area facility layout by genetic search. *IIE Transactions*, 27(4), 465 – 472.
- Vitayasak, S., Pongcharoen, P., & Hicks, C. (2019). Robust machine layout design under dynamic environment: dynamic customer demand and machine maintenance. *Expert Systems with Applications* 3, 100015.
- Vitayasak, S., & Pongcharoen, P. (2015b). Re-layout and robust machine layout design under stochastic demand. *Applied Mechanics and Materials* 789, 1252-1257.
- Vitayasak, S., & Pongcharoen, P. (25-28 June 2015a). Genetic algorithm based robust layout design by considering various demand variations. *International Conference in Swarm Intelligence*. Springer, Cham, pp. 257-265. Beijing, China.
- Vitayasak, S., & Pongcharoen, P. (7-9 December 2016). Application of genetic algorithm for quantifying the affect of breakdown maintenance on machine layout. *International Workshop on Multi-disciplinary Trends in Artificial Intelligence*. Springer, Cham, pp. 208-218. Chiang Mai, Thailand.
- Xiao, X., Hu, Y., Wang, W., & Ren, W. (2019). A robust optimization approach for unequal-area dynamic facility layout with demand uncertainty. *Procedia CIRP* 81, 594-599.

- Xiao, Y., Xie, Y., Kulturel-Konak, S., & Konak, A. (2017). A problem evolution algorithm with linear programming for the dynamic facility layout problem—A general layout formulation. *Computers & Operations Research*, 88, 187 – 207.
- Xie, Y., Zhou, S., Xiao, Y., & Chang, W. (2016). A New Area Linearization Method for Unequal Area Facility Layout Problem. *IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, DOI: 10.1109/IEEM.2016.7798086
- Zha, S., Guo, Y., Huang, S., Wang, F., & Huang, X. (2017). Robust facility layout design under uncertain product demands. *Procedia CIRP* 63, 354-359.
- Zhao, Y., & Wallace, S.W. (2015). A heuristic for the single-product capacitated facility layout problem with random demand. *EURO Journal on Transportation and Logistics* 4, 379-398.

APPENDICES

APPENDIX 1: The Lingo Code for the Robust FS-BCBDL Problem with Discrete Random Variables (Illustrative Example)

```
Sets:
Machine/1..6/:ava;
Location/1..6/;
Capability/1..6/;
Set1(Location,Location):d;
Set2(Machine,Capability):a;
Set3(Machine,Location):y;
Set4(Location,Capability);
Set5(Location,Capability,Location):z,ff;
Set6(Capability,Capability):f1,f2,f3;
Endsets
Data:
a=1,0,0,0,0,0,
  0,1,0,0,0,0,
  0,0,1,0,0,0,
  0,0,0,1,0,0,
  0,0,0,0,1,0,
  0,0,0,0,0,1;
d=0,1,2,1,2,3,
  1,0,1,2,1,2,
  2,1,0,3,2,1,
  1,2,3,0,1,2,
  2,1,2,1,0,1,
  3,2,1,2,1,0;
nsamp=5;
prb1=0.8;
prb2=0.9;
f1=0,0.01,0.01,0.019,0.019,0.128,
  0.019,0,0.019,0.032,0.01,0.05,
  0.01,0.05,0,0.01,0.07,0.01,
  0.05,0.019,0.01,0,0.032,0.01,
  0.01,0.019,0.032,0.01,0,0.07,
  0.01,0.07,0.05,0.05,0.01,0;
f2=0,0.016,0.011,0.021,0.021,0.131,
  0.021,0,0.021,0.032,0.016,0.053,
  0.01,0.053,0,0.011,0.075,0.016,
  0.053,0.021,0.016,0,0.032,0.011,
  0.011,0.021,0.032,0.016,0,0.075,
  0.011,0.075,0.053,0.053,0.011,0;
f3=0,0.016,0.012,0.024,0.024,0.145,
  0.024,0,0.024,0.032,0.016,0.064,
  0.016,0.064,0,0.012,0.08,0.016,
  0.064,0.024,0.016,0,0.032,0.012,
  0.016,0.024,0.032,0.016,0,0.08,
  0.012,0.08,0.064,0.064,0.016,0;
alpha=1;
Enddata
! (1);Min=@sum(Location(i):@sum(Location(j)|j#ne#i:@sum(Capability(r)
:ff(i,r,j)*d(i,j))));
```

```

!Subject to;
!(2);@for(Machine(k):@sum(Location(i):y(k,i))=1);
!(3);@for(Location(i):@sum(Machine(k):y(k,i))=1);
!(4);@for(Location(i):@for(Capability(r):@sum(Location(j):z(i,r,j))<
=11*(1-@sum(Location(j):z(j,r,i)))));
!Stochastic declarations;
@for(Machine(k):@SPSTGRNDV(1,ava(k)));
@sptableshape('avai', 2, 1);
@sptableoutc('avai', 0.9, 1);
@sptableoutc('avai', 0.1, 0);
@sptableinst('avai', 'ava1');
@sptableinst('avai', 'ava2');
@sptableinst('avai', 'ava3');
@sptableinst('avai', 'ava4');
@sptableinst('avai', 'ava5');
@sptableinst('avai', 'ava6');
@sptablerndv('ava1', ava(1));
@sptablerndv('ava2', ava(2));
@sptablerndv('ava3', ava(3));
@sptablerndv('ava4', ava(4));
@sptablerndv('ava5', ava(5));
@sptablerndv('ava6', ava(6));
@spsampsize(1,nsamp);
!(5);@for(Set4(i,r):
[chance1] @sum(Location(j)|j#ne#i:z(j,r,i))>=1-
(@sum(Machine(k):a(k,r)*y(k,i)*ava(k)));
);
@spchance('CCP1','>=',prb1);
@for(Set4(i,r):
@spchance('CCP1',chance1)
);
!(6);@for(Location(i):@for(Capability(r):
[chance2]
@sum(machine(k):@sum(Capability(t)|t#ne#r:alpha*((f2(t,r)+f3(t,r))/2
)*a(k,t)*y(k,i)*ava(k)+(1-
alpha)*((f1(t,r)+f2(t,r))/2)*a(k,t)*y(k,i)*ava(k)))-
@sum(Location(j)|j#ne#i:ff(i,r,j))<=999*(1-
@sum(Location(j)|j#ne#i:z(j,r,i)));
));
@spchance('CCP2','>=',prb2);
@for(Location(i):@for(Capability(r):
@spchance('CCP2',chance2)
));
!(7);@for(Location(i):@for(Machine(k):@bin(y(k,i))));
!(8);@for(Location(i):@for(Capability(r):@for(Location(j):@bin(z(i,r
,j)))));

!(9);@for(Location(i):@for(Capability(r):@for(Location(j)|j#ne#i:ff(
i,r,j)<=999*z(j,r,i))));

```

APPENDIX 2: The Lingo Code for Robust FS-BCBDL Problem with Binomial Distribution (Illustrative Example)

```

Sets:
Machine/1..6/:ava,prb;
Location/1..6/;
Capability/1..6/;
Set1(Location,Location):d;
Set2(Machine,Capability):a;
Set3(Machine,Location):y;
Set4(Location,Capability);
Set5(Location,Capability,Location):z,ff;
Set6(Capability,Capability):f1,f2,f3;
Endsets
Data:
a=1,0,0,0,0,0,
  0,1,0,0,0,0,
  0,0,1,0,0,0,
  0,0,0,1,0,0,
  0,0,0,0,1,0,
  0,0,0,0,0,1;
d=0,1,2,1,2,3,
  1,0,1,2,1,2,
  2,1,0,3,2,1,
  1,2,3,0,1,2,
  2,1,2,1,0,1,
  3,2,1,2,1,0;
nsamp=5;
prb1=0.8;
prb2=0.9;
f1=0,0.01,0.01,0.019,0.019,0.128,
  0.019,0,0.019,0.032,0.01,0.05,
  0.01,0.05,0,0.01,0.07,0.01,
  0.05,0.019,0.01,0,0.032,0.01,
  0.01,0.019,0.032,0.01,0,0.07,
  0.01,0.07,0.05,0.05,0.01,0;
f2=0,0.016,0.011,0.021,0.021,0.131,
  0.021,0,0.021,0.032,0.016,0.053,
  0.01,0.053,0,0.011,0.075,0.016,
  0.053,0.021,0.016,0,0.032,0.011,
  0.011,0.021,0.032,0.016,0,0.075,
  0.011,0.075,0.053,0.053,0.011,0;
f3=0,0.016,0.012,0.024,0.024,0.145,
  0.024,0,0.024,0.032,0.016,0.064,
  0.016,0.064,0,0.012,0.08,0.016,
  0.064,0.024,0.016,0,0.032,0.012,
  0.016,0.024,0.032,0.016,0,0.08,
  0.012,0.08,0.064,0.064,0.016,0;
prb=0.9,0.9,0.9,0.9,0.9,0.9;
alpha=1;
Enddata
!(1);Min=@sum(Location(i):@sum(Location(j)|j#ne#i:@sum(Capability(r)
:ff(i,r,j)*d(i,j)))));

!Subject to;
!(2);@for(Machine(k):@sum(Location(i):y(k,i))=1);
!(3);@for(Location(i):@sum(Machine(k):y(k,i))=1);

```

```

!(4);@for(Location(i):@for(Capability(r):@sum(Location(j):z(i,r,j))<
=11*(1-@sum(Location(j):z(j,r,i)))));
!Stochastic declarations;
@for(Machine(k):@SPSTGRNDV(1,ava(k)));
@for(Machine(k):@spdistbino(1,prb(k),ava(k)));
@spsampsize(1,nsamp);
!(5);@for(Set4(i,r):
[chance1] @sum(Location(j)|j#ne#i:z(j,r,i))>=1-
(@sum(Machine(k):a(k,r)*y(k,i)*ava(k))));
);
@spchance('CCP1','>=',prb1);
@for(Set4(i,r):
@spchance('CCP1',chance1)
);
!(6);@for(Location(i):@for(Capability(r):
[chance2]
@sum(machine(k):@sum(Capability(t)|t#ne#r:alpha*((f2(t,r)+f3(t,r))/2
)*a(k,t)*y(k,i)*ava(k)+(1-
alpha)*((f1(t,r)+f2(t,r))/2)*a(k,t)*y(k,i)*ava(k)))-
@sum(Location(j)|j#ne#i:ff(i,r,j))<=999*(1-
@sum(Location(j)|j#ne#i:z(j,r,i))));
));
@spchance('CCP2','>=',prb2);
@for(Location(i):@for(Capability(r):
@spchance('CCP2',chance2)
));
!(7);@for(Location(i):@for(Machine(k):@bin(y(k,i))));
!(8);@for(Location(i):@for(Capability(r):@for(Location(j):@bin(z(i,r
,j)))));

!(9);@for(Location(i):@for(Capability(r):@for(Location(j)|j#ne#i:ff(
i,r,j)<=999*z(j,r,i))));

```

APPENDIX 3: The Lingo Code for Robust FS-BCBDL Problem with Binomial Distribution (Real-Life Case Study)

```
Sets:
Machine/1..12/:ava,prb;
Location/1..12/;
Capability/1..11/;
Set1(Location,Location):d;
Set2(Machine,Capability):a;
Set3(Machine,Location):y;
Set4(Location,Capability);
Set5(Location,Capability,Location):z,ff;
Set6(Capability,Capability):f1,f2,f3;
Endsets
Data:
a=1,0,0,0,0,0,0,0,0,0,0,0,
1,1,1,0,0,0,1,1,1,1,1,
1,1,1,0,0,0,1,1,1,1,1,
1,1,1,0,0,0,1,1,1,1,1,
1,1,0,1,0,0,1,0,0,0,0,
1,1,0,1,0,0,1,0,0,0,0,
1,1,0,1,0,0,1,0,0,1,0,
1,1,0,1,0,0,1,0,0,1,0,
0,0,0,0,1,1,0,0,0,0,0,
0,0,0,0,1,1,0,0,0,0,0,
0,0,0,0,1,0,0,0,0,0,0,
0,0,0,0,1,0,0,0,0,0,0;
d=0,1,2,1,2,3,2,3,4,3,4,5,
1,0,1,2,1,2,3,2,3,4,3,4,
2,1,0,3,2,1,4,3,2,5,4,3,
1,2,3,0,1,2,1,2,3,2,3,4,
2,1,2,1,0,1,2,1,2,3,2,3,
3,2,1,2,1,0,3,2,1,4,3,2,
2,3,4,1,2,3,0,1,2,1,2,3,
3,2,3,2,1,2,1,0,1,2,1,2,
4,3,2,3,2,1,2,1,0,3,2,1,
3,4,5,2,3,4,1,2,3,0,1,2,
4,3,4,3,2,3,2,1,2,1,0,1,
5,4,3,4,3,2,3,2,1,2,1,0;
nsamp=5;
prb=0.95,0.9,0.9,0.9,0.85,0.85,0.85,0.85,0.9,0.9,0.95,0.95;
prb1=0.8;
prb2=0.8;
f1=0,0.117,0,0,0,0,0,0,0,0,0,
0,0,0.011,0.036,0,0,0,0,0,0,0,
0,0,0,0.044,0,0,0,0,0,0,0,
0,0,0,0,0.017,0,0,0,0,0,0,
0.036,0.011,0,0,0,0.082,0,0.036,0,0,0,
0,0,0,0,0,0.096,0,0.054,0,0,0,
0,0,0,0.016,0.031,0,0,0.064,0,0,0,
0,0,0,0,0.017,0.042,0.032,0,0.066,0.036,0,
0,0,0,0,0,0,0.011,0,0.052,0,
0,0,0,0,0,0,0,0.011,0,0.039,
0,0,0,0,0,0,0,0,0,0,0;
f2=0,0.118,0,0,0,0,0,0,0,0,0,
0,0,0.012,0.037,0,0,0,0,0,0,0,
0,0,0,0.046,0,0,0,0,0,0,0,
```

```

0,0,0,0,0.018,0,0,0,0,0,0,
0.037,0.016,0,0,0,0.084,0,0.037,0,0,0,
0,0,0,0,0,0,0.099,0,0.056,0,0,
0,0,0,0.017,0.036,0,0,0.067,0,0,0,
0,0,0,0,0.018,0.043,0.036,0,0.069,0.037,0,
0,0,0,0,0,0,0.012,0,0.054,0,
0,0,0,0,0,0,0,0.012,0,0.040,
0,0,0,0,0,0,0,0,0,0,0,0;
f3=0,0.119,0,0,0,0,0,0,0,0,0,0,
0,0,0.015,0.038,0,0,0,0,0,0,0,0,
0,0,0,0.048,0,0,0,0,0,0,0,0,
0,0,0,0,0.019,0,0,0,0,0,0,0,
0.038,0.018,0,0,0,0.087,0,0.38,0,0,0,
0,0,0,0,0,0,0.104,0,0.064,0,0,
0,0,0,0.018,0.037,0,0,0.071,0,0,0,
0,0,0,0,0.019,0.044,0.037,0,0.075,0.038,0,
0,0,0,0,0,0,0,0.015,0,0.057,0,
0,0,0,0,0,0,0,0,0.015,0,0.042,
0,0,0,0,0,0,0,0,0,0,0,0;
alpha=1;
Enddata
!(1);Min=@sum(Location(i):@sum(Location(j)|j#ne#i:@sum(Capability(r)
:ff(i,r,j)*d(i,j)))));
!Subject to;
!(2);@for(Machine(k):@sum(Location(i):y(k,i))=1);
!(3);@for(Location(i):@sum(Machine(k):y(k,i))=1);
!(4);@for(Location(i):@for(Capability(r):@sum(Location(j)|j#ne#i:z(i
,r,j))<=11*(1-@sum(Location(j):z(j,r,i)))));
!Stochastic declarations;
@for(Machine(k):@SPSTGRNDV(1,ava(k)));
@for(Machine(k):@spdistbino(1,prb(k),ava(k)));
@spsampsize(1,nsamp);
!(5);@for(Set4(i,r):
[chance1] @sum(Location(j)|j#ne#i:z(j,r,i))>=1-
(@sum(Machine(k):a(k,r)*y(k,i)*ava(k))));
);
@spchance('CCP1','>=',prb1);
@for(Set4(i,r):
@spchance('CCP1',chance1)
);
!(6);@for(Location(i):@for(Capability(r):
[chance2]
@sum(machine(k):@sum(Capability(t)|t#ne#r:alpha*((f2(t,r)+f3(t,r))/2
)*a(k,t)*y(k,i)*ava(k)+(1-
alpha)*((f1(t,r)+f2(t,r))/2)*a(k,t)*y(k,i)*ava(k)))-
@sum(Location(j)|j#ne#i:ff(i,r,j))<=20*(1-
@sum(Location(j)|j#ne#i:z(j,r,i))));
);
@spchance('CCP2','>=',prb2);
@for(Location(i):@for(Capability(r):
@spchance('CCP2',chance2)
);
);
!(7);@for(Location(i):@for(Machine(k):@bin(y(k,i))));
!(8);@for(Location(i):@for(Capability(r):@for(Location(j):@bin(z(i,r
,j)))));
);
!(9);@for(Location(i):@for(Capability(r):@for(Location(j)|j#ne#i:ff(
i,r,j)<=999*z(j,r,i))));
);

```

APPENDIX 4: The Gurobi-Python Code of the submodel-1 for the UA-CBFLD Problem (Real-life Case Study)

```
import gurobipy as gp
from gurobipy import GRB
from gurobipy import quicksum
from time import time
start_time=time()
start_time=time()

## Indices & Sets
n_N = 12 #Number of departments
ix = range(n_N)

n_M = 6 #Number of lines
mx = range(n_M)

n_R = 11 #Number of REs
rx = range(n_R)

## Parameters
BX = 15
BY = 10
ratio = [2,2,2,2,3,3,2,2,2,2,2,2]
alpha=[[-1.554,-1.554,-1.554,-1.554,-2.33,-2.33,-1.554,-1.554,-1.554,-1.554,-1.554,-1.554],
        [-0.9382,-0.9382,-0.9382,-0.9382,-1.407,-1.407,-0.9382,-0.9382,-0.9382,-0.9382,-0.9382, 0.9382],
        [-0.5664,-0.5664,-0.5664,-0.5664,-0.849,-0.849,-0.5664,-0.5664,-0.5664,-0.5664,-0.5664, 0.5664],
        [-0.3419,-0.3419,-0.3419,-0.3419,-0.512,-0.512,-0.3419,-0.3419,-0.3419,-0.3419,-0.3419, 0.3419],
        [-0.2064,-0.2064,-0.2064,-0.2064,-0.309,-0.309,-0.2064,-0.2064,-0.2064,-0.2064,-0.2064, 0.2064],
        [-0.1246,-0.1246,-0.1246,-0.1246,-0.186,-0.186,-0.1246,-0.1246,-0.1246,-0.1246,-0.1246, 0.1246]]

beta=[[7.987,8.143,8.143,8.143,11.979,11.979,9.683,9.683,8.124,8.124,9.322,8.471],
      [6.206,6.327,6.327,6.327,9.307,9.307,7.523,7.523,6.312,6.312,7.243,6.582],
      [4.822,4.916,4.916,4.916,7.232,7.232,5.846,5.846,4.904,4.904,5.628,5.114],
      [3.746,3.820,3.820,3.820,5.619,5.619,4.542,4.542,3.811,3.811,4.373,3.973],
      [2.911,2.968,2.968,2.968,4.366,4.366,3.529,3.529,2.961,2.961,3.398,3.087],
      [2.262,2.306,2.306,2.306,3.392,3.392,2.742,2.742,2.300,2.300,2.640,2.399]]

f = [[0, 0, 0, 0, 0, 0, 0, 9500, 0, 0, 0, 0],
     [0, 0, 0, 0, 0, 0, 3900, 0, 0, 0, 0, 0],
     [0, 3200, 5200, 1000, 0, 0, 0, 0, 0, 0, 0, 0],
     [0, 0, 8200, 0, 2500, 0, 0, 0, 3500, 0, 0, 1520],
     [0, 0, 0, 5400, 0, 0, 1480, 0, 0, 0, 0, 2500],
     [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
     [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1480, 0],
     [0, 1000, 0, 0, 0, 0, 3000, 0, 0, 0, 0, 0],
     [0, 0, 4500, 0, 8400, 0, 0, 0, 0, 0, 0, 0],
     [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
     [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
     [3000, 0, 0, 3000, 0, 0, 0, 1300, 7000, 0, 0, 0]]

model = gp.Model("UA-FLP")
model.Params.OutputFlag=0

# Desicion variables
cx = model.addVars(ix,vtype=GRB.CONTINUOUS,lb=0,name="cx")
```

```

cy = model.addVars(ix,vtype=GRB.CONTINUOUS,lb=0,name="cy")
lx = model.addVars(ix,vtype=GRB.CONTINUOUS,lb=0,name="lx")
ly = model.addVars(ix,vtype=GRB.CONTINUOUS,lb=0,name="ly")
dx = model.addVars(ix,ix,vtype=GRB.CONTINUOUS,lb=0,name="dx")
dy = model.addVars(ix,ix,vtype=GRB.CONTINUOUS,lb=0,name="dy")
zx = model.addVars(ix,ix,vtype=GRB.BINARY,name="zx")
zy = model.addVars(ix,ix,vtype=GRB.BINARY,name="zy")

model.update()

obj = (quicksum(ff[i][j]*(dx[i,j]+dy[i,j]) for i in ix for j in ix if j!=i))

model.setObjective(obj, GRB.MINIMIZE)

#Constraints
for i in ix:
    for j in ix:
        if j!=i:
            model.addConstr(cx[j]-lx[j]>=cx[i]+lx[i]-BX*(1-zx[i,j]))
            model.addConstr(cy[j]-ly[j]>=cy[i]+ly[i]-BY*(1-zy[i,j]))
            model.addConstr(zx[i,j]+zx[j,i]+zy[i,j]+zy[j,i]==1)
            model.addConstr(dx[i,j]>=cx[i]-cx[j])
            model.addConstr(dy[i,j]>=cy[i]-cy[j])
            model.addConstr(dx[i,j]>=cx[j]-cx[i])
            model.addConstr(dy[i,j]>=cy[j]-cy[i])

for i in ix:
    model.addConstr(lx[i]<=ratio[i]*ly[i])
    model.addConstr(ly[i]<=ratio[i]*lx[i])
    model.addConstr(cx[i]+lx[i]<=BX)
    model.addConstr(cy[i]+ly[i]<=BY)
    model.addConstr(cx[i]-lx[i]>=0)
    model.addConstr(cy[i]-ly[i]>=0)

for i in ix:
    for m in mx:
        model.addConstr(2*ly[i]>=2*alpha[m][i]*lx[i]+beta[m][i])

for i in ix:
    for j in ix:
        if i==j:
            model.addConstr(dy[i,j]==0)
            model.addConstr(dx[i,j]==0)
            model.addConstr(zy[i,j]==0)
            model.addConstr(zx[i,j]==0)

for i in ix:
    for j in ix:
        model.addConstr(dx[i,j]==dx[j,i])
        model.addConstr(dy[i,j]==dy[j,i])

model.setParam('TimeLimit', 3600)
model.optimize()

if model.status==GRB.OPTIMAL:
    print('\nObjective Value=',model.ObjVal)
    print('\nObjective Bound=',model.ObjBound)

```

```
    print('\nOptimality Gap=',model.MIPGap)
elif model.status==GRB.INFEASIBLE:
    print('\nModel is infeasible')
elif model.status==GRB.UNBOUNDED:
    print('\nModel is unbounded')
else:
    print('\nOptimization ended with status %d' % model.status)
    print('\nObjective Value=',model.ObjVal)
    print('\nObjective Bound=',model.ObjBound)
    print('\nOptimality Gap=',model.MIPGap)

current_time=time()
elapsed_time=(current_time-start_time)
print("Elapsed Time = ", elapsed_time)
```



APPENDIX 5: The Gurobi-Python Code of the submodel-2 for the UA-CBFLD Problem (Real-life Case Study)

```

import gurobipy as gp
from gurobipy import GRB
from gurobipy import quicksum
from time import time
start_time=time()
start_time=time()

## Indices & Sets
n_N = 12 #Number of departments
ix = range(n_N)

n_M = 6 #Number of lines
mx = range(n_M)

n_R = 11 #Number of REs
rx = range(n_R)

## Parameters
BX = 14.9
BY = 10
BigM = 999999

lambda1 = [[0, 9500, 0, 0, 0, 0, 0, 0, 0, 0, 0],
            [0, 0, 1000, 3000, 0, 0, 0, 0, 0, 0, 0],
            [0, 0, 0, 3900, 0, 0, 0, 0, 0, 0, 0],
            [0, 0, 0, 0, 1480, 0, 0, 0, 0, 0, 0],
            [3000, 1300, 0, 0, 0, 7000, 0, 3000, 0, 0, 0],
            [0, 0, 0, 0, 0, 0, 8400, 0, 4500, 0, 0],
            [0, 0, 0, 1480, 2500, 0, 0, 5400, 0, 0, 0],
            [0, 0, 0, 0, 1520, 3500, 2500, 0, 5200, 3000, 0],
            [0, 0, 0, 0, 0, 0, 0, 1000, 0, 4200, 0],
            [0, 0, 0, 0, 0, 0, 0, 0, 1000, 0, 3200],
            [0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]]

aa=[[1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0],
    [1, 1, 1, 0, 0, 0, 1, 1, 1, 1, 1],
    [1, 1, 1, 0, 0, 0, 1, 1, 1, 1, 1],
    [1, 1, 1, 0, 0, 0, 1, 1, 1, 1, 1],
    [1, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0],
    [1, 1, 0, 1, 0, 0, 1, 0, 0, 0, 0],
    [1, 1, 0, 1, 0, 0, 1, 0, 0, 1, 0],
    [1, 1, 0, 1, 0, 0, 1, 0, 0, 1, 0],
    [0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0],
    [0, 0, 0, 0, 0, 1, 1, 0, 0, 0, 0],
    [0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0],
    [0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0]]

Teta=[20200,32800,8900,23200,37380,33700,40240,49540,24700,25000,13600]
UCAP=[40000,40000,50000,50000,50000,50000,50000,50000,50000,50000,50000]
LCAP=[20000,20000,20000,20000,20000,20000,20000,20000,20000,20000,20000]

model = gp.Model("UA1-FLP")
model.Params.OutputFlag=0

```

```

# Decision variables
f = model.addVars(ix,ix,vtype=GRB.INTEGER,lb=0,name="f")
H = model.addVars(ix,vtype=GRB.INTEGER,lb=0,name="H")
y = model.addVars(ix,rx,vtype=GRB.BINARY,name="y")
w = model.addVars(ix,ix,rx,rx,vtype=GRB.INTEGER,lb=0,name="w")
z = model.addVars(ix,ix,rx,rx,vtype=GRB.BINARY,name="z")
US = model.addVars(ix,vtype=GRB.BINARY,name="US")
Betam = model.addVars(ix,rx,vtype=GRB.INTEGER,lb=0,name="Betam")

model.update()

obj = (quicksum(f[i,j]*(dx[i][j]+dy[i][j]) for i in ix for j in ix if j!=i))

model.setObjective(obj, GRB.MINIMIZE)

for i in ix:
    for j in ix:
        for q in rx:
            for r in rx:
                if r != q:
                    model.addConstr(w[i,j,q,r]<=lambda1[q][r]*z[i,j,q,r])
                    model.addConstr(z[i,j,q,r]<=y[i,q])
                    model.addConstr(z[i,j,q,r]<=y[j,r])
                    model.addConstr(z[i,j,q,r]>=y[i,q]+y[j,r]-1)

for q in rx:
    model.addConstr(quicksum(y[i,q]*aa[i][q] for i in ix)>=1)

for i in ix:
    for r in rx:
        model.addConstr(y[i,r]<=aa[i][r])

model.addConstr(quicksum(w[i,j,q,r] for i in ix for j in ix for q in rx for r in rx if q !=
r)==quicksum(lambda1[q][r] for r in rx for q in rx if q != r))

for i in ix:
    for j in ix:
        model.addConstr(f[i,j]==quicksum(w[i,j,q,r] for q in rx for r in rx if r!=q))

for q in rx:
    for r in rx:
        if r!=q:
            model.addConstr(quicksum(w[i,j,q,r] for i in ix for j in ix)==lambda1[q][r])

model.addConstr(quicksum(H[i] for i in ix)>=quicksum(Teta[r] for r in rx))

for i in ix:
    for q in rx:
        model.addConstr(US[i]>=y[i,q])

for i in ix:
    model.addConstr(quicksum(y[i,q]*Teta[q] for q in rx)<=H[i])
    model.addConstr(H[i]<=UCAP[i]*US[i])
    model.addConstr(LCAP[i]*US[i]<=H[i])

for i in ix:
    for q in rx:

```

```

model.addConstr(Betam[i,q]<=y[i,q]*BigM)
model.addConstr(Betam[i,q]<=H[i])
model.addConstr(Betam[i,q]>=H[i]-(1-y[i,q])*BigM)

for q in rx:
    model.addConstr(quicksum(Betam[i,q] for i in ix)>=Teta[q])

for i in ix:
    model.addConstr(H[i]<=quicksum(f[i,j] for j in ix)*BigM)

model.setParam('TimeLimit', 3600)
model.optimize()

if model.status==GRB.OPTIMAL:
    print('\nObjective Value=',model.ObjVal)
    print('\nObjective Bound=',model.ObjBound)
    print('\nOptimality Gap=',model.MIPGap)
elif model.status==GRB.INFEASIBLE:
    print('\nModel is infeasible')
elif model.status==GRB.UNBOUNDED:
    print('\nModel is unbounded')
else:
    print('\nOptimization ended with status %d' % model.status)
    print('\nObjective Value=',model.ObjVal)
    print('\nObjective Bound=',model.ObjBound)
    print('\nOptimality Gap=',model.MIPGap)

for v in w.values():
    if v.X != 0:
        print("{}: {}".format(v.varName, v.X))

current_time=time()
elapsed_time=(current_time-start_time)
print("Elapsed Time = ", elapsed_time)

```