ASSEMBLY LINE BALANCING WITH MULTI-MANNED TASKS

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

BY

CEYHAN ERDEM ESİN

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

SEPTEMBER 2007

Approval of the thesis:

ASSEMBLY LINE BALANCING WITH MULTI-MANNED TASKS

submitted by **CEYHAN ERDEM ESİN** in partial fulfillment of the requirements for the degree of **Master of Science in Industrial Engineering Department, Middle East Technical University** by,

Prof. Dr. Canan Özgen ______________________
Dean, Graduate School of **Natural and Applied Sciences**

Prof. Dr. Çağlar Güven ______________________
Head of Department, **Industrial Engineering**

Prof. Dr. Ömer Kırca ______________________
Supervisor, **Industrial Engineering Dept., METU**

Assist. Prof. Dr. Ayten Türkcan ______________________
Co-Supervisor, **Biomedical Engineering Dept., Purdue University of USA**

Examining Committee Members:

Prof. Dr. Meral Azizoğlu ______________________
Industrial Engineering Dept., METU

Prof. Dr. Ömer Kırca ______________________
Industrial Engineering Dept., METU

Assist. Prof. Dr. Ayten Türkcan ______________________
Biomedical Engineering Dept., Purdue University of USA

Assist. Prof. Dr. Sedef Meral ______________________
Industrial Engineering Dept., METU

Assist. Prof. Dr. Ferda Can Çetinkaya ______________________
Industrial Engineering Dept., Çankaya University

Date: 28.09.2007

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

Name, Last name: Ceyhan Erdem, ESİN

Signature :

ABSTRACT

ASSEMBLY LINE BALANCING WITH MULTI-MANNED TASKS

ESİN, Ceyhan Erdem
M.S., Department of Industrial Engineering
Supervisor: Prof. Dr. Ömer KIRCA
Co-Supervisor: Assist. Prof. Dr. Ayten TÜRKCAN

September 2007, 126 pages

In this thesis, we define a new problem area for assembly lines. In the literature, there are various studies on assembly line balancing, but none of them consider multi-manned tasks, task to which at least two operators have to be assigned. Two mathematical models and one constraint programming model are developed for both Type-I and Type-II ALB problems. The objective of Type-I problem is to minimize the number of stations whereas the objective of Type-II problem is to minimize the cycle time. In addition to this, valid inequalities are introduced to make models more efficient. Moreover, heuristic algorithms for both types are developed for large-sized problems. All formulations are applied to a real case study and then experimental analysis are conducted for all formulations to see the effects of problem parameters on performance measures. Exact models are compared each other and performance of heuristic algorithms are compared against the lower bounds.

Keywords: Assembly Line Balancing, Multi-manned Tasks, Constraint Programming.

ÖZ

ÇOK-ADAMLI OPERASYONLAR İÇEREN MONTAJ HATTI DENGELEME

ESİN, Ceyhan Erdem
Yüksek Lisans, Endüstri Mühendisliği Bölümü
Tez Yöneticisi: Prof. Dr. Ömer KIRCA
Ortak Tez Yöneticisi: Yrd. Doç. Dr. Ayten TÜRKCAN

Eylül 2007, 126 sayfa

Bu tez çalışmasında montaj hattı dengeleme probleminde yeni bir alan tanımlanmıştır. Montaj hattı dengeleme problemi konusunda literatürde birçok çalışma bulunmaktadır ancak hiçbirinde en az iki operatöre atanması zorunlu çok-adamlı operasyonlar içeren montaj hatları üzerine çalışma bulunmamaktadır. Tip-1 ve Tip-2 problemleri için iki matematiksel model ve bir kısıt programlama modeli yazılmıştır. Tip-1 probleminde amaç toplam istasyon sayısını enazlamaktır, bunun yanında Tip-2 problemleri için amaç çevrim zamanını enazlamaktır. Bunlara ek olarak alan daraltma yöntemini uygulamak için modellere geçerli eşitsizlikler eklenmiştir. Ayrıca her iki tipte büyük boyutlu problemleri çözmek için sezgisel yöntemler geliştirilmiştir. Geliştirilen tüm modeller bir gerçek hayat problemi ile problem parametrelerinin performans ölçütlerine etkisini görebilmek için deneysel problemler üzerinde uygulanmıştır. Kesin çözüm veren modeller birbiriyle, sezgisel yöntemler de performanslarını analiz etmek için en iyi çözümle karşılaştırılmıştır.

Anahtar Kelimeler: Montaj Hattı Dengeleme, Çok-Adamlı Operasyonlar, Kısıt Programlama

To my family and my love

ACKNOWLEDGEMENTS

I would like to express my deepest gratitude to Prof. Dr. Ömer Kırca for his guidance, encouragement, advice and support throughout this study and especially for his help on problem definition, genius and brilliant ideas on developing mathematical models.

I would also like to express my great gratitude to Asst. Prof Dr. Ayten Türkcan for her valuable guidance, advice and support on the whole study and especially on constraint programming studies.

Most importantly, I would like to express my deepest thanks to my parents, Özden and Aysel Esin for their encouragement, patience and endless love.

My special thanks go to my fiancée, Şule, for her love, patience and support during this thesis study.

I am also grateful to my dear friend Ozan Hacıbekiroğlu for his invaluable support. Without him, this study would not be completed for a long time.

Finally, completion of this thesis would not be possible without my manager İlker Kocaman and my dear friends E. Berk Gürtekin, İ. Çağatay Kepek, Burak Hazer, H. Çağatay Babacan and Çilem Aygül for their consideration, support and encouragement since the beginning of the study.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

CHAPTER 1

INTRODUCTION

The assembly line balancing problem has been a focus of interest to the production and operations management community for the last four decades. Assembly line production methods are extensively used in high-volume manufacturing.

An assembly line consists of stations $k = 1,\ldots m$ arranged along a conveyor belt or a similar mechanical material handling equipment (Scholl et al., 2006). For large production volumes, division of tasks becomes a necessity. The problem of balancing an assembly line is a classical industrial engineering problem (Gökçen et al., 2006). Because of the high level of output in assembly line production, a small reduction in per unit cost results in substantial overall savings. Firstly Salveson formulated single-model assembly line balancing problem (SMALB) in 1956 and then several researchers (Arcus, 1966; Buxey, 1974; Freeman and Jucker, 1967; Gutjhar et al. 1964) have suggested both exact and heuristic procedures.

More recently, extensions to the SMALB such as multi model lines, mixed model lines, stochastic balancing and finally balancing with station paralleling have been proposed. *Single-model* lines are used to assemble large numbers of one type of a product. In *multi-model* lines, units of different models are produced in an intermixed sequence. *Mixed-model* lines are used to assemble different models of a product, the models are launched to the line one by one. Stochastic balancing allows for variability in the task times and considers the consequences of incompletion of tasks at stations (Pinto et al. 1975).

Traditionally, assembly lines are as a *serial line*, on which single stations are arranged along a straight conveyor belt. To improve the flexibility and a better balance of station loads, *U-shaped assembly lines* are designed. These lines are arranged as a narrow “U” shape.

Another improvement in assembly line balancing is allowing *parallelism*. With paralleling of stations, tasks are duplicated and are done in two or more paralleled work stations. This is often ‘necessity’ if the required cycle time of the line is shorter than the longest task time. But the amount of paralleling necessary is dependent upon the ratio of total task durations within the work center to the cycle time. In the paralleling of a station, the additional cost is the cost of all paralleled production facilities (equipment cost, labor cost etc.) of that station.

Furthermore, for the assembly of heavy workpieces, it is usually a necessity to design a *two-sided line,* in which two connected serial lines are in parallel (left-hand side and right-hand side) and operators work simultaneously at opposite sides of the same workpiece (Bartholdi, 1993).

The other improvement is to consider *assignment restrictions*. Several types of restrictions may curtail the assignment of tasks and so balance of the assembly line. These restrictions are *position related restrictions, task related restrictions, station related restrictions* and *operator related restrictions.*

As mentioned above, an assembly line is a set of sequential workstations linked by a material handling system. In each workstation, a set of tasks are performed using a predefined assembly process in which the following issues are defined:

- Task time, the time required to perform each task
- A set of precedence relationships, which determines the sequence of the tasks

The assembly work is completed on the line as the parts pass each station in sequence, with every station adding its work content to the assembly task. The *cycle time* of the assembly line is determined by the workstation with the maximum work content time.

Four formulation types are possible for this problem (Mastor, 1970, Boysen et al., 2006):

- *Type-I problems*, where the required production rate (i.e. cycle time), assembly tasks, tasks times, and precedence requirements are given and the objective is to *minimize the number of workstations*;
- *Type-II problems*, where the number of workstations or production employees is fixed and the objective is to *minimize the cycle time and maximize the production rate*. These types of balancing problems generally occur when the organization wants to produce the optimum number of items using a fixed number of workstations without purchasing new machines or expanding its facilities.
- *Type-E problems,* where the number of workstations and cycle time are given and the objective is to maximize the line efficiency. Line efficiency is the productive fraction of the line's total operating time t_{sum} and defined as $t_{sum}/(numberofstation)*(cycletime)$. Maximizing the line efficiency also minimize idle times of stations.
- *Type-F problems,* where the number of workstations and cycle time are given and the objective is to find a feasible balance.

In this study a new assembly line balancing problem is defined hence only *Type-I* and *Type-II* problems are of interest. Different from all other studies on assembly line problem, this problem includes some tasks which require more than one operator and so these tasks are assigned to two operators for the same time interval. In this thesis, these tasks are named as "*multi-manned tasks*". For example, consider a very heavy equipment "X". The task is *fix "X" to "Y"*. While one operator is fixing "X", the other operator has to hold "X" at the same time. Hence, this task requires two operators and should be called as a *multi-manned task*. Consequently, some stations of assembly lines, which include multi-manned tasks, require at least two operators. In fact the solution should satisfy that operators are not restricted to operate only these *multi-manned tasks*. In the same station they can make operations individually. This situation is mostly faced in heavy metal assembly lines such as construction machines (*excavators, backhoe-loaders etc*.) and truck&bus assembly lines.

In Chapter 2, literature survey on similar issues are classified and provided in detail. A brief literature review is given on *Assembly Line Balancing*. We then focus on *station paralleling*, *task paralleling* and *two-sided* assembly lines. Also constraint programming models for *resource constrained project scheduling problem (RCPSP)* are examined.

Two mathematical formulations for both *Type-I* and *Type-II* are given in Chapter 3. The first model is a general mathematical model which assigns tasks to stations and workers and finds the starting times of tasks at each station. The second model is written as if the problem is a *resource constrained project scheduling problem (RCPSP).* All tasks are thought as activities and resources are thought as operators. If a task is a *multi-manned task*, then capacity requirement of this task is 2, otherwise capacity requirement is 1. Also constraint programming formulations for both *Type-I* and *Type-II* are given. These constraint programming models are written as *RCPSP.* Next, after giving exact formulations, additional inequalities are presented for RCPSP and CP models of both Type I and Type II. The aim is to restrict the solution space and so provide faster solutions. Totally we have five solution alternatives for *Type-I* problem and five solution alternatives for *Type-II* problem. Afterwards, a case study is presented for a real problem which is an *excavator assembly line*. Firstly, the problem area is described, and the data of the assembly line is given. Next all ten solution alternatives are used to solve the problem and results and comparison table of the models are given. After solving the problem in the case study, computational experiments are conducted with test problems. Effects of these parameters on performance measures are analyzed, and comparison tables and graphs of presented models are given.

In Chapter 4, heuristic algorithms are developed and presented to solve large-sized problems. Heuristic algorithm for Type-I problem is a three-step algorithm. In the first step, tasks are assigned to stations without considering in-station precedence constraints. In the second step, a mathematical model is solved for every station to schedule tasks in stations considering precedence relations. The third step is a simple heuristic which relocates tasks whose start times exceed the cycle time. Heuristic algorithm for Type-II problem is a two-step algorithm. In the first step, tasks are again assigned to stations without considering in-station precedence constraints. In the second step, the same model is solved which is the second step of Type-I heuristic algorithm. Besides, output of these heuristics are used as additional inequalities for general mathematical models given in Chapter 3 for both Type-I and Type-II. The proposed heuristics and the general mathematical models with additional constraints are applied to the same case study. Also they are applied to same experimental test problems and the effects of parameters on performance measures are analyzed. In addition to this, heuristic algorithms are compared against with lower bounds. In the last section, totally seven formulations for Type-I and seven formulations for Type-II are compared with each other and best formulations for different problem parameters for both types are determined.

Finally, Chapter 5 concludes the study, gives a brief summary of major contributions and possible future research directions.

CHAPTER 2

LITERATURE REVIEW

In the literature, there are various studies on assembly line balancing models but none of these studies include *multi-manned* tasks. Assembly line balancing problem is a class of NP-*hard* optimization problem (Baybars, 1986), therefore heuristic methods are more common than exact solution procedures.

The first assembly line is credited to Henry Ford in 1915 after which it has been widely used in various production systems (Erel et al., 1998). Firstly Salveson formulated single-model assembly line balancing problem (SMALB) in 1956. He formulated Type-I ALBP as a linear programming problem, then Bowman (1960) was the first who suggested integer programming approaches for ALBP (Baybars, 1986). White (1961) improved Bowman's model and added binary variables to represent assignments.

Arcus (1966) developed Computer Method of Sequencing Operations for Assembly Lines (COMSOAL). The procedure assigns tasks to the current station from the available task set with respect to precedence relationship. Procedure ends when all tasks are assigned. Klein and Scholl (1996) presented Simple Assembly Line Balancing Optimization Method which is called as SALOME. This method is a bi-directional branch and bound procedure. In each node a local lower bound is found and planning direction is determined by this lower bound.

As mentioned before, *multi-manned tasks* have not been defined in the literature. Consequently, literature survey is focused on parallelism, RCPSP and constraint programming formulations.

The survey is classified into six categories:

- Survey on station paralleling
- Survey on task paralleling
- Survey on two-sided assembly lines
- Survey on SALBP formulated as RCPSP
- Survey on SALBP formulations with constraint programming techniques
- Survey on RCPSP formulations with constraint programming techniques

2.1 Station Paralleling

Freeman and Jucker (1967) were the first to suggest the paralleling stations. They demonstrated that under certain circumstances, the use of paralleled stations can result in lower total costs. They point out that paralleling would require additional production facilities since each paralleling task must be performed at more than one station. Consequently, the costs of these additional facilities need to be considered when the paralleling approach is followed. However, they did not offer any solution procedure or algorithm for the paralleling problem.

Buxey (1974) suggested two heuristic algorithms. One approach is based on the *positional weight method*, and obtains a balance by optimizing each station in turn and provides a unique solution. It takes both task durations and positional importance. The other approach generates a number of random sequences and selects the best. By employing a random number, generator elements from the ‘fit’ list is selected while this ‘fit’ list is continuously updated. The ‘fit’ list includes all elements that are currently free from precedence constraints and whose duration is not greater than the remaining unused station time. Both algorithms incorporate a limit on the number of work elements per station, and use stations in parallel *only for the longer elements*.

Pinto et. al. (1981) developed a branch and bound algorithm to include possible paralleling of two stations. They propose a model which allows the paralleling of stations to achieve a higher production rate and lower labor costs. A mathematical formulation, which is mixed-integer programming, is developed, and in the objective function both the cost of labor and cost of additional production facilities are considered. To facilitate comparison between the fixed facilities cost and the labor cost, the fixed costs are spread over the estimated useful life of the facility to obtain an equivalently daily facility cost. The branch and bound method proceeds by partitioning the set of all combinations into subsets of 'partial' combinations. A partial combination includes three classes of states: 'fixed paralleled', 'fixed not paralleled' and undecided. A 'full' combination is achieved by fixing all states either to be paralleled or not paralleled. Also a heuristic procedure is proposed to reduce the computational time requirement. However many heuristics for solving assembly line balancing problems do not guarantee the condition of Pinto's heuristic.

Sarker and Shantikumar (1983) suggested a general approach that can be applied for both serial or parallel line balancing. They develop a generalized heuristic to balance an assembly line where task times may be smaller or greater than the specified cycle time. The approach is composed of two phases. First phase deals with assigning the tasks to different stations according to the largest candidate rule and the second phase deals with reducing the balance loss of the line by trades and transfers. The paralleling, when the cycle time is less than at least one task time, is adopted in the first phase. But the computation is very complex and this method is not very suitable for practical applications. This method gives the most economical results, but it does not give the best of balancing and optimizing an assembly line. However, as mentioned above, this algorithm can be applied both to serial and parallel lines.

Bard (1989) was the first to develop a dynamic programming algorithm in which both stations and tasks are paralleled. The algorithm attempts to meet the required cycle time with the minimal total number of stations, while improving the line efficiency (reducing 'dead' time at stations) by using parallel stations, and selecting the parallel configuration with minimum cost due to the equipment duplication necessary at the parallel stations. He considered both task costs and equipment costs.

A second feature of this algorithm is that it takes into account unproductive or 'dead' time during a cycle. An advantage of dynamic programming is that the minimum cost function can be redefined without altering the form of the recursive relationship. The disadvantage is that explicit enumeration is required because bounds are not always available. Also it should be mentioned that the effectiveness of the basic dynamic programming algorithm is limited by the number of tasks and order strength of the network. Actually, this algorithm is effective up to approximately 100 tasks.

Udomkesmalee and Daganzo (1989) focused on a flexible mixed-model assembly line with parallel stations. An undesirable effect that may occur in such a situation, where task times for different models are allowed to vary, is that jobs (specific work orders for assembly of models) may get out of the initially planned processing sequence. This in turn affects the preplanned supply of parts and materials to the assembly line. The work analyzes the effect of variation in process times on the job sequences in a given parallel-processing assembly line. To solve the problem, the use of either material or job buffers is suggested to re-sequence materials or jobs, as needed, and the work provides analytical models to determine the size of the buffers needed.

Askin et al. (1997) proposed a non-linear integer program for assembly line balancing with station paralleling. Their model allows mixed-model production and the use of identical parallel workstations at each stage of the serial production system and considers equipment/tooling duplicating cost. The model is intended for a situation where the material handling system can alternately route units to each of the paralleled workstations at each stage. Thus, the model is applicable only for systems where intelligent, flexible material handling systems exist. Due to the complexity of computations, they introduce a greedy heuristic method but this method does not respond well to economical implications of equipment/tooling cost and idle worker time.

McMullen et al. (1998) developed a simulated annealing based technique for line balancing with station paralleling when one or more objectives are concerned.

Simulated annealing heuristic consists of eight algorithms. There are two primary objectives in the algorithm: one is the total cost (labor and equipment) per part, and the degree to which the desired cycle time was achieved. This heuristic provides good solutions if the cycle time performance is of concern. However, it does not perform well, when the primary concern is the design cost (such as equipment cost).

Süer (1998) studied on a heuristic to design parallel assembly lines. The objective is to determine the number of assembly lines with minimum number of operators. He propose a three-phase methodology; *balance the assembly line, determine parallel stations and determine parallel lines.* In the first phase tasks are grouped into stations. In the second phase, for each manpower level, many different alternatives are generated. In the third phase, a mathematical model determines the number of assembly lines. In the paper it is mentioned that configurations with fewer stations result in better assembly rates.

Buchkin et al. (2003) studied the problem of assembly line design, focusing on station paralleling and equipment selection. They provide a comprehensive approach for the assembly line design for minimizing the associated cost. In the studied system several equipment types as well as a human operator that are capable of performing the assembly operations are considered and paralleling of stations is allowed. The model is formulated to incorporate cost/weight factor for different paralleling situations whose formulation is similar to the model suggested by Askin et al. (1997) as mentioned above. Indeed, detailed analysis of the relation between problem parameters and the balancing improvement that can be achieved by paralleling are made, and also the problem is extended to deal with multiple equipment types option.

Finally, Gökçen et al. (2006) proposed new procedures and a mathematical model on SALBP with parallel lines. The approach they presented is quite different from Süer's (1998) approach. Number of parallel lines is not of interest. The objective is to balance more than one assembly line together. Three procedures are presented; *passive case procedure, active case procedure and different cycle time situation procedure.* Numerical examples are also given for all these procedures. The performance of active case procedure is tested on test problems and results are sufficient after comparison with optimal solutions. Indeed the proposed model provide a significant improvement in assembly line efficiency when more than one line is necessary.

2.2 Task Paralleling

Pinto et al.(1975) developed a branch and bound procedure for selecting tasks to be paralleled, with the objective to minimize total cost (labor, including overtime, and equipment duplication costs). Their work suggests that only certain tasks are duplicated in different stations. The problem is formulated as mixed integer programme. To facilitate the comparison between the cost of fixed facilities and the labor cost, they convert fixed costs into equivalent manpower. The branch and bound method applied here is the same with the method in Pinto (1981). They also provide an illustrative problem and give preliminary computational results. However this procedure is difficult to implement and control in practice.

Bukchin et al (2005) allowed a common task to be assigned to different stations. The goal is to minimize total cost of stations and task duplication. An optimal solution procedure based on a backtracking branch and bound algorithm, depth-first approach. Firstly a fast feasible solution is generated and an upper bound is found. Next, other solutions are obtained while improving the upper bound. Computational experiments are done and algorithm performance is examined. Proposed algorithm represents considerations more realistically in the design of a line and find near-optimal solutions.

2.3 Two-sided Assembly Lines

Bartholdi (1993) was the first who study on two-sided assembly lines and give data for a real assembly line. In this assembly line two operator work on the opposite sides of the station, left or right completing different tasks but working on the same item. Tasks are grouped in three sets in terms of their location to be completed; *left-side, right-side* and *either-side.* He implements a modified version of "First Fit".

The procedure identifies the next task on the list whose predecessors have been assigned, schedule this task at the first possible station and earliest possible time. He also shows, that for any fixed cycle time, minimum number of stations required for a *two-sided line* is always less than or equal to minimum number of stations required for *one-sided line*. He develops a computer program to test his procedure. Algorithm is fast enough but cannot find an optimal solution.

Kim et al. (2000) developed a genetic algorithm for two-sided assembly lines. The objective is to minimize the number of stations. They devise a genetic encoding and decoding scheme and genetic operators suitable for the problem. Five well-known test problems are used to analyze the procedure. The computational times are very reasonable, and in addition to this, the algorithm always found feasible solutions for large-sized problems. The major advantage of the proposed algorithm is that the algorithm can be applied to one-sided problems, type-II problems, and also can be used as an engine for line balancing software.

Lee et al. (2001) developed two assignment procedures. First one tries to maximize work relatedness and maximize work slackness. In other words, procedure, maximizes the slack time between the finishing time of one task and starting time of its successor which is in the opposite side of the same station. The second procedure is a group assignment procedure. This procedure assigns tasks to the same stations which are directly related in the precedence graph. Computational experiments are done to test the performance of the procedures and results are acceptable.

2.4 SALBP Formulated as RCPSP

Firstly De Reyck et al. (1997) formulated SALBP-1 as a special case of generalized RCPSP. They proposed an exact procedure but it is not competitive to specialized SALBP procedures. However, Sprecher (1999) presented a branch-and-bound algorithm for solving SALBP-1 as a RCPSP. His formulation is with a single renewable resource hence the problem can be solved with the general algorithm. The computational results are competitive with the specialized SALBP procedures formulated up to 1999.

2.5 SALBP Formulations With Constraint Programming Techniques

Constraint programming is a declarative programming for solving combinatorial satisfiability and optimization problems and it is often called as constraint logic programming. The search in constraint programming earns its efficiency from *destructive techniques* such as *constraint propagation* and *domain reduction* which are commonly called as *filtering algorithms*. Constraint propagation is based on an idea of using constraints actively to prune the domain (Hooker, 2000). Each constraint is assigned to a special filtering algorithm which will reduce the domains of variables in the constraint by removing the values that cannot take part in any feasible solution. When a variable's domain is modified, the effects of this modification are then communicated to any constraint that interacts with that variable. Filtering algorithms are analogous to cutting plane algorithms in integer programming.

Global constraints are collections of many small constraints that are presented as a unique constraint (Hooker, 2000). For resource constraint project scheduling problems, *cumulative* constraint is used as a global constraint in constraint programming formulations. This constraint includes four indices which are start times of activities, durations of activities, capacity requirements of activities and total maximum capacity of the resource for a given time interval.

Bockmayr et al. (2001) presented a hybrid solver for SALBP which combines integer programming and constraint programming (CP). They developed a branch-and-cut procedure with adding valid inequalities. In the constraint programming model, the *cumulative* constraint which is a global constraint is used. They could only give preliminary results and so computational results do not guarantee whether CP and IP, alone or together is able to find optimal solutions for unsolved instances in the literature.

2.6 RCPSP Formulations With Constraint Programming Techniques

Akker et al. (1991) presented a destructive lower bound for a number of RCPS problems based on column generation. They consider a number of variants of the RCPS problem with one or more resources. Because of the close relation between RCPS and the cumulative constraint in constraint programming, their method can be used as an efficient filtering algorithm for the cumulative constraint as well.

Klein et al. (1999) gave two meta-strategies for computing lower bounds (for minimization problems). In fact they are the only ones to compute a lower bound for the optimal makespan. They make use of a *destructive* procedure: constraint-propagation rules are applied in order to prove that no feasible schedule with makespan lower than T exists, yielding a fortiori that $T + 1$ is a lower bound.

Brucker et al. (2000) worked on a problem in which during the processing period of an activity constant amounts of renewable resources are needed where the available capacity of each resource type is limited.

Carlier et al. (2000) dealth with the computation of lower bounds for cumulative scheduling problems. These lower bounds take into account how resource requirements can be satisfied simultaneously for a given resource capacity.

Brucker (2002) used constraint propagation techniques for RCPS problems and machine scheduling problems. Also he discuss possible applications of these techniques in connection with lower bound calculations, branch-and-bound methods, and heuristics.

Brucker et al. (2003) studied on destructive lower bound for the multi-mode resource-constrained project scheduling problem with minimal and maximal time-lags. The lower bound calculations are based on two methods for proving infeasibility of a given threshold value T for the makespan. The first uses constraint propagation techniques, while the second is based on a linear programming formulation which is solved by a column generation procedure.

Finally Demassey et al. (2005) proposed a cooperation method between constraint programming and integer programming to compute lower bounds for the RCPS problem. The lower bounds are evaluated through linear-programming (LP) relaxations of two different integer linear formulations.

CHAPTER 3

MATHEMATICAL AND CONSTRAINT PROGRAMMING MODELS FOR TYPE-I AND TYPE-II PROBLEMS

In this chapter, firstly terminology and notation used in ALB and in this thesis are given. Then mathematical formulations for Type-I and Type-II problems are presented respectively.

3.1 Terminology, Assumptions and Notation

3.1.1 Terminology

Task: Indivisible work element partitioned from the total work content.

Station: Location where some portion of *tasks* are done repeatedly.

Task Time (P_i): Required time to perform *task* i .

Cycle Time: Maximum time available for each work cycle.

Precedence Relation: Relations between tasks due to the technological or operational restrictions.

Precedence Graph: A network based graphical representation of *precedence relations*. Nodes of the graph are *tasks*, node weights are *task times* and arcs are *precedence relations*. An example of a *precedence graph* is given in Figure 3.1.

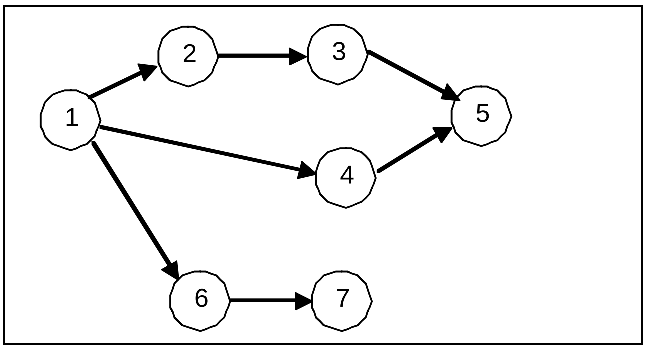

Figure 3.1: Example Precedence Graph (Problem)

As seen from Figure 3.1, *task 2* can be started only if *task 1* is completed. In other words *task 1* precedes *task 2.*

Predecessor(s) of a task: Set of tasks that must be completed before starting to this task. For example, from Figure 3.1, *predecessors set* of *task 5* is {1,2,3,4}.

Immediate Predecessor(s) of a task: Set of tasks that must be completed immediately before starting to this task. For example, from Figure 3.1, *immediate predecessor set* of *task 5* is {3,4}.

Successor(s) of a task: Set of tasks that cannot be started before completion of this task. For example, in Figure 3.1, *successors set* of *task* 2 is {3,5}.

Immediate Successor(s) of a task: Set of tasks that cannot be started immediately before completion of this task. For example, in Figure 3.1, *immediate successor set* of *task 2* is 3.

Precedence Relation Path: Paths that are constructed by connected arcs in the precedence graph. In Figure 3.1, there are three *precedence relation paths* which are {1,2,3,5}, {1,4,5} and {1,6,7}. The order of the set is from left to right. Simply *path* will be used instead of *precedence relation path* in the following chapters.

Precedence Path Set of a task: Set of tasks which are in the left side of this task in the *precedence relation path.* There may be more than one *precedence path set of a task*. In Figure 3.1, there is only one *precedence path set of "task 7"* which is {1, 6, 7}. However, there are two *precedence path sets of "task 5"* which are {1, 2, 3, 5} and {1, 4, 5}.

Precedence Subset: First element of this duplicate set is the *immediate predecessor* and the second element is the *immediate successor*. For the precedence graph given in Figure 3.1, *precedence subsets* are {1,2}, {1,4}, {1,6}, {2,3}, {3,5}, {4,5} and {6,7}.

Precedence Set: Set of all *precedence subsets.*

Disconnectivity Subset: Duplicate set whose elements are not connected in the given *precedence graph.* In other words, if one task is not element of another task's *path*, then these tasks construct a *disconnectivity subset.* For the precedence graph given in Figure 3.1, task 2 and task 4 are not included in the same path. So {2,4} is one of the disconnectivity subsets of the given precedence graph. The other disconnectivity subsets are {3,4}, {2,6}, {3,6}, {4,6}, {5,6}, {2,7}, {3,7}, {4,7}, {5,7}.

Disconnectivity Set: Set of all *disconnectivity subsets.*

Immediate Precedence Matrix: A 0-1 upper-triangular square matrix which represents *precedence relations*. If *task i* is the *immediate predecessor* of *task j*, then the value of "row i, column j" is "1", otherwise its value is "0". The *immediate precedence matrix* for the *precedence graph* given in Figure 3.1 is presented in Table 3.1.

Table 3-1: Example Immediate Precedence Matrix

Tasks	1	2	3	4	5	6	7
1	-	1	0	1	0	1	0
2		-	1	0	0	0	0
3			-	0	1	0	0
4				-	0	0	0
5					-	0	0
6						-	1
7							-

Precedence Matrix: This matrix is again a 0-1 upper-triangular square matrix. But the value of "row i, column j" is "1" if *task i* is the *predecessor* of *task j* otherwise its value is "0". The *precedence matrix* for the *precedence graph* given in Figure 3.1 is presented in Table 3.2.

Table 3-2: Example Precedence Matrix

Tasks	1	2	3	4	5	6	7
1	-	1	1	1	1	1	1
2		-	1	0	1	0	0
3			-	0	1	0	0
4				-	1	0	0
5					-	0	0
6						-	1
7							-

Duplicate Pair of a Multi-manned task: For some mathematical formulations, multi-manned tasks are presented as two duplicate pair tasks. For example, in Figure 3.1, assume that "task 3" is a multi-manned task. Then a new task is created, say "task 8" and (3,8) tasks are *duplicate pair of a multi-manned task*. In other words a multi-manned task is divided into two pairs. In fact task times remain same. For example, say *task time* of "task 3" which is a multi-manned task is 30 minutes. But this task has to be done by two operators in 30 minutes. This task is visualized as duplicate pairs and duplicate pairs can be performed by one operator but again in 30 minutes.

Duplicate Task Set: Set of *duplicate pairs of multi-manned tasks*. For the problem given in Figure 3.1 assuming "task 3" and "task 6" are *multi-manned tasks*, *duplicate task set* is {(3,8),(6,9)}.

Set of Tasks: Set of tasks. For the problem given in Figure 3.1, *set of tasks* is {1,2,3,4,5,6,7}.

Set of Single tasks: Set of tasks which can be performed by one operator. For the problem given in Figure 3.1, assuming only “task 3” and “task 6” are *multi-manned tasks*, *set of single tasks* is $\{1,2,4,5,7\}$.

Set of Multi-manned tasks: Set of tasks which can be performed by two operator. For the problem given in Figure 3.1, assuming only “task 3” and “task 6” are *multi-manned tasks*, *set of multi-manned tasks* is $\{3,6\}$.

Multi-manned task percentage: This is obtained by multiplying the division of number of *multi-manned tasks* to the total number of tasks with 100. For the problem given in Figure 3.1, assuming only “task 3” and “task 6” are *multi-manned tasks*, *multi-manned task percentage* is $100*(2/7)=28,57\%$

Set of Dummy Tasks: Set of dummy duplicate pairs. The number of duplicate pairs is the same with the number of stations. For the problem given in Figure 3.1 assuming the upper bound for the number of stations is 3 and also assuming “task 3” and “task 6” are *multi-manned tasks* , then *set of dummy task* will be $\{(10,11), (12,13), (14,15)\}$. Tasks in *set of dummy tasks* should be added to the *set of multi-manned tasks.*

Set of All Tasks: Task set which duplicate pairs of multi-manned tasks are added to *set of tasks*. For the problem given in Figure 3.1 assuming “task 3” and “task 6” are *multi-manned tasks*, *set of all tasks* is $\{1,2,3,4,5,6,7,8,9\}$.

Makespan of a Station: Completion time of a station.

Makespan of the Line: Completion time of the line.

Order Strength (Os): This is the total number of *precedences* in the precedence network divided by the largest number of precedences which is $[n*(n-1)]/2$ (Bartholdi 1993 and Akagi et al. 1983). For the example precedence matrix given in Table 3.2, the number of *precedences* is **10**. The largest number of precedences is **21** (i.e. $[7*(7-1)]/2$ **)**. Then “*Os*” values is $10/21=0,47$. Actually *Os* value is between $[0,1]$. If there is not any precedence relation, then Os is “0”, if all tasks have precedence relation (i.e. upper side of diagonal of the precedence matrix is “1”) then the *Os* value is “1”.

3.1.2 Assumptions

In this thesis, both Type-I and Type-II SALB problems are in interest. Major assumptions are as follows:

- Paced line with a fixed *cycle time* (given or variable)
- Mass-production of one homogenous product
- *Task times* are deterministic
- All stations are adequately equipped with respect to machines and operators
- Line is serial but at most two worker is allowed in stations
- *Precedence graphs* are known and fixed.

3.1.3 Notation

The following notations are used throughout the mathematical formulations.

Indices:

i,j:	Task number = {1,2,..,N}
u:	Operator number = {1,2}
k:	Station number = {1,2,..,K}

Parameters:

CYC:	Cycle Time (For Type-I problems)
MKS_k:	Makespan of *k*’th station
MKL:	Makespan of the line

P_i: Task time of *i*'th task

N: Number of tasks which are element of *set of tasks*

M: Number of multi-manned tasks which are element of *set of multi-manned tasks*

T: Double sum of all task times. (i.e. $2 * \sum_i P_i$)

K: Number of stations (For Type-II problems)

<u>*Sets:*</u>

SET: Set of tasks. *SET = {1,2,..N}*

SST: Set of single tasks.

SMT: Set of multi-manned tasks.

SDT(i,j): Set of dummy duplicate pairs.

SEAT: Set of all tasks.

SEATD: Set of all tasks with dummy duplicate pairs.

SAT_k: Set of tasks assigned to station "k".

$PPS_{i,n}$: *n'th* precedence path set of task "i".

Pred(i,j): Precedence set

Duplicate(i,j): Duplicate task set.

Discon(i,j): Disconnectivity set

<u>*Decision Variables:*</u>

$$X_{u,i}^{k} = \begin{cases} 1, & \text{if } i\text{'th task is assigned to } u\text{'th operator in station } k. \\ 0, & \text{otherwise} \end{cases}$$

$$Y_{u,i,j}^{k} = \begin{cases} 1, & \text{if } i'\text{th task is performed before } j'\text{th task by} \\ & u'\text{th operator in station } k. \\ 0, & \text{otherwise} \end{cases}$$

$$W_{u}^{k} = \begin{cases} 1, & \text{if } u'\text{th operator is utilized in station } k. \\ 0, & \text{otherwise} \end{cases}$$

$$V^{k} = \begin{cases} 1, & \text{if } k'\text{th station is opened.} \\ 0, & \text{otherwise} \end{cases}$$

$ST_{u,i}^{k}$ = Start time of i'th task in k'th station by u'th operator.

For some formulations, *X, Y, and ST variables are used without "k" or "u" indices.*

3.2 Mathematical Formulations For Type-I SALB Problems

We have developed two mathematical models for Type-I problem. The first one is a general mixed integer mathematical model which gives exact solution. The second one is a mathematical model which is written as if the problem is a RCPSP. This model also gives exact solutions. For both models, objective is to minimize the number of stations, when the cycle times are given.

3.2.1 General Mathematical Model-I

Minimize $\sum_{k} k * V^{k}$ **(3.1)**

Subject to

$$\sum_{u} \sum_{k} X_{u,i}^{k} = 1 \qquad \forall i \in SEAT \qquad \textbf{(3.2)}$$

$$\sum_{k} \{k * \left(\sum_{u} X_{u,j}^{k}\right) - k * \left(\sum_{u} X_{u,i}^{k}\right)\} \geq 0 \qquad \forall i, j \in Pred(i, j) \qquad \textbf{(3.3)}$$

$$\sum_{i} (X_{u,i}^{k} * P_i) - CYC * W_{u}^{k} \leq 0 \qquad \forall i \in SEAT, \forall k \in K, \forall u \in \{1,2\} \qquad \textbf{(3.4)}$$

$$\sum_u X^k_{u,i} - \sum_u X^k_{u,j} = 0 \qquad \forall i,j \in Duplicate(i,j) \quad \forall k \in K \qquad \textbf{(3.5)}$$

$$X^k_{u,i} + X^k_{u,j} \le 1 \qquad \forall i,j \in Duplicate(i,j) \quad \forall k \in K, \forall u \in \{1,2\} \qquad \textbf{(3.6)}$$

$$T * \left\{1 - \left(\sum_u X^k_{u,j}\right)\right\} + \sum_u St^k_{u,j} - \left\{\left(P_i * \sum_u X^k_{u,i}\right) + \sum_u St^k_{u,i}\right\} \ge 0$$

$$\forall i,j \in Pred(i,j), \forall k \in K \qquad \textbf{(3.7)}$$

$$CYC * V^k - \left\{\sum_u St^k_{u,i} + \{P_i * \sum_u X^k_{u,i}\}\right\} \ge 0 \qquad \forall i \in SEAT, \forall k \in K \qquad \textbf{(3.8)}$$

$$\sum_u St^k_{u,i} - \sum_u St^k_{u,j} = 0 \qquad \forall i,j \in Duplicate(i,j) \quad \forall k \in K \qquad \textbf{(3.9)}$$

$$\left.\begin{array}{l} St^k_{u,i} - St^k_{u,j} - \{P_j * X^k_{u,j}\} + T * \left(1 - Y^k_{u,j,i}\right) \ge 0 \\ St^k_{u,j} - St^k_{u,i} - \{P_i * X^k_{u,i}\} + T * \left(Y^k_{u,j,i}\right) \ge 0 \end{array}\right\} \quad \forall i,j \in Discon(i,j), \forall k \in K, \forall u \in \{1,2\} \qquad \textbf{(3.10)}$$

$$St^k_{u,i} - \left(T * X^k_{u,i}\right) \le 0 \qquad \forall i \in SEAT, \forall k \in K, \forall u \in \{1,2\} \qquad \textbf{(3.11)}$$

$$\sum_u W^k_u \le u * V^k \qquad \forall k \in K, \forall u \in \{1,2\} \qquad \textbf{(3.12)}$$

$$\left.\begin{array}{l} X^k_{u,i}, Y^k_{u,i,j}, W^k_u, V^k \in (0,1) \\ St^k_{u,i} \ge 0 \end{array}\right\} \quad \forall i \in SEAT, \forall k \in K, \forall u \in \{1,2\} \qquad \textbf{(3.13)}$$

The objective function (3.1) minimizes the number of stations. Note that we do not focus on the total number of utilized operator.

Constraint (3.2) ensures that all tasks must be assigned to one station and one operator. Precedence restrictions are in constraint (3.3) such that for Pred(i,j), task '*i*' must be assigned before or at the same station that task 'j' is assigned. Constraint (3.4) set the cycle time as an upper bound for the total task times of assigned tasks to each operator. Constraint (3.5) guarantees that duplicate pairs of multi-manned tasks are assigned to the same station. When the duplicate pairs of multi-manned tasks are assigned to same station, constraint (3.6) prevents them to be assigned to the same operators. Constraint (3.7) ensures that if job 'i' is assigned to one operator, job 'j' has to start after job 'i' finishes no matter which operator that job 'j' and job 'i' are assigned. In other word this constraint makes *in-scheduling* of the station in terms of the precedence relationship. In addition to constraint (3.4), constraint (3.8) expresses the completion time of operators has to be lower than the cycle time. Constraint (3.9) imposes that the start time of duplicate pairs of multi-manned tasks must be same for every station. Constraint set (3.10) prevents tasks to be performed in the same time interval by the same operators. Namely, if this constraint is removed, start time of tasks will be same for the same operator. Instead of *SEAT*, *discon(i,j)* is used to reduce the domain. There is no need to use tasks that have precedence relations for this constraint set. Constraint (3.11) sets the start time "0" if task 'i' is not assigned to a station and an operator. Constraint (3.12) imposes that If at least one operator in station 'k' is utilized, then that station must be opened. Finally constraint set (3.13) represents the binary and non-negativity restrictions.

It should be noted that if precedence relations get more complex (i.e. *Order Strength* increases), then number of variables decreases.

3.2.2 Mathematical Model Written as RCPSP-I

This model is written as if the problem is RCPSP. Tasks are presented as activities and operators are resources. In addition to this, new dummy duplicate pairs are created to impose the station breaks. Task times of these duplicate pairs are set to "0" and start times are defined with respect to the cycle time.

Mathematical model is as follows;

Minimize $$\sum_k k * V^k \tag{3.1}$$

Subject to

$$\sum_u X_{u,i} = 1 \qquad \forall i \in SEAT, \forall u \in \{1,2\} \tag{3.14}$$

$$X_{u,i} + X_{u,j} \leq 1 \qquad \forall i, j \in Duplicate(i, j), \forall u \in \{1,2\} \tag{3.15}$$

$$\sum_u St_{u,j} - \left\{ \left(P_j * \sum_u X_{u,i} \right) + \sum_u St_{u,i} \right\} \geq 0 \qquad \forall i, j \in \Pr ed(i, j), \forall u \in \{1,2\} \tag{3.16}$$

$$\sum_u St_{u,i} - \sum_u St_{u,j} = 0 \qquad \forall i, j \in Duplicate(i, j), \forall u \in \{1,2\} \tag{3.17}$$

$$\left. \begin{array}{l} St_{u,i} - St_{u,j} - \{P_j * X_{u,j}\} + T * (1 - Y_{u,j,i}) \geq 0 \\ St_{u,j} - St_{u,i} - \{P_i * X_{u,i}\} + T * (Y_{u,j,i}) \geq 0 \end{array} \right\} \quad \forall i, j \in Discon(i, j), \forall u \in \{1,2\} \tag{3.18}$$

$$St_{u,i} - (T * X_{u,i}) \leq 0 \qquad \forall i \in SEAT, \forall u \in \{1,2\} \tag{3.19}$$

$$T * W_u^k - St_{u,i} + (k-1) * CYC \geq 0 \qquad \forall i \in SET, \forall u \in \{1,2\}, \forall k \in K \tag{3.20}$$

$$\sum_u ST_{u,i} - k * CYC = 0 \qquad i = k^{th} element\ of\ SDT(i, j) \tag{3.21}$$

$$\left. \begin{array}{l} X_{u,i}^k, Y_{u,i,j}^k, W_u^k, V^k \in (0,1) \\ St_{u,i} \geq 0 \end{array} \right\} \quad \forall i \in SEAT, \forall k \in K, \forall u \in \{1,2\} \tag{3.22}$$

and constraint (3.12) is in the constraint set.

Constraints (3.14), (3.15), (3.16), (3.17), (3.18) and (3.19) are actually same with constraints (3.2), (3.6), (3.7), (3.9), (3.10) and (3.11), respectively. In constraints (3.14), (3.15), (3.16), (3.17), (3.18) and (3.19), the only difference is "k" indices are removed. Constraint (3.20) determines which station and operator will be utilized. Constraint (3.21) imposes the station breaks through the assembly line. Actually in this model, the whole line is thought as a station. Tasks are first assigned to operators and then assigning tasks to stations and scheduling of tasks are done with adding station breaks.

3.3 Mathematical Formulations For Type-II SALB Problems

As for Type-I problems, two formulations are developed for Type-II problems. The first one is a general mixed integer mathematical model which gives exact solutions. The second one is a mathematical model which is written as if the problem is a RCPSP. This model also gives exact solutions. For both models, objective is to minimize cycle times where the number of stations are given.

3.3.1 General Mathematical Model-II

This formulation is almost same with the model presented in section 3.2.1. This time the objective is to minimize the cycle time with respect to given number of stations. The objective function is;

$$\text{Minimize} \qquad CYC \qquad \textbf{(3.23)}$$

Constraints (3.2), (3.3), (3.5), (3.6), (3.7), (3.8), (3.9), (3.10), (3.11), (3.13) are again in the constraint set. In constraint (3.4), W_u^k is removed because number of stations is given in this model. New constraint is:

$$\sum_i (X_{u,i}^k * P_i) - CYC \le 0 \qquad \forall i \in SEAT, \forall k \in K, \forall u \in \{1,2\} \qquad \textbf{(3.24)}$$

3.3.2 Mathematical Model Written as RCPSP-II

The approach of the model is the same with the model presented in section 3.2.2. Tasks are presented as activities and operators are resources. Again in this model, new dummy duplicate pairs are created to impose the station breaks. Task times of these duplicate pairs are set to “0” and start times are defined with respect to cycle time.

Mathematical model is as follows;

Minimize CYC **(3.23)**

Constraints (3.14), (3.15), (3.16), (3.17), (3.18), (3.19) and (3.21) are in the constraint set.

Actually the difference of the constraint set of this model from the model presented in section 3.2.2 is that constraints (3.12) and (3.20) are removed. The whole line is thought as a station and like in section 3.2.2, scheduling of tasks are done by adding station breaks.

3.4 Constraint Programming Formulations for TYPE-I and TYPE-II Problems

In constraint programming formulations, models are written as if the problem is RCPSP. These models are written and solved to make comparison between the mathematical models and constraint programming models.

All tasks are thought as activities, resources are thought as operators. If a task is a multi-manned task, then capacity requirement of this task is 2, otherwise capacity requirement is 1. Like the mathematical models in sections 3.2.2 and 3.3.2, new dummy duplicate tasks are created to impose the station breaks. But in this formulation, dummy duplicate tasks are presented as multi-manned tasks whose capacity requirement is two, they are not presented as duplicate pairs as in mathematical formulations. Task times of these dummy duplicate pairs are set to “0” and start times are defined with respect to cycle time. Different from mathematical models in sections 3.2.2 and 3.3.2, *duplicate task set* and *disconnectivity set* are not needed in constraint programming formulations.

In the literature, assembly line balancing problem for both types are written and solved for the first time in this thesis. Firstly notations for constraint programming are given, then formulations for both types are presented in detail in the following sections.

3.4.1 Notation

The following notation is used throughout the constraint programming formulations.

Indices:

i: Task number = {1,2,..,N}

k: Station number = {1,2,..,K}

Parameters:

a(i).start : Start time of task "i".

a(i).end : Completion time of task "i".

Demand(i): Capacity requirement of task "i". If task "i" is a multi-manned task, then *demand(i)* is 2, otherwise *demand(i)* is 1.

3.4.2 Constraint Programming Formulation-I

Minimize $$\sum_k k * V^k \qquad \textbf{(3.1)}$$

Subject to

$$a(k).start \geq a(i).end \qquad k = k^{th} element\ of\ SDT, \forall i \in SET \qquad \textbf{(3.25)}$$

$$a(j).start \geq a(i).end \qquad \forall i, j \in \Pr ed(i, j) \qquad \textbf{(3.26)}$$

$$a(i)\ requires\ Demand(i) \qquad \forall i \in SET \cup SDT \qquad \textbf{(3.27)}$$

$$T * V^k - a(i).start + (k-1) * CYC \geq 0 \qquad \forall i \in SET \ , \forall k \in K \qquad \textbf{(3.28)}$$

$$a(i).start - k * CYC = 0 \qquad i = k^{th} element\ of\ SDT, \forall k \in K \qquad \textbf{(3.29)}$$

Objective function (3.1) minimizes the number of stations. Constraint (3.25) imposes that every task has to be done before the last dummy duplicate task. Precedence restrictions are in constraint (3.26) that for Pred(i,j), task 'i' must be completed before or at the same time that task 'j' is started. Constraint (3.27) ensures that single tasks will be assigned to one operator and multi-manned tasks will be assigned to two operators. Constraint (3.28) determines which stations will be opened. Finally, dummy duplicate tasks are assigned to start times (station-breaks) with constraint (3.29).

3.4.3 Constraint Programming Formulation-II

This formulation is also written as RCPSP. The aim of the model is to minimize cycle time. The model is as follows;

Minimize $\quad CYC \qquad$ **(3.23)**

Constraints are (3.25), (3.26), (3.27) and (3.29) which are given in the constraint set of Type-I problem.

The main advantage of constraint programming formulation is that to generate models for both types are less tedious and models can be revised in a very short time.

3.5 Proposed Valid Inequalities

Additional inequalities are added to *model written as RCPSP* and *constraint programming model* for both Type-I and Type-II problems. The objective is to reduce the domain, to find exact solutions and to decrease the solution times. In the following sections, these inequalities are presented for each model. After completion of introducing additional inequalities, a brief summary table for all formulations developed up to now is presented.

3.5.1 Additional Inequalities for Type-I Formulations

In this section set of extra constraints are introduced to two proposed models for Type-I problems. Valid inequality set for the *model written as RCPSP* and *constraint programming model* is almost the same. The valid inequalities are introduced with brief examples.

The approach of introducing new set of additional constraints are the same for both models, hence they are presented in the same section. The valid inequality set is composed of lower bounds for *start times* of tasks. Recall the terminology of *precedence relation path*. Tasks are ordered in the *paths*, with respect to their *precedence relations*. Tasks may be included in more than one path or a task may not be included in any path if it has not any precedence relation. Consider a path {a, b, l, p, r}. As defined before that the order is from left to right, "a" precedes "b" and "b" precedes "l". Precedence path set of "task l" is {a, b}. Therefore with no doubt, *start time* of "task l" has to be greater than the total of *task times* of "task a" and "task b". But for the same problem, there may be another path like {m, n, l, s, t}. Consequently, *start time* of "task l" has to be greater than the total of *task times* of "task m" and "task n". In this case, instead of using two constraints for one task, the maximum of the total *task times* is used as a lower bound of *start time* of task "l". The equation for the above situation is:

$$\sum_u St_{u,l} \geq Max\left[\{P_a + P_b\}, \{P_m + P_n\}\right]$$

Valid inequality set introduced to the model is composed of inequalities for all tasks and the generalized constraint set is as follows:

$$\sum_u St_{u,i} \geq \underset{PPS_n}{Max}\left[\sum_{j \in PPS_{i,n}} P_j\right] \qquad \forall i \in SET \qquad \textbf{(3.30)}$$

A valid equality for one duplicate pair of a multi-manned task is sufficient, so *SET* is used for the constraint set.

<u>Example 1:</u> For an example problem, the precedence relation is same with the diagram given in Figure 3.1. Multi-manned tasks are “3” and “6”. Duplicate task set is $\{(3,8),(6,9)\}$. Task time are given in the Table 3.3.

Table 3-3: Task times of an example problem

Task	Task Time (min.)
1	40
2	20
3	40
4	15
5	10
6	20
7	30

For example 1, Valid inequality set will be,

$$\sum_u St_{u,1} \geq 0 \qquad \sum_u St_{u,2} \geq 40$$
$$\sum_u St_{u,3} \geq 60 \qquad \sum_u St_{u,4} \geq 40$$
$$\sum_u St_{u,5} \geq 100 \qquad \sum_u St_{u,6} \geq 40$$
$$\sum_u St_{u,7} \geq 60$$

This set is both added to the *model written as RCPSP* and *constraint programming model*.

3.5.2 Additional Inequalities for Type-II Formulations

In this section the same set of extra constraints are introduced to two proposed models for Type-II problems.

The objective of Type-II problem is to minimize cycle time, but the additional constraint set is totally same with the constraint set given for Type-I problem which is constraint (3.30).

3.6 Brief Summary of All Formulations

Due to the fact that there are various formulations; abbreviations, objective functions and characteristics of all formulations are presented in Table 3.4. In the following sections, these abbreviations will be used for proposed formulations.

Table 3-4: Brief summary and abbreviations of all formulations

	Name of Formulation	Section	Abbreviation	Solution	Other
TYPE-I	General Mathematical Model-I	3.2.1	GMM-I	Optimal	
	Mathematical Model Written as RCPSP-I	3.2.3	RCPSP-I	Optimal	
	Constraint Programming Formulation-I	3.4.2	CP-I	Optimal	
TYPE-II	General Mathematical Model-II	3.3.1	GMM-II	Optimal	
	Mathematical Model Written as RCPSP-II	3.3.3	RCPSP-II	Optimal	
	Constraint Programming Formulation-II	3.4.3	CP-II	Optimal	
TYPE-I	Additional Inequality Set for Model Written as RCPSP-I	4.1.2	AD-RCPSP-I	Optimal	Constraint set (3.30) is added
	Additional Inequality Set for Constraint Programming Model-I	4.1.2	AD-CP-I	Optimal	Constraint set (3.30) is added
TYPE-II	Additional Inequality Set for Model Written as RCPSP-II	4.2.2	AD-RCPSP-II	Optimal	Constraint set (3.30) is added
	Additional Inequality Set for Constraint Programming Model-II	4.2.2	AD-CP-II	Optimal	Constraint set (3.30) is added

For Type-I and Type-II problems, GMM(I-II), RCPSP(I-II) and CP(I-II) find optimal solutions.

AD-RCPSP-I, AD-CP-I, AD-RCPSP-II and AD-CP-II are generated by adding constraint set (3.30) to the models RCPSP-I, CP-I, RCPSP-II and CP-II respectively.

In the following section, in order to illustrate all these formulations, these are tested with a real case study which is an excavator assembly line and additional test problems.

3.7 Test of Formulations

In this section first both Type-I and Type-II formulations are solved for a real case study which is an excavator assembly line. Next, experimental runs for 120 test problems are conducted for all formulations and then comparison of formulations are presented.

3.7.1 Case Study

In the previous sections, proposed formulations for assembly line balancing with multi-manned tasks are presented. In order to generate these formulations, some definitions and notations are firstly defined in the literature. However, these are illustrated with a small-sized example problem. Consequently, a real case study is necessary to present formulations to make them understandable. As mentioned before, this is an excavator assembly line of a company located in Ankara.

In this section first the problem environment is presented. Next, data set of the problem, precedence relations, terminology, sets and parameters are given. After solving the problem with all 10 formulations (for both Type-I and Type-II), the optimal solutions and comparison table of all formulations are presented.

3.7.1.1 Problem Environment

A leading company in construction machine manufacturing, constructs an assembly line for excavator production. The aim is to produce excavators for both domestic and global market. Assembly line is composed of two independent lines. In the first line, there is no multi-manned task hence this line is not of interest for this case study. In the second line, 4 work stations are fixed on the conveyor. At most 3 additional stations are optional after product leaves the conveyor. In addition to this, the company's objective is to produce 3 units per shift. Since one shift is actually

480 minutes, desired cycle time 160 minutes. This assembly line is a paced line with deterministic task times. All stations are equally equipped and at most two workers are allowed in workstations.

3.7.1.2 Data Set

There are 29 indivisible tasks. 7 of them are *multi-manned tasks*. *Set of tasks* is {1,2,..,29}. *Set of multi-manned tasks* is {1, 5, 6, 20, 21, 23, 27}. The *multi-manned task percentage* is 24,13%. Task numbers, task times and immediate predecessors of tasks are given in Table 3.5.

Table 3-5: Data Set of the Case Study

Task Number	Task Time (min.)	Immediate Predecessor
1	75	-
2	40	1
3	75	1
4	90	1
5	15	4
6	25	5
7	60	6
8	40	7
9	20	8
10	70	5
11	70	10
12	100	11
13	10	12
14	30	12
15	70	13,14
16	40	15
17	10	10
18	25	14
19	40	18
20	10	9,19
21	75	20
22	70	1
23	25	20
24	10	16
25	10	16
26	10	5
27	35	20
28	30	20
29	65	20

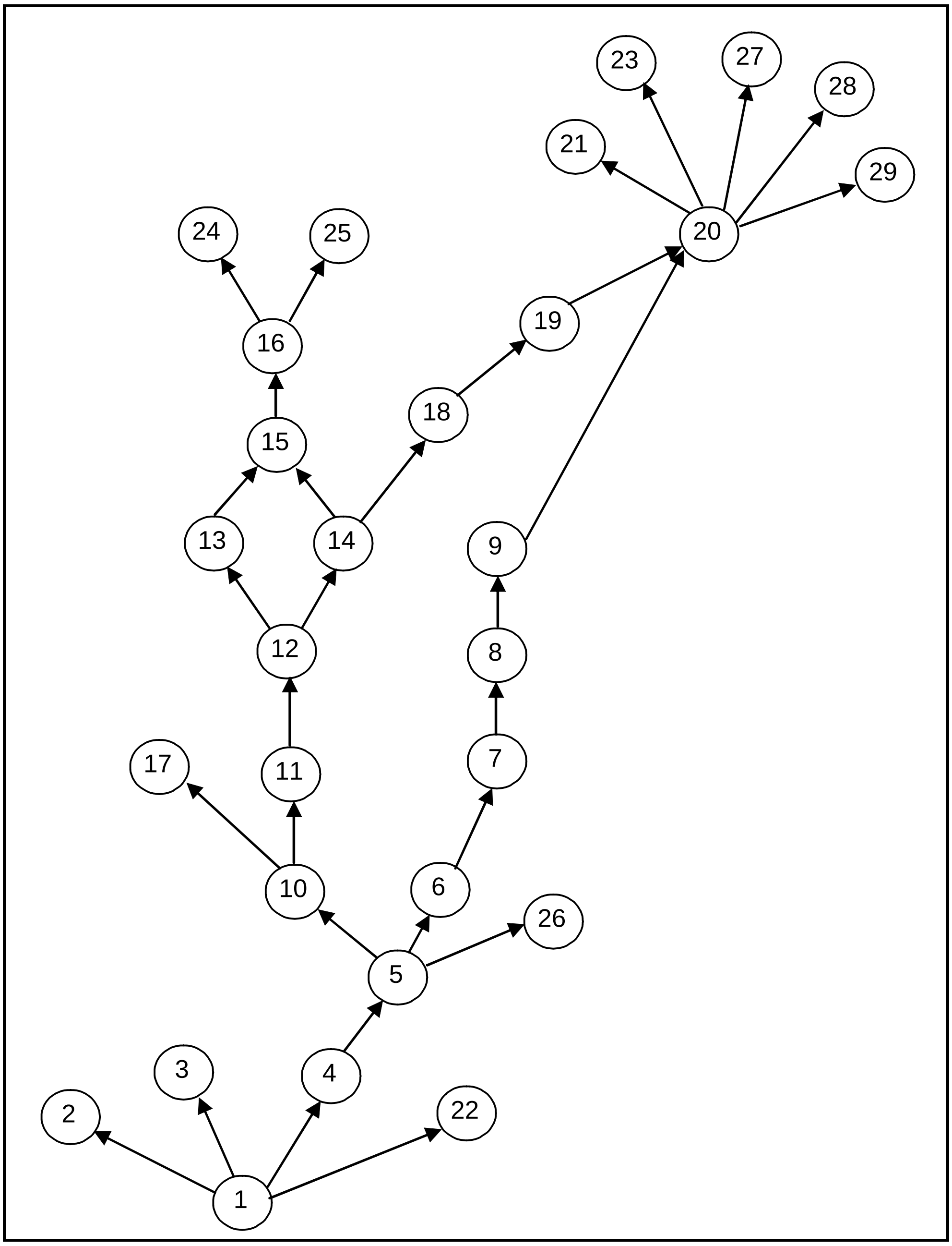

Figure 3.2: Precedence Graph of the Case Study

The precedence graph of the case study is given in Figure 3.2. *The precedence graph* presented in Figure 3.2 has 19 *precedence relation paths* which are {1, 2}, {1, 3}, {1, 4, 5, 10, 17}, {1, 4, 5, 10, 11, 12, 13, 15, 16, 24}, {1, 4, 5, 10, 11, 12, 13, 15, 16, 25}, {1, 4, 5, 10, 11, 12, 14, 15, 16, 24}, {1, 4, 5, 10, 11, 12, 14, 15, 16, 25}, {1, 4, 5, 10, 11, 12, 14, 18, 19, 20, 21}, {1, 4, 5, 10, 11, 12, 14, 18, 19, 20, 23}, {1, 4, 5, 10, 11, 12, 14, 18, 19, 20, 27}, {1, 4, 5, 10, 11, 12, 14, 18, 19, 20, 28}, {1, 4, 5, 10, 11, 12, 14, 18, 19, 20, 29}, {1, 4, 5, 6, 7, 8, 9, 20, 21}, {1, 4, 5, 6, 7, 8, 9, 20, 23}, {1, 4, 5, 6, 7, 8, 9, 20, 27}, {1, 4, 5, 6, 7, 8, 9, 20, 28}, {1, 4, 5, 6, 7, 8, 9, 20, 29}, {1, 4, 5, 26}, {1, 22}.

Duplicate task set is {(1,30), (5,31), (6,32), (20,33), (21,34), (23,35), (27,36)}. Hence *set of all tasks* is {1,2,..,36}.

Set of dummy tasks is {(37,38), (39,40),…, (36+(2*k)-1,36+2*k)} for any given number of station “k”.

The number of *precedences* is **<u>189</u>**. The largest number of precedences is **<u>406</u>** (i.e. $[29*(29-1)]/2$ **)**. For this case, “*Os*” value is $189/406 = 0{,}465$.

3.7.1.3 Solutions

In this case study, the company wants to see whether 3 excavators can be assembled in one shift (i.e. cycle time is 160 minutes) with 5 workstations. Consequently, this case study is solved for both Type I and Type II problems. In the first section, Type-I formulations’ solutions are presented with given cycle time **<u>160</u>** minutes (i.e. $CYC = 160$). In the second section, Type-II formulations’ solutions are presented with given **<u>5</u>** work stations (i.e. $K = 5$) and all workstations have two operators.

3.7.1.3.1 Type-I Solutions

Type-I formulations, GMM-I and RCPSP-I are solved using optimization software Cplex 10.1, and CP-I is solved using optimization software Ilog-OPL Studio 3.6.1. The objective is to minimize the number of stations. The upper bound of number of station is **<u>9</u>** (i.e. $K = 9$). Since models of GMM-I and RCPSP-I are too long, they cannot be presented but model of CP-I is given in Appendix A.1.

GMM-I Solution

$Objective\ Function : 6$
$Number\ of\ variables : 5089$
$Nodes : 3428$
$Solution\ Time(\text{sec.}) : 611$

RCPSP-I Solution

Set of dummy tasks is added to this formulation. Since $K = 9$, *set of dummy tasks* is {(37,38), (39,40), (41,42), (43,44), (45,46), (47,48), (49,50) , (51,52), (53,54)}.

$Objective\ Function : 7$
$Number\ of\ variables : 1497$
$Nodes : 30611$
$Solution\ Time(\text{sec.}) : 3\ hour$ timeout

CP-I Solution

The same set of dummy tasks which is given for RCPSP-I is added to CP-I model.

$Objective\ Function : 6$
$Number\ of\ variables : 238$
$Choice : 68$
$Solution\ Time(\text{sec.}) : 418$

AD-RCPSP-I Solution

The constraint set added to RCPSP-I model is as follows:

ST_1_1 + ST_2_1 >= 0	ST_1_16 + ST_2_16 >= 520
ST_1_2 + ST_2_2 >= 75	ST_1_17 + ST_2_17 >= 250
ST_1_3 + ST_2_3 >= 75	ST_1_18 + ST_2_18 >= 450
ST_1_4 + ST_2_4 >= 75	ST_1_19 + ST_2_19 >= 475
ST_1_5 + ST_2_5 >= 165	ST_1_20 + ST_2_20 >= 515
ST_1_6 + ST_2_6 >= 180	ST_1_21 + ST_2_21 >= 525
ST_1_7 + ST_2_7 >= 205	ST_1_22 + ST_2_22 >= 75
ST_1_8 + ST_2_8 >= 265	ST_1_23 + ST_2_23 >= 525
ST_1_9 + ST_2_9 >= 305	ST_1_24 + ST_2_24 >= 560
ST_1_10 + ST_2_10 >= 180	ST_1_25 + ST_2_25 >= 560
ST_1_11 + ST_2_11 >= 250	ST_1_26 + ST_2_26 >= 180
ST_1_12 + ST_2_12 >= 320	ST_1_27 + ST_2_27 >= 525
ST_1_13 + ST_2_13 >= 420	ST_1_28 + ST_2_28 >= 525
ST_1_14 + ST_2_14 >= 420	ST_1_29 + ST_2_29 >= 525
ST_1_15 + ST_2_15 >= 450	

$Objective\ Function : 7$
$Number\ of\ \text{var}iables : 1497$
$Nodes : 22549$
$Solution\ Time(\text{sec.}) : 3\ hour$ timeout

AD-CP-I Solution

The constraint set which is added in the previous section is added to the model CP-I.

$Objective\ Function : 6$
$Number\ of\ \text{var}iables : 238$
$Choice : 12$
$Solution\ Time(\text{sec.}) : 323$

3.7.1.3.2 Type-II Solutions

Type-II formulations, GMM-II and RCPSP-II are solved using optimization software Cplex 10.1 and CP-II is solved using optimization software Ilog-OPL Studio 3.6.1. The objective is to minimize cycle time (CYC). The model of CP-II is given in Appendix A.2.

GMM-II Solution

$Objective\ Function : 170$
$Number\ of\ variables : 3621$
$Nodes : 6303$
$Solution\ Time(\text{sec.}) : 3\ hour$ timeout

RCPSP-II Solution

Set of dummy tasks is added to this formulation. Since $K = 5$, *set of dummy tasks* is {(37,38), (39,40), (41,42), (43,44), (45,46)}.

$Objective\ Function : 180$
$Number\ of\ variables : 1229$
$Nodes : 850068$
$Solution\ Time(\text{sec.}) : 3\ hour$ timeout

CP-II Solution

The same set of dummy tasks which is given for RCPSP-II, is added to CP-II model.

$Objective\ Function : 170$
$Number\ of\ variables : 239$
$Choice : 109835704$
$Solution\ Time(\text{sec.}) : 3\ hour$ timeout

AD-RCPSP-II Solution

The constraint set which is added in AD-RCPSP-I is added to the model RCPSP-II.

$Objective\ Function : 175$
$Number\ of\ variables : 1229$
$Nodes : 376902$
$Solution\ Time(\text{sec.}) : 3\ hour$ timeout

AD-CP-II

The constraint set which is added in AD-CP-I is added to the model CP-II.

$Objective\ Function: 165$
$Number\ of\ variables: 239$
$Choice: 4628041$
$Solution\ Time(\sec.): 824$

3.7.1.4 Summary of Solutions and Results

The results of formulations for Type-I and Type-II are presented in Table 3.6 and 3.7, respectively.

Table 3-6: Results of Type-I Case Study

	TYPE-I (Given Cycle Time = 160 min.)				
Formulations	**GMM-I**	**RCPSP-I**	**CP-I**	**AD-RCPSP-I**	**AD-CP-I**
Number of Stations	6	7	6	7	6
Number of Variables	5089	1497	238	1497	238
Nodes / Choice Points	3428	30611	68	22549	12
Solution Time (Sec.)	611	3 hour timeout	418	3 hour timeout	323
Gap Between Optimal Solution	0%	17%	0%	17%	0%

Table 3-7: Results of Type-II Case Study

	TYPE-II (Given Number of Stations = 5)				
Formulations	GMM-II	RCPSP-II	CP-II	AD-RCPSP-II	AD-CP-II
Cycle Time	170	180	170	175	165
Number of Variables	3621	1229	239	1229	239
Nodes / Choice Points	6303	850068	109835704	376902	4628041
Solution Time (Sec.)	3 hour timeout	3 hour timeout	3 hour timeout	3 hour timeout	824
Gap Between Optimal Solution	3%	9%	3%	6%	0%

For Type-I case study, except RCPSP-I and AD-RCPSP-I, other formulations find optimal solutions in reasonable solution times. Additional inequalities for CP-I formulation decreases solution time from 418 seconds to 323 seconds. For Type-II case study, AD-CP-II find optimal solutions and solution time of AD-CP-II is acceptable (824 seconds). GMM-II, CP-II, AD-RCPSP-II find near optimal solutions when the solver is interrupted after 3 hours. As a result, acceptable alternative formulations for this case study with respect to objective functions and solution times are AD-CP-I and AD-CP-II for Type-I and Type-II respectively. Comparison of all formulations for both types are done in the following section by applying computational experiments.

3.7.2 Computational Experiments

In this section, experimental runs for 120 test problems are conducted for all formulations, namely, 1200 runs are done for the whole experiment. Mathematical models are coded with C++ and solved by CPLEX 10.1.0., constraint programming models are solved by Ilog OPL Studio 3.6.1 on Pentium IV 1.6 Ghz with 512 Mb Ram. All formulations are solved with a time limit of 3 hours, if the model cannot be solved in 3 hours, the best optimum solution in 3 hours is recorded for the solution.

Due to the fact that there is no research on assembly line balancing with multi-manned tasks, formulations presented in this thesis cannot be compared with any well-known test problem results for both Type-I and Type-II problems. First, parameters of the experimental runs are given, then performance measures are introduced for comparison of formulations. Finally, after completing experimental runs, comparison of formulations and discussion of the parameters' effects on performance measures are made.

3.7.2.1 Experimental Parameters

Since there is no test problem with multi-manned tasks in the literature, random test problems are generated using mainly the same approach with Kaplan (2004) and Ege (2001). Experimental runs are done according to the following parameters:

Set of Task (N): Number of tasks are selected as 30, 40, 50 and 60.

Task Times (P_i): All task times are integer and uniformly distributed between 10 and 100 (Kaplan, 2004 and Ege, 2001).

Cycle Time (CYC) : Cycle times are given for Type-I problems. They are taken as 1.4 and 1.8 times the maximum task time.

Number of Stations (K): Number of stations are given for Type-II problems. K_1 and K_2 are taken as number of stations.

$$K_1 = \left(\sum_i P_i \right) / \{2 * 1,4 * \max(P_i)\} \qquad K_2 = \left(\sum_i P_i \right) / \{2 * 1,8 * \max(P_i)\}$$

Order Strength (Os): Problems with having Os value as 0.10, 0.25, 0.50, 0.75 and 0.9 are used in the experiments.

Multi-manned task percentage ($100 * (M / N)\%$): Three different percentages are used for the experiment which are 25%, 50% and 75%.

Consequently 120 test problems are generated according to the above parameters.

3.7.2.2 Performance Measures

Performance measures used for the experimental analysis are as follows:

Cycle Time (*CYC*) : Objective function value for Type-II problems.

Number of Stations (*K*): Optimum number of stations opened for Type-I problems.

Solution Time (Sec.): The running time of the solver.

Gap Percentage (%): The gap between the optimum solution and the solution found.

$$Gap\% = \left\{100 * \left([Found - Optimum] \middle/ Optimum \right)\right\}$$

3.7.2.3 Discussion on the Experimental Results

Firstly, effects of the parameters and effects of additional inequalities on performance measures are analyzed. Then comparison of formulations are given and acceptable formulations for different parameter sets are determined. These analysis are done separately for Type-I and Type-II formulations.

In order to present parameter sets in an easy way, they will be presented as $\{\alpha, \phi, \delta, \gamma\}$ where,

$\alpha =$ Set of Task, $\alpha = 30, 40, 50, 60$

$\phi =$ Multi-manned task percentage, $\phi = 25\%, 50\%, 75\%$

$\delta =$ Order Strength, $\delta = 0.10, 0.25, 0.50, 0.75, 0.90$

$\gamma =$ *Cycle Time* or *Number of Stations* multiplier, $\gamma = 1.4, 1.8$

3.7.2.4 Discussion of Type-I Formulations

All results of Type-I formulations for 120 test problems is given in Appendix B. Tables from Table B.1 to B.5 represents the objective functions, solution times and gap percentage for parameters defined in the section 3.8.1.

3.7.2.5 Effects of Parameter Set on Solution Times

In the experiment, number of tasks are selected as 30, 40, 50 and 60. When the number of tasks are increased, while the other parameters are fixed, it is easily seen in the Figures 3.3 and 3.4, solutions times of formulations GMM-I and CP-I increases.

In the following figures, parameters *multiplier*, *Os value* and *multi-manned task percentage* are given at the bottom of the graph, parameter *number of task* is presented as colored lines.

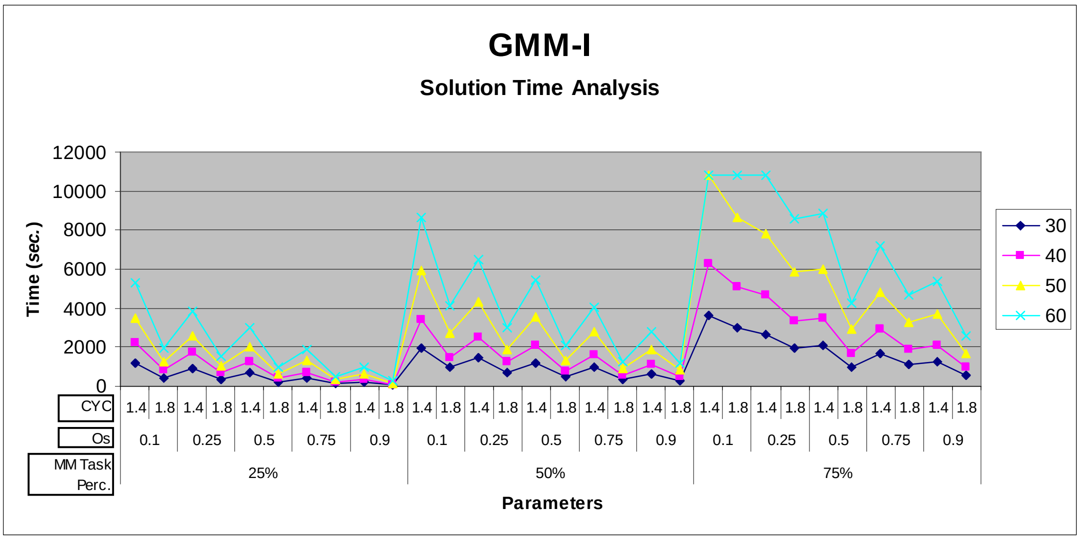

Figure 3.3: GMM-I solution time analysis

If the multi-manned task percentage is increased while the other parameters are kept constant, solutions times of GMM-I and CP-I increases dramatically. For formulations GMM-I and CP-I, when Os value increases while the other parameters are fixed, solution times decrease apparently.

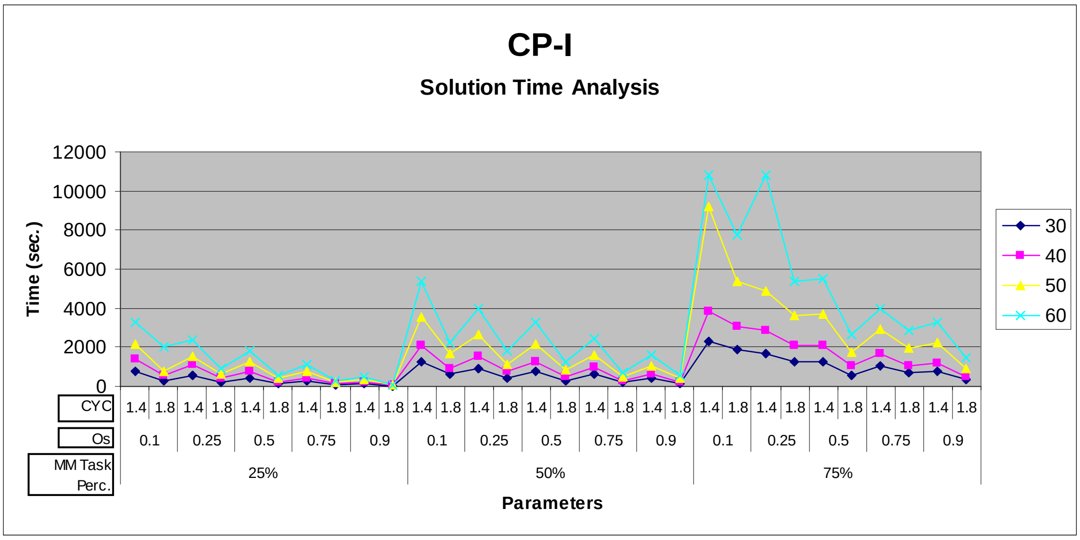

Figure 3.4: CP-I solution time analysis

Finally, when the cycle time multiplier increases from 1.4 to 1.8, these three formulations' solution times decreases about 54% and 58% for GMM-I and CP-I respectively. Formulation RCPSP-I is not completed in 3 hours for all test problems hence effects of parameters on solution times for this formulation cannot be analyzed.

3.7.2.6 Effects of Parameter Set on Gap Percentages

For all task numbers, GMM-I and CP-I found optimum solutions. RCPSP-I could not find optimum solutions in 3 hours. The gap percentage analysis of RCPSP-I is given in Figure 3.5.

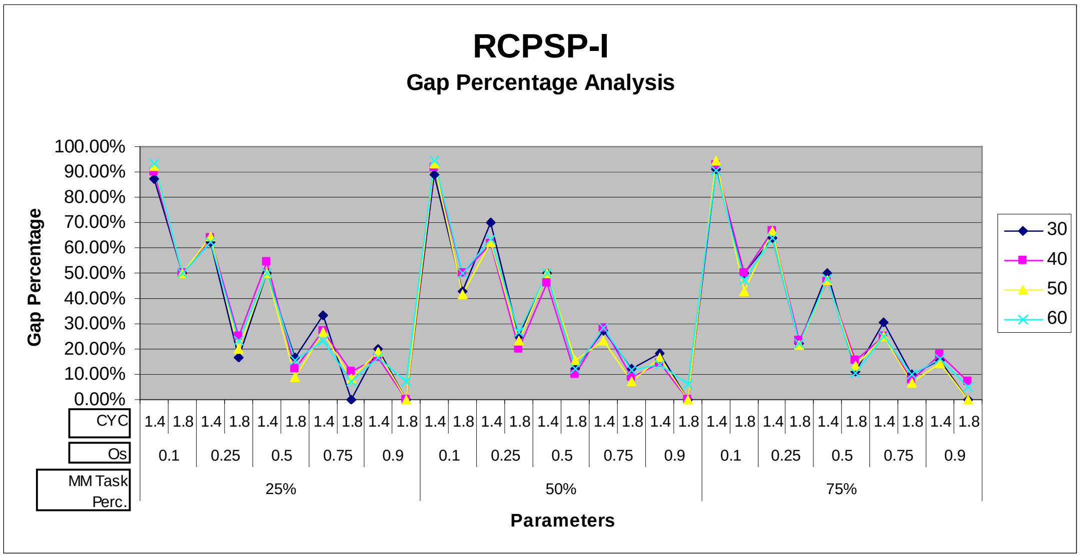

Figure 3.5: RCPSP-I gap percentage analysis

Number of tasks actually does not affect gap percentage of RCPSP-I formulation. Multi-manned task percentage slightly increases the gap percentage. When Os value increases, gap percentage decreases apparently. The same manner is acceptable for cycle time multiplier. When it changes from 1.4 to 1.8, gap percentage decreases dramatically about 68% in average.

3.7.2.7 Effects of Additional Inequalities

Additional inequalities to RCPSP-I does not affect the solution time, but gap percentages decrease. Improvement on average gap percentage is about 10% in average which is given in Figure 3.6.

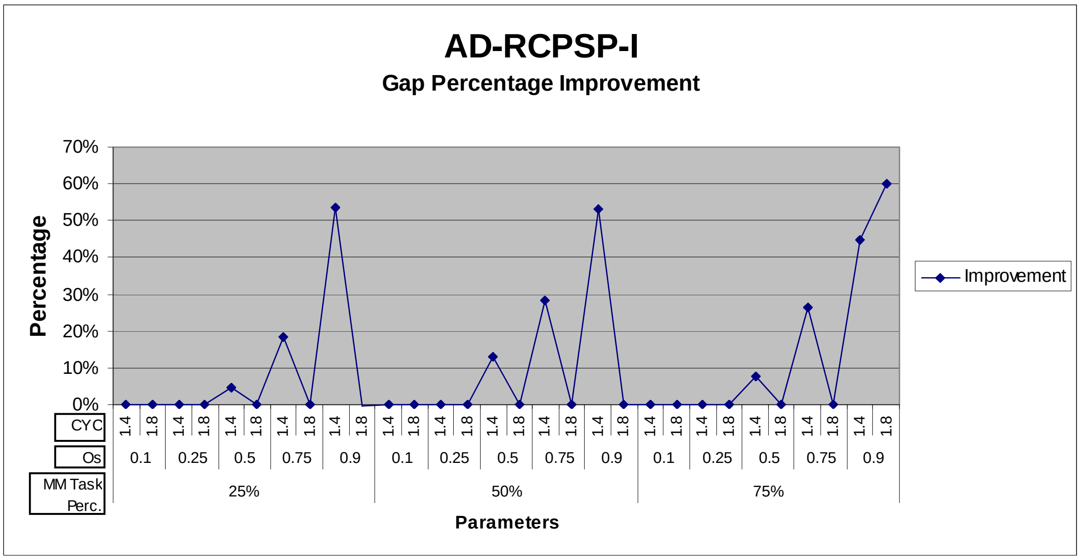

Figure 3.6: AD-RCPSP-I gap percentage improvement

As it is seen in Figure 3.6, only for problems with Os values 0.75 and 0.9, additional inequalities affect the gap percentage about 4% to 21%.

The improvement on solution times when valid inequalities added to CP-I problem is about 5% in average.

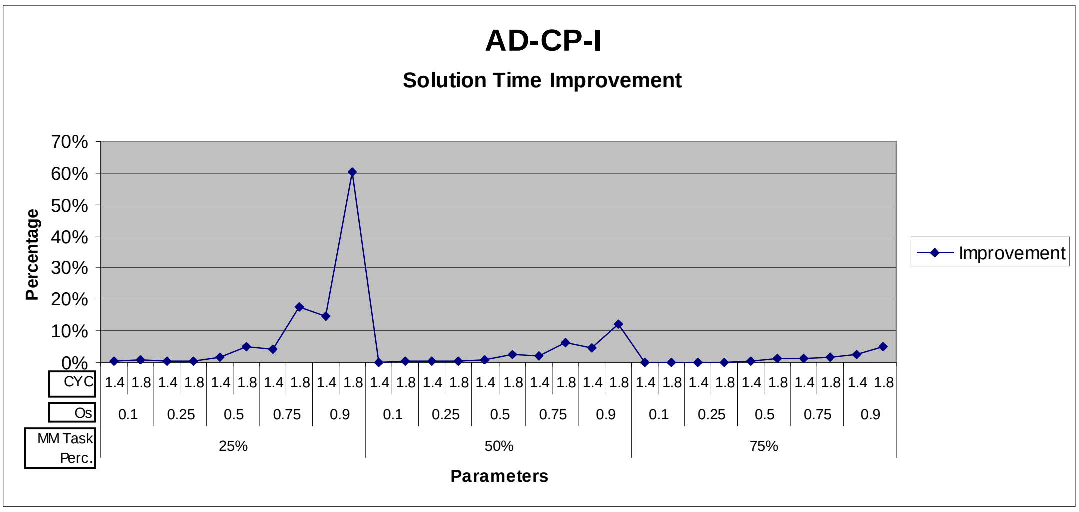

Figure 3.7: AD-CP-I solution time improvement

For parameter set $\{\alpha, 25\%, 0.9, 1.8\}$, the improvement is about 60% but as multi-manned task percentage increases, the improvement on average solution time decreases dramatically which is presented in Figure 3.7.

3.7.2.8 Discussion of Type-II Formulations

Results of Type-II formulations for 120 test problems is given in Appendix B. Tables from Table B.6 to B.10 represents the objective functions, solution times and gap percentage for parameters defined in the section 3.8.1.

3.7.2.9 Effects of Parameter Set on Solution Times

GMM-II, CP-II and RCPSP-II cannot be completed for any test problem in 3 hours. Therefore effects of parameters on solution times for these formulations cannot be analyzed.

3.7.2.10 Effects of Parameter Set on Gap Percentages

For small-sized problems, GMM-II and CP-II can find optimum solutions. Gap percentage analysis for GMM-II, CP-II and RCPSP-II are presented in Figure 3.8, 3.9 and 3.10 respectively.

For both GMM-II and CP-II formulations, when number of tasks increases while the other parameters are kept constant, gap percentages increase clearly. If the multi-manned task percentage increases, gap percentages increase slightly. However, when Os values are increased, gap percentages of GMM-II and CP-II decrease apparently and CP-II's descent is more than GMM-II's descent but RCPSP-II's solution time declines slightly. Finally, if number of station multiplier is changed from 1.4 to 1.8, gap percentages decrease about 25%, 27% and 15% for GMM-II, CP-II and RCPSP-II in average, respectively.

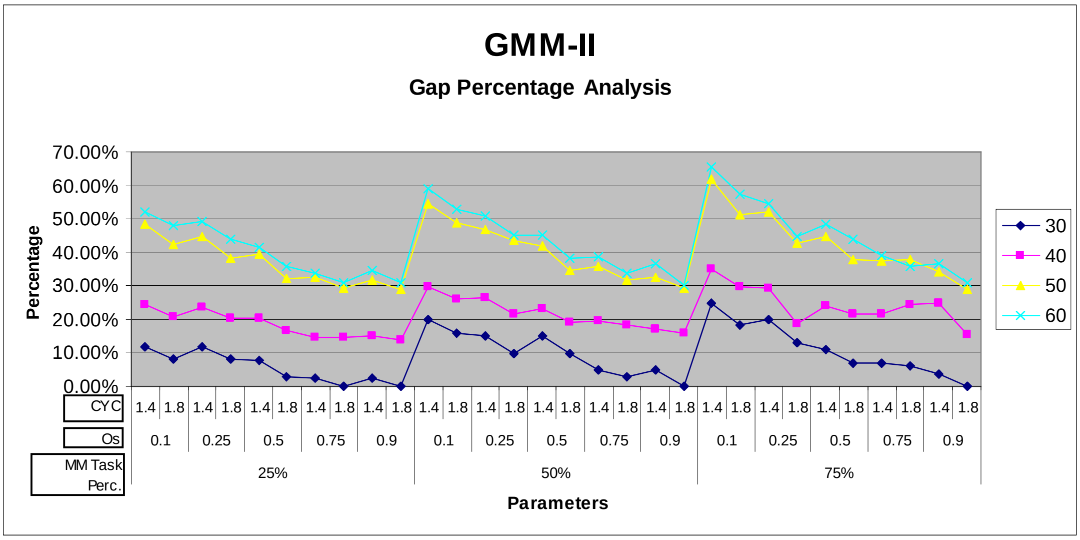

Figure 3.8: GMM-II gap percentage analysis

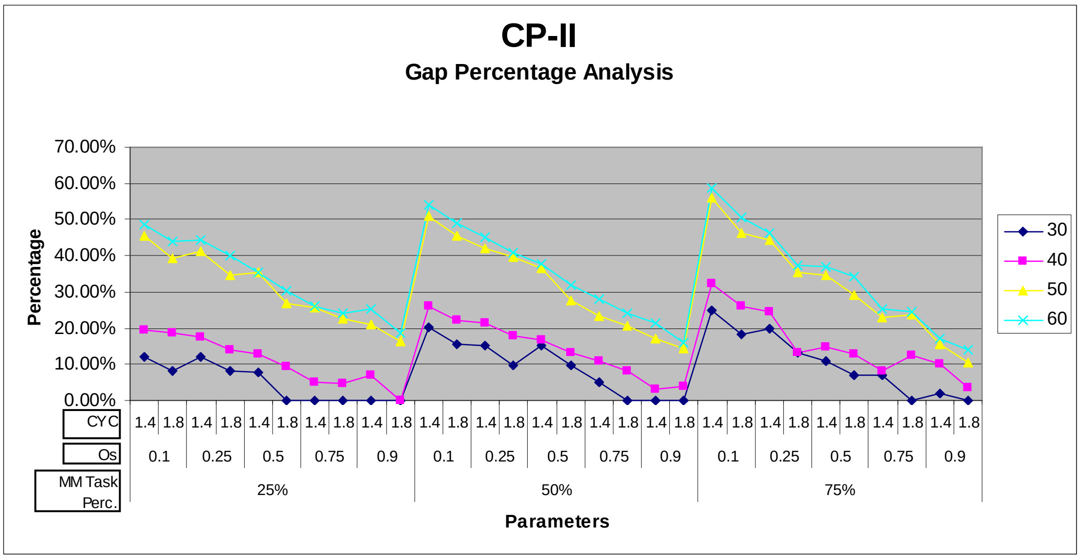

Figure 3.9: CP-II gap percentage analysis

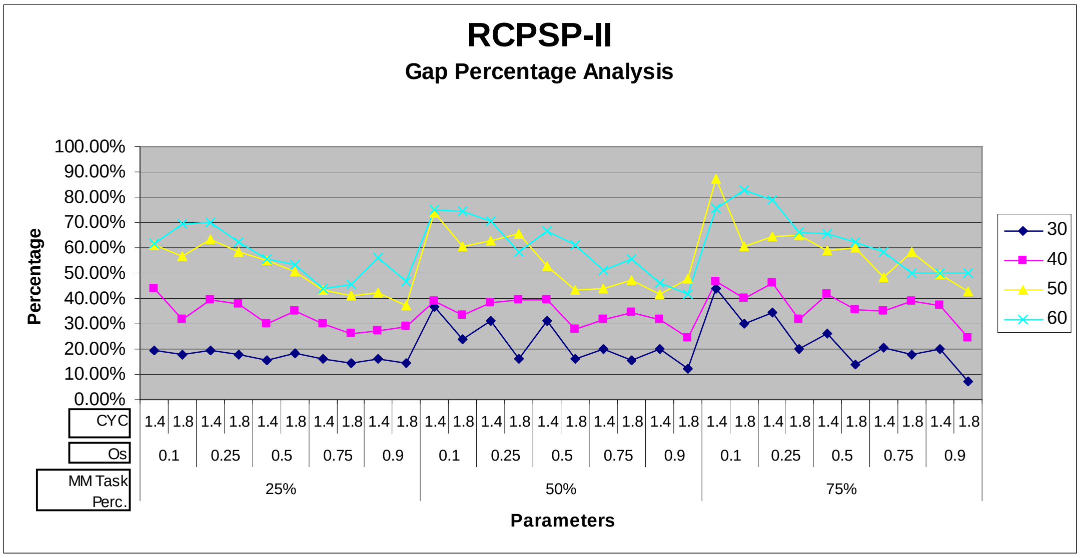

Figure 3.10: RCPSP-II gap percentage analysis

3.7.2.11 Effects of Additional Inequalities

Adding valid inequalities to RCPSP-II formulation, only gap percentages of problems with Os values 0.75 and 0.90 declined. The figure of the improvement of gap percentages for RCPSP-II is presented in Figure 3.11. The average improvement of RCPSP-II is 3%.

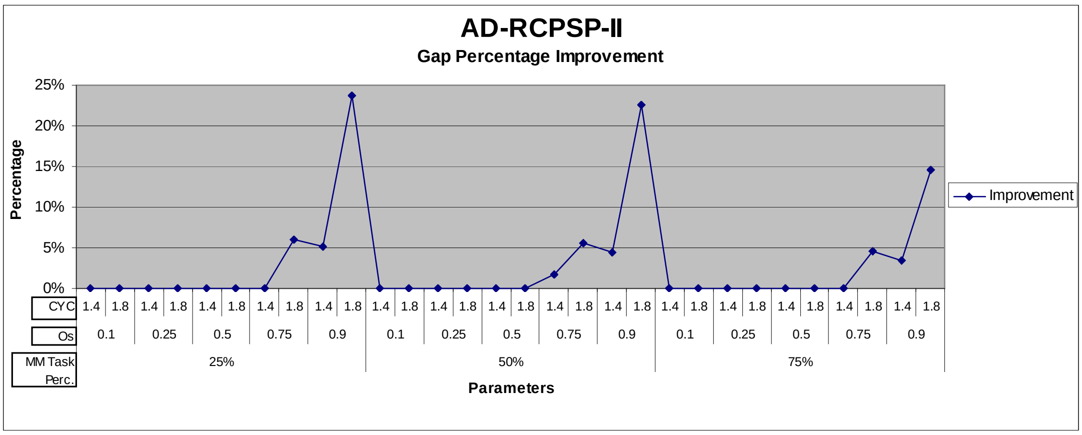

Figure 3.11: AD-RCPSP-II gap percentage improvement

The improvement on solution times when valid inequalities are added to CP-II problem is very significant. Problems with parameter sets $\{(30,40,50),\phi,\delta,\gamma\}$ can be completed by AD-CP-II less than 3 hours. The average improvement on solution times is 49%. The solution time improvement analysis is presented in Figure 3.12.

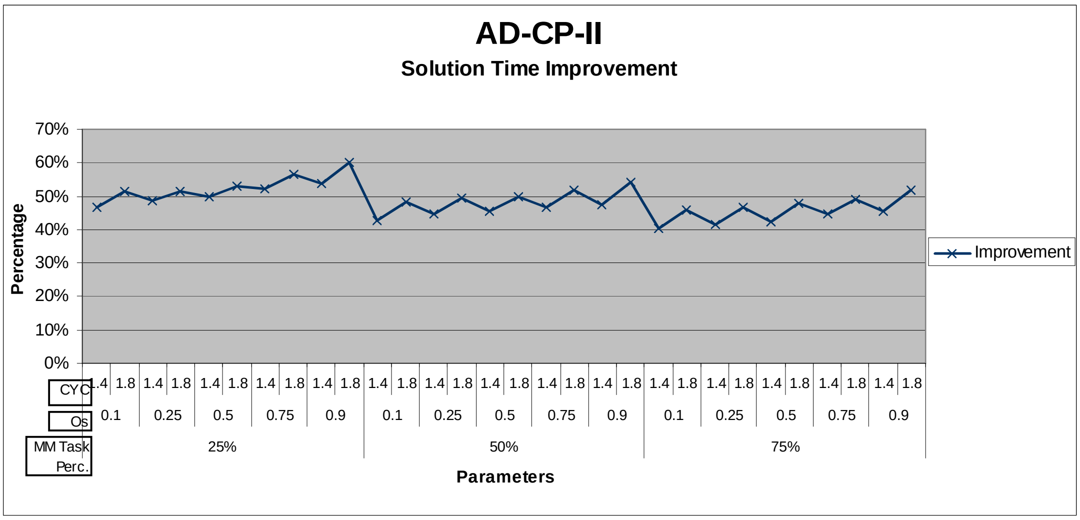

Figure 3.12: AD-CP-II solution time improvement

For Os values greater than 0.75, the improvement percentage is greater than or equal to 50%.

CHAPTER 4

HEURISTIC APPROACH

In Chapter 3, formulations for *assembly line balancing with multi-manned tasks* problem are presented for both Type-I and Type-II. Two mathematical models and one constraint programming model are developed and presented. In addition to these, valid inequalities are introduced to models RCPSP-I,II and CP-I,II. However, after conducting experimental runs, solution times are not acceptable for problems with large number of tasks and low Os values.

For Type-I case, the best formulation, AD-CP-I, could find optimum solutions for problems with number of tasks less than 50, but solution times are very long for problems with low Os values. For Type-II case, the best formulation, AD-CP-II, could not find optimum solutions in 3 hours for problems with task number greater than 50. Indeed gap percentages are not acceptable for problems with low Os values and large number of tasks. Hence heuristic approaches for both types have to be developed which will solve problems with high number of tasks and low Os values in reasonable times or will have acceptable gap percentages.

In this chapter, heuristic approaches for both types are first presented. Then outputs of these algorithms are used as a lower bound for GMM-I, II formulations. These generated four formulations are applied to the case study given in section 3.7.1. Finally, the same experiment given in 3.8.1. is conducted for these formulations and performance and comparison of all formulations are presented.

4.1 Heuristic Approach For Type-I SALB Problems

This approach is a three step approach which gives feasible solution but may not be optimum.

In the first step a mathematical model is solved as a bin-packing problem with respect to precedence constraints. Tasks are assigned to stations (not to operators) but duplicate pairs of multi-manned tasks are not used. Indeed task times of multi-manned tasks are defined as doubled. Mathematical model of the first step is as follows;

4.1.1 First Step of Heuristic-I *(Relax Model)*

Objective function is again (3.1) and also constraint (3.12) is again in the constraint set. New constraints are as follows;

$$\sum_k X_i^k = 1 \qquad \forall i \in SET \qquad \textbf{(4. 1)}$$

$$\sum_k \{k * X_j^k - k * X_i^k\} \geq 0 \qquad \forall i, j \in Pred(i, j) \qquad \textbf{(4. 2)}$$

$$\left[\sum_i (X_i^k * P_i)\right] + \left[\sum_j (X_j^k * P_j)\right] * 2 - (CYC * W_u^k) \leq 0$$

$$\forall i \in SST, \forall j \in SMT, \forall k \in K, \forall u \in \{1,2\} \qquad \textbf{(4. 3)}$$

$$2 * X_j^k - \left(\sum_u W_u^k\right) \leq 0 \qquad \forall j \in SMT, \forall k \in K, \forall u \in \{1,2\} \qquad \textbf{(4. 4)}$$

$$X_i^k, W_u^k, V^k \in (0,1) \qquad \forall i \in SET, \forall k \in K, \forall u \in \{1,2\} \qquad \textbf{(4. 5)}$$

Objective function is again (3.1). Constraints (4. 1) and (4.2) are like constraints (3.2) and (3.3) respectively except "u" indices are removed. Constraint (4.3) also resembles constraint (3.4) but this time set of all tasks (SEAT) are separated to two task sets which are set of single tasks (SST) and set of multi-manned tasks (SMT). Task times of multi-manned tasks are doubled because duplicate pairs are not included in this formulation. Constraint (4.4) is the only new added constraint. This constraint satisfies that if a multi-manned task is assigned to a station, then at least two operators must be assigned to that station. In *general mathematical model*, it is not needed to use this constraint because constraint sets (3.2), (3.5) and (3.6) are sufficient to satisfy this situation. Constraint (3.12) and (3.18) are used again but differently in constraint (4.5) set of tasks (SET) is used instead of set of all tasks (SEAT) since duplicate pairs of multi-manned tasks are not defined in this formulation.

This formulation does not consider the precedence relations on the stations. Hence the solution may not be a feasible solution. The solution data is an input for the second step of the approach.

In the next step, scheduling of stations considering precedence relations are found station by station separately. The model is solved for "$\sum_k V^k$ times" found in the first step. The mathematical model for any "k'th station" is as follows;

4.1.2 Second Step of Heuristic-I *(Station Scheduling)*

Minimize MKS$_k$ **(4.6)**

Subject to

$$\sum_u X_{u,i} = 1 \qquad \forall i \in SAT_k, \forall u \in \{1,2\} \qquad \textbf{(3.14)}$$

$$X_{u,i} + X_{u,j} \leq 1 \qquad \forall i, j \in Duplicate(i,j), \forall u \in \{1,2\} \qquad \textbf{(3.15)}$$

$$\sum_u St_{u,j} - \left\{ \left(P_j * \sum_u X_{u,i} \right) + \sum_u St_{u,i} \right\} \geq 0 \qquad \forall i, j \in Pred(i,j), \forall u \in \{1,2\} \qquad \textbf{(3.16)}$$

$$MKS_k - \left\{ \sum_u St_{u,i} + \{P_i * \sum_u X_{u,i}\} \right\} \geq 0 \qquad \forall i \in SAT_k, \forall u \in \{1,2\} \qquad \textbf{(4.7)}$$

$$\sum_u St_{u,i} - \sum_u St_{u,j} = 0 \qquad \forall i, j \in Duplicate(i,j), \forall u \in \{1,2\} \qquad \textbf{(3.17)}$$

$$\left.\begin{aligned} & St_{u,i} - St_{u,j} - \{P_j * X_{u,j}\} + T * (1 - Y_{u,j,i}) \geq 0 \\ & St_{u,j} - St_{u,i} - \{P_i * X_{u,i}\} + T * (Y_{u,j,i}) \geq 0 \end{aligned}\right\} \quad \forall i, j \in Discon(i, j), \forall u \in \{1,2\} \qquad \textbf{(3.18)}$$

$$St_{u,i} - (T * X_{u,i}) \leq 0 \qquad \forall i \in SAT_k, \forall u \in \{1,2\} \qquad \textbf{(3.19)}$$

$$\left.\begin{aligned} & X_{u,i}, Y_{u,i,j} \in (0,1) \\ & St_{u,i}, MKS_k \geq 0 \end{aligned}\right\} \quad \forall i \in SEAT, \forall k \in K, \forall u \in \{1,2\} \qquad \textbf{(4.8)}$$

Objective function is to minimize the makespan of the station. Actually this model resembles the model presented in section 3.2.2. In this model only station breaks are removed. Hence constraint (4.7) *MKS_k is used instead of CYC.*

After solving this model, MKS_k's are found for all stations. Surely some MKS_k's are greater than CYC. Hence a method is proposed to reduce these MKS_k's to a value less than or equal to CYC. This method is the *third step of the heuristic.*

4.1.3 Third Step of Heuristic-I *(Finalization)*

This is the final step of the *heuristic approach*. The aim of this method is to reduce the value of MKS_k's, which are greater than CYC, to a value less than or equal to CYC. But operators to which tasks are assigned are not changed. In other words, if task "4" is assigned to any operator "2" $(u \in \{1,2\})$, this task remains assigned to operator "2" through the heuristic procedure. The solution found in this step is the final solution, and it will be feasible but may not be optimum.

The third step starts with the first station. First it checks whether there are tasks whose start times are greater than $(CYC - P_i)$ for both operators. Unless there is any task, the procedure continues with the second station. But if there are tasks whose start times are greater than $(CYC - P_i)$, these tasks are re-located to the start of second station, in the same order and same assigned operators. After re-locating these tasks, start times of tasks which are already in second station at the beginning of the third step are increased by total task times of tasks which are newly added to the second station. At this time, a new set of tasks assigned to second station is derived. Next the procedure continues with checking whether there are tasks whose start times are greater than $(CYC - P_i)$ for both operators until there are no tasks exceeding $(CYC - P_i)$. To illustrate the three steps, the procedure is proposed to the same example problem given in section 3.5.1.

In the first step a mathematical model is solved with a given cycle time of **<u>60</u>** minutes. The solution of the first step is given in Figure 4.1.

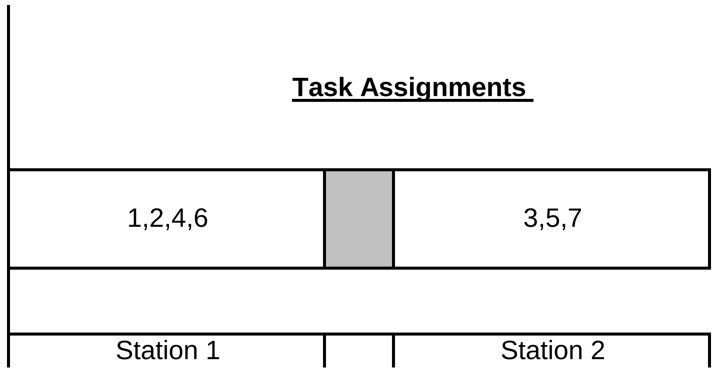

Figure 4.1: Task assignment for two stations

Next for the second step, models are solved to schedule stations for two stations. The solutions for two stations are presented in Figures 4.2 and 4.3 respectively.

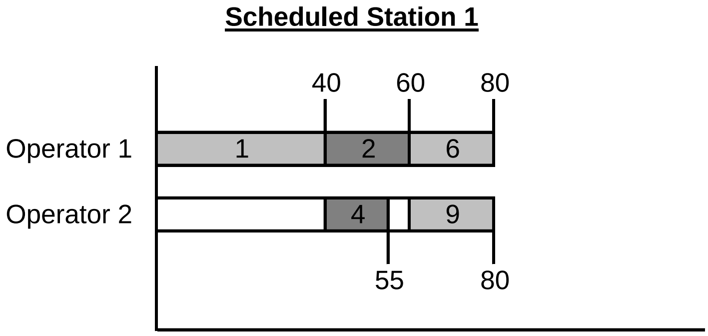

Figure 4.2: Scheduled “station 1” after second step

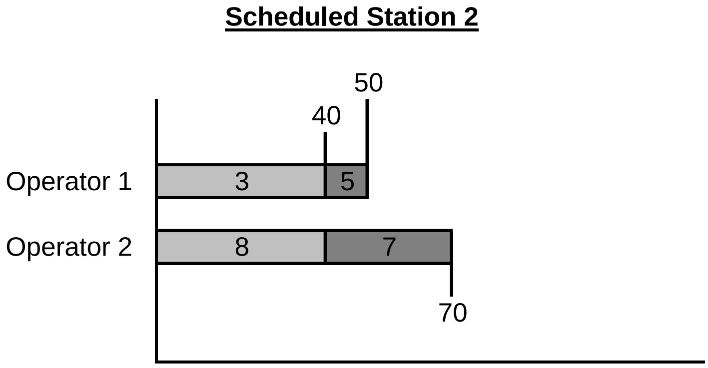

Figure 4.3: Scheduled “station 2” after second step

After obtaining scheduled task assignments, $MKS_1 = 80$ and $MKS_2 = 70$ are obtained. Next third step of the heuristic is applied. The third step is started from the first station. Only start time of “task 6” is greater than $(CYC - P_i)$, in other words $60 \geq (60 - 20)$. Then this task is re-located at station 2. The new task assignments for two stations are presented in Figures 4.4 and 4.5 respectively.

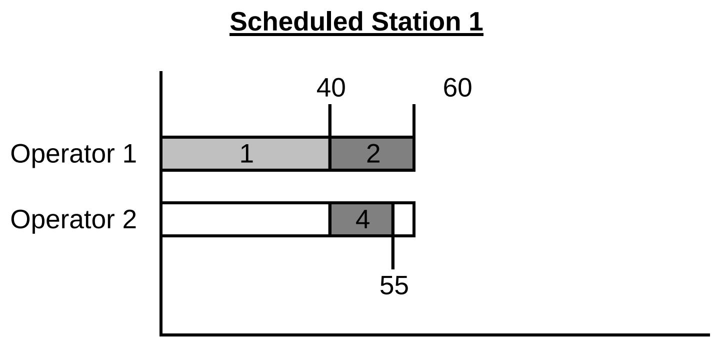

Figure 4.4: Scheduled “station 1” after third step

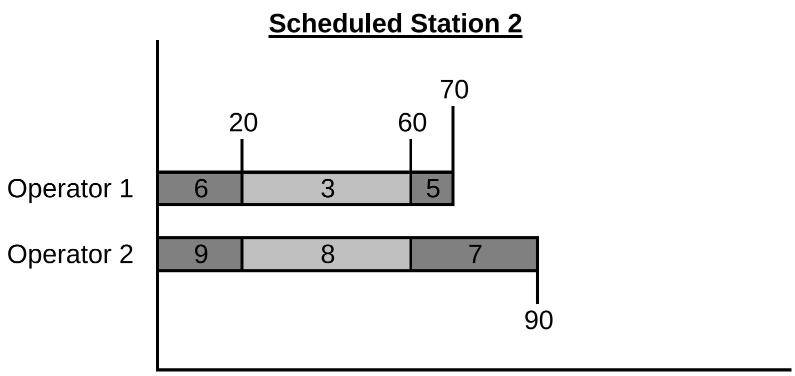

Figure 4.5: Scheduled “station 2” on third step

Now for station 2, “task 5” and “task 7” have to be re-located to next station. The new task assignments for two stations are presented in Figures 4.6 and 4.7 respectively.

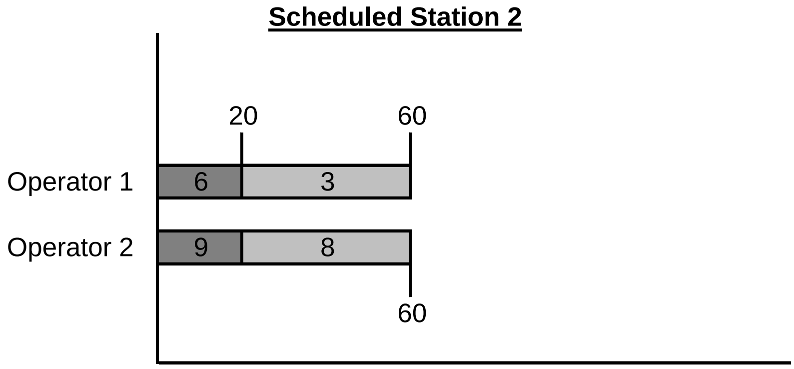

Figure 4.6: Scheduled “station 2” after third step

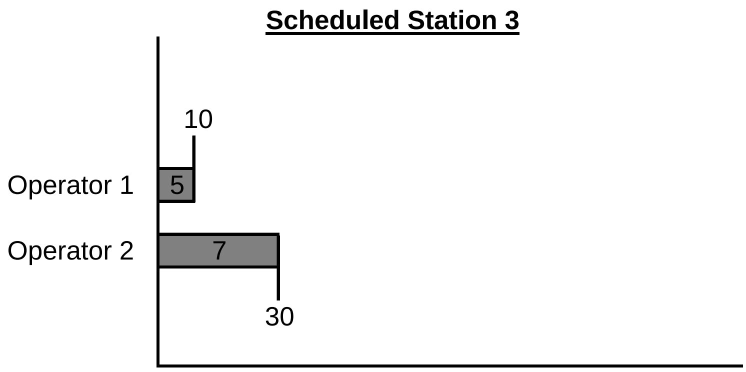

Figure 4.7: Scheduled “station 3” after third step

There is no task whose start time is greater than $(CYC - P_i)$. Therefore the heuristic is completed. First step of the approach opens two stations, but after third step, three stations are opened. The flowchart of the third step is presented in Figure 4.8.

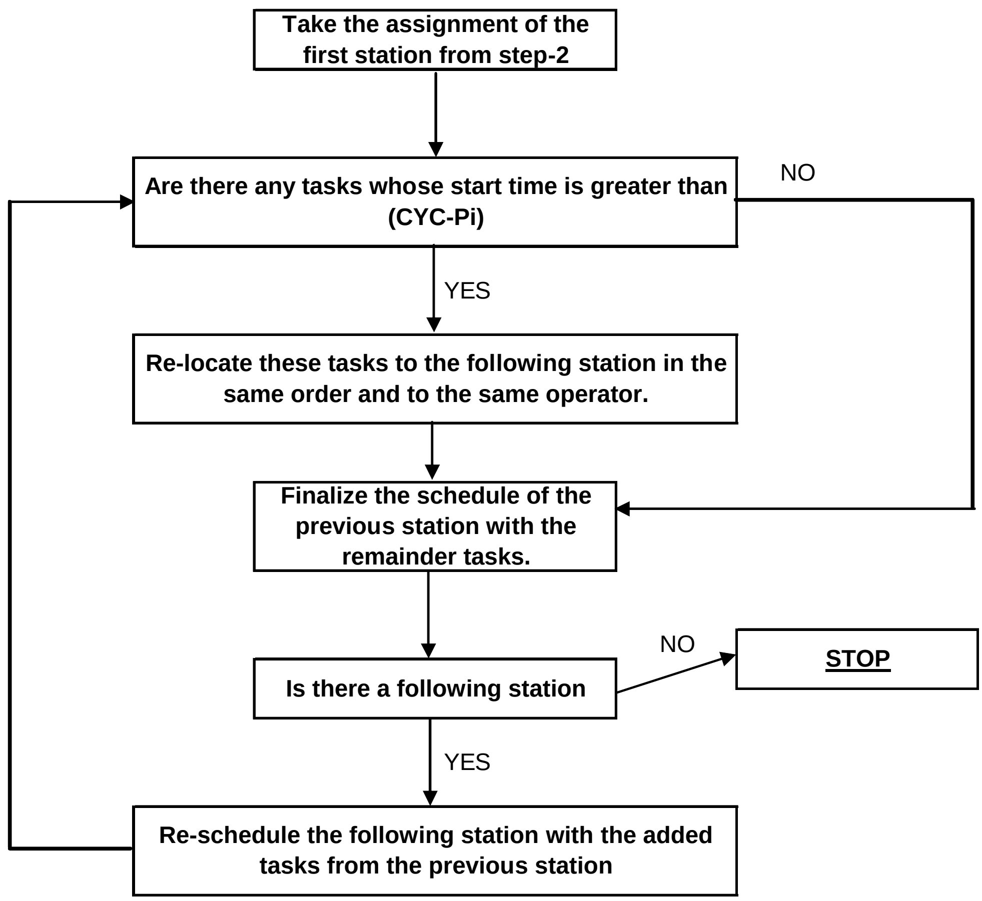

Figure 4.8: The flowchart of the third step of heuristic approach

4.2 Heuristic Approach For Type-II SALB Problems

This algorithm is two-step heuristic approach. Like the algorithm for Type-I problems, this method also may or may not find optimum but feasible solutions. The procedure is composed of two steps. In the first step, a model is developed which is almost the same model given in section 4.1.1.

4.2.1 First Step of Heuristic-II *(Relax Model)*

The model is again a bin-packing problem with respect to precedence constraints. The objective function is;

Minimize CYC **(3.23)**

Constraints (4.1), (4.2), (4.5) are in the constraint set. Constraint (4.3) is changed and W_u^k is removed. New constraint is;

$$\left[\sum_i (X_i^k * P_i)\right] + \left[\sum_j (X_j^k * P_j)\right] * 2 - (CYC) \le 0$$

$$\forall i \in SST, \forall j \in SMT, \forall k \in K \qquad \textbf{(4.9)}$$

The solution of this model may not be a feasible solution. Task assignments generated from this formulation is an input for the second step of the relax model.

4.2.2 Second Step of Heuristic-II *(Finalization)*

This step is actually same with the step presented in the section 4.1.2. The same model is solved for all stations and MKS_k's and task assignments are found for all stations. The solutions of this step is the final solution of line. The maximum MKS_k is the cycle time of the line. Also the minimum MKS_k is used as a lower bound for the general mathematical model presented in the section 3.3.1. To illustrate heuristic, consider example 1. Differently, cycle time is a variable but number of stations is given as **2**.

The solution of the first step is given in Figure 4.9 which is the same given in Figure 4.1.

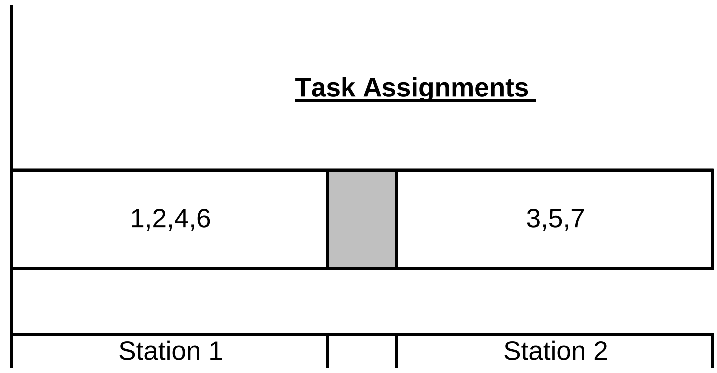

Figure 4.9: Task assignment for two stations after first step

Then the model in the second step is solved for both two stations separately. Task assignments for both stations are presented in Figures 4.10 and 4.11 respectively.

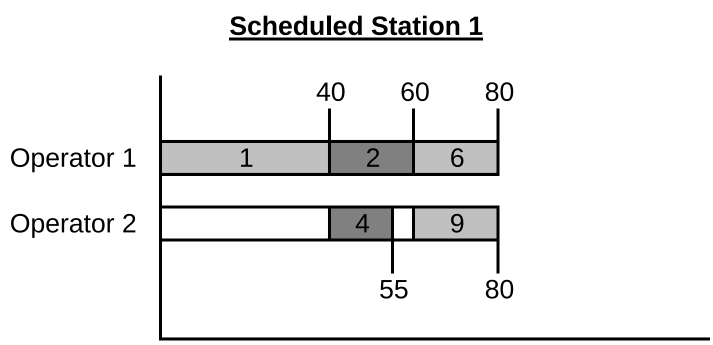

Figure 4.10: Scheduled "station 1" after second step

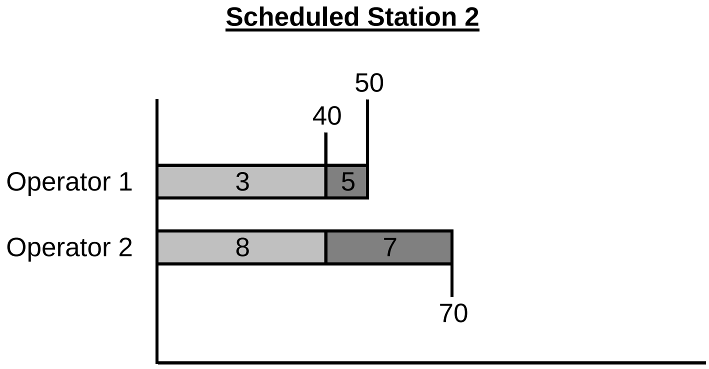

Figure 4.11: Scheduled "station 2" after second step

The procedure is completed after the second step. $MKS_1 = 80$ and $MKS_2 = 70$ are found, so cycle time of this line is 80 minutes.

4.3 Proposed Inequalities Using Heuristic Output

4.3.1 Additional Inequality for GMM-I

There is only one valid inequality constraint for this model. A lower bound is introduced for the objective function value. The objective function value of the *first step of heuristic-I* has to be lower than the objective function value of general mathematical model. The additional constraint is as follows:

$$\sum_k k * V^k \geq Solution\_found\_in\_4.1.1 \qquad \textbf{(4.10)}$$

Consider example 1 again. The solution found after first step is "2". If we try to solve the same problem with *general mathematical model*, the additional constraint will be;

$$\sum_k k * V^k \geq 2$$

4.3.2 Additional Inequality for GMM-II

Like Type-I problem, there is only one valid inequality constraint for this model. A lower bound is introduced for the objective function value. In section 4.2.2, at the end of *second step of heuristic of Type-I*, MKS_k values are found for all "k" stations and the minimum MKS is a lower bound for cycle time. The additional constraint is as follows:

$$CYC \geq \underset{k}{Min}\left[MKS_k\right] \qquad \forall k \in K \qquad \textbf{(4.11)}$$

Consider example 1. After step 2, $MKS_1 = 70$ and $MKS_1 = 80$. So the additional constraint will be;

$$CYC \geq 70$$

4.4 Brief Summary of All Formulations

In Chapter 3, ten formulations are presented. In this chapter, two heuristic algorithms and additional inequalities for GMM-I and GMM-II are given. Therefore in total, seven formulations for Type-I and seven formulations for Type-II are presented. Abbreviations, objective functions and characteristics of all formulations given in this chapter are presented in Table 4.1.

Table 4-1: Brief summary and abbreviations for four formulations

	Name of Formulation	Section	Abbreviation	Solution	Other
TYPE-I	Heuristic-I	4.1	RM-I	Feasible	Feasible solution found in 3 steps
TYPE-II	Heuristic-II	4.2	RM-II	Feasible	Feasible solution found in 2 steps
TYPE-I	Additional Inequality for General Mathematical Model-I	4.3.1	AD-GMM-I	Optimal	Constraint (4.10) is added
TYPE-II	Additional Inequality for General Mathematical Model-II	4.3.2	AD-GMM-II	Optimal	Constraint (4.11) is added

RM-I finds feasible (and may be optimal) in three steps, RM-II finds feasible (and may be optimal) in two steps.

AD-GMM-I is obtained by adding constraint (4.10) to the model GMM-I. Constraint (4.11) is inserted to the model GMM-II to derive AD-GMM-II.

In the following section, in order to illustrate all these four formulations, these are applied to the case study given in section 3.7.1 and test problems given in section 3.7.2.

4.5 Test of Formulations

4.5.1 Case Study

Case study presented in section 3.7.1 is applied for four formulations presented in this chapter. Again these formulations are solved by optimization software Cplex 10.1.0.

4.5.1.1 Type-I Solutions

RM-I Solution

In the first step of RM-I, number of stations is found as 5. This value will be used in the formulation AD-GMM-I as a lower bound. After second step, MKS_k values are found.

$$MKS_1 = 150$$
$$MKS_2 = 245$$
$$MKS_3 = 155$$
$$MKS_4 = 215$$
$$MKS_5 = 160$$

Next, third step of the heuristic is applied. Result is as follows;

$$Objective\ Function : 6$$
$$Solution\ Time(\text{sec.}) : 12$$

AD-GMM-I Solution

The solution of the first step of RM-I model is added to the GMM-I model. Added constraint is $\sum_k k * V^k \geq 15$.

$Objective\ Function: 6$
$Number\ of\ variables: 5089$
$Nodes: 325$
$Solution\ Time(\text{sec.}): 232$

4.5.1.2 Type-II Solutions

RM-II Solution

After second step of RM-II, MKS_k values are found.

$MKS_1 = 150$
$MKS_2 = 200$
$MKS_3 = 280$
$MKS_4 = 190$
$MKS_5 = 185$

The maximum MKS_k is the *cycle time* of the line 280. Also the minimum MKS_k will be used in the formulation AD-GMM-II as a lower bound.

$Objective\ Function: 280$
$Solution\ Time(\text{sec.}): 894$

AD-GMM-II Solution

The minimum MKS_k found in the second step of RM-II is added to the model GMM-II. Added constraint is $CYC \geq 150$

$Objective\ Function: 165$
$Number\ of\ variables: 3621$
$Nodes: 3205$
$Solution\ Time(\text{sec.}): 3\ hour$ timeout

4.5.1.3 Summary of Solutions and Results

The results of four formulations for Type-I and Type-II are presented in Table 4.2 and 4.3 respectively.

Table 4-2: Results of Type-I Case Study

	TYPE-I (Given Cycle Time = 160 min.)	
Formulations	RM-I	AD-GMM-I
Number of Stations	6	6
Number of Variables		5089
Nodes / Choice Points		325
Solution Time (Sec.)	12	232
Gap Between Optimal Solution	0%	0%
Performance Gap	20%	

Table 4-3: Results of Type-II Case Study

	TYPE-II (Given Number of Stations = 5)	
Formulations	RM-II	AD-GMM-II
Cycle Time	280	165
Number of Variables		3621
Nodes / Choice Points		3205
Solution Time (Sec.)	894	3 hour timeout
Gap Between Optimal Solution	70%	0%
Performance Gap	85%	

Comparison tables for all seven formulations for Type-I and Type-II are presented in Table 4.4 and 4.5 respectively.

Table 4-4: Comparison of all formulations for Type-I

	TYPE-I (Given Cycle Time = 160 min.)						
Formulations	**GMM-I**	**RCPSP-I**	**CP-I**	**AD-RCPSP-I**	**AD-CP-I**	**RM-I**	**AD-GMM-I**
Number of Stations	6	7	6	7	6	6	6
Number of Variables	5089	1497	238	1497	238		5089
Nodes / Choice Points	3428	30611	68	22549	12		325
Solution Time (Sec.)	611	3 hour timeout	418	3 hour timeout	323	12	232
Gap Between Optimal Solution	0%	17%	0%	17%	0%	0%	0%

Table 4-5: Comparison of all formulations for Type-II

	TYPE-II (Given Number of Stations = 5)						
Formulations	**GMM-II**	**RCPSP-II**	**CP-II**	**AD-RCPSP-II**	**AD-CP-II**	**RM-II**	**AD-GMM-II**
Cycle Time	170	180	170	175	165	280	165
Number of Variables	3621	1229	239	1229	239		3621
Nodes / Choice Points	6303	850068	109835704	376902	4628041		3205
Solution Time (Sec.)	3 hour timeout	3 hour timeout	3 hour timeout	3 hour timeout	824	894	3 hour timeout
Gap Between Optimal Solution	3%	9%	3%	6%	0%	70%	0%

For Type-I case study, additional inequalities of GMM-I formulation decreases solution time from 611 seconds to 232 seconds. Besides RM-I formulation also finds the optimal solution although this formulation is a three-step heuristic approach.

For Type-II case study, AD-GMM-II find optimal solutions but solution time is not acceptable. Also RM-II's gap is very large (70%) and its solution is not acceptable for this case study. After introducing heuristic algorithms and lower bounds for GMM-I,II, still AD-CP-I and AD-CP-II are best alternative formulations for Type-I and Type-I for this case study.

4.5.2 Computational Experiments

The procedure given in section in 3.7.2 is applied again for AD-GMM-I and AD-GMM-II.

For RM-I and RM-II, set of tasks are increased. Modified parameters for RM-I and RM-II are as follows.

Set of Task (*N*): Number of tasks are selected as 30, 40, 50, 60 and 100.

Task Times (P_i): All task times are integer and uniformly distributed between 10 and 100. Different from procedure given in 3.7.2, 5 replications are done for $N = 30,40,50,60$, 1 replication is done for $N = 100$.

Moreover a new performance measure which is added for RM-I and RM-II is as follows.

Performance Gap Percentage (%): The gap between the lower bound and the solution found.

Lower Bound for Type-I = $\sum_i P_i \Big/ (2 * K)$

Lower Bound for Type-II = $\sum_i P_i \Big/ (2 * CYC)$

$$Performance\_Gap\% = \left\{ 100 * \left([Found - L.B.] \Big/ L.B. \right) \right\}$$

Consequently, 120 test problems are generated for AD-GMM-I, AD-GMM-II, 30 test problems for RM-I, RM-II with $N = 100$, 600 test problems with 5 replications of task times are generated for RM-I with $N = 30,40,50,60$, 120 test problems with 1 replication of task times are generated for RM-II with $N = 30,40,50,60$. Therefore totally, 1020 runs are made for this experiment.

4.5.3 Discussion of Experiment

4.5.3.1 Discussion of RM-I and AD-GMM-I Formulations

Average solution times, gap percentages and performance gaps are given in Table B.11 for RM-I formulation. Table B.12 represents the objective functions, solution times and gap percentages for AD-GMM-I formulation. These results are obtained with respect to parameters defined in section 4.5.2.

When the number of tasks are increased while the other parameters are fixed, it is easily seen in the Figure 4.12, average solutions times of formulation RM-I increases.

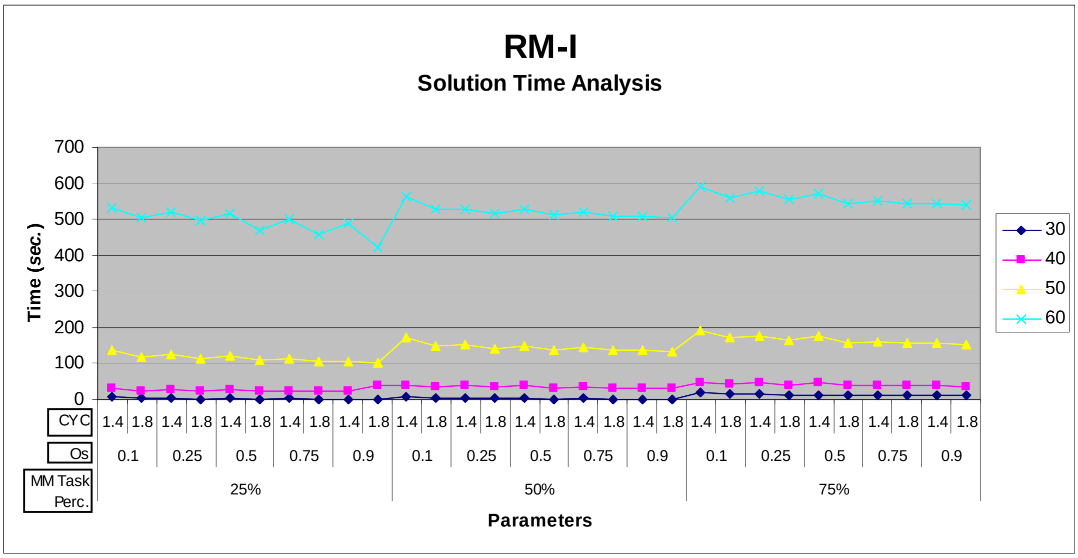

Figure 4.12: RM-I solution time analysis

If the multi-manned task percentage is increased while the other parameters are kept constant, average solutions times of RM-I increase slightly, whereas when Os value increases, this does not effect RM-I's solution times. Finally, when the cycle time multiplier increases from 1.4 to 1.8, average solution times decrease about 19%.

RM-I does not guarantee optimum solutions as mentioned in section 4.1. The gap percentage analysis of RM-I formulation is presented in Figure 4.13.

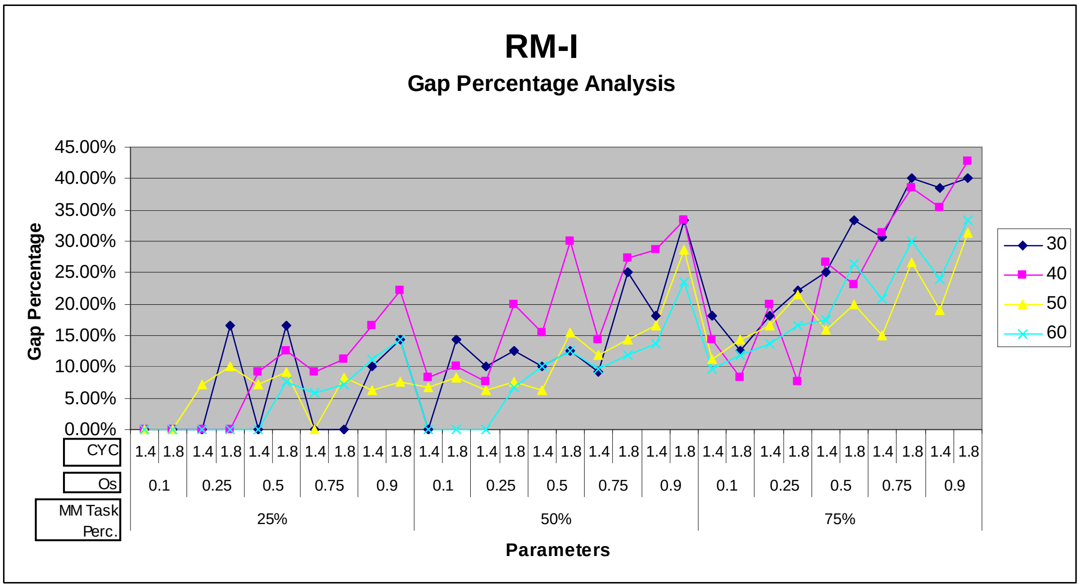

Figure 4.13: RM-I gap percentage analysis

Number of tasks actually does not effect gap percentage of RM-I formulation clearly. When Os value increases, the gap percentage increases dramatically as seen in Figure 4.13. This is because of the fact that when Os value is increased then MKS_k values are found high and so in the third step, more stations are needed to be opened. Moreover, for parameter sets $\{\alpha, 25\%, 0.1, \gamma\}$, gap percentages are 0%. If the multi-manned task percentage is increased while the other parameters are fixed, higher gap percentages are found. The average gap percentages for *multi-manned task percentage values* {25%, 50%, 75%} are 6%, 13.73, 23.03% respectively. This relation is again because of the high MKS_k values. In addition to these, when the cycle time multiplier is changed from 1.4 to 1.8, the gap percentages increases about 25% in average.

Adding valid inequalities to GMM-I formulation, solution times decreases about 59% in average. For smaller Os values and multi-manned task percentage, the improvement is high but cycle time multiplier does not effect the improvement. The graph of the improvement of solution times for GMM-I is presented in Figure 4.14.

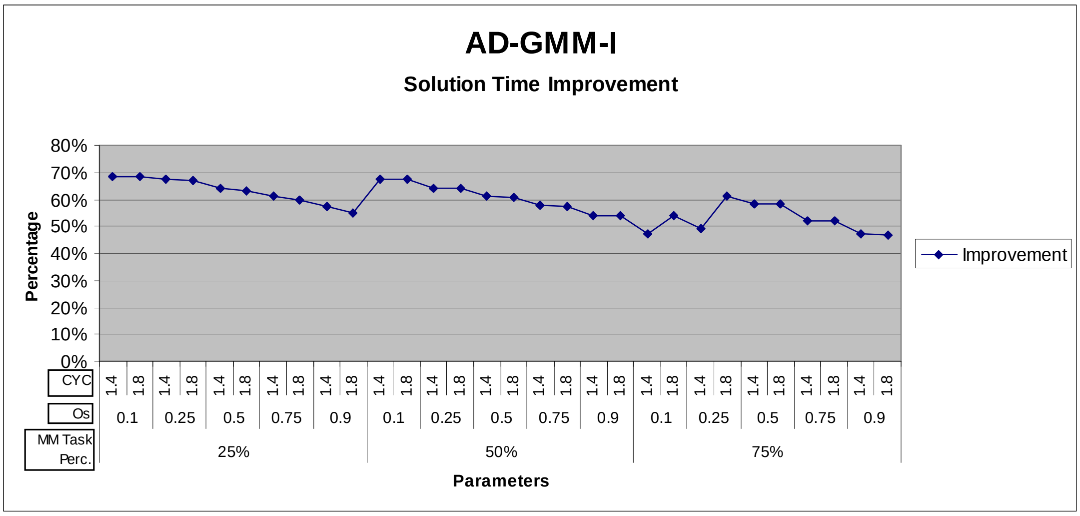

Figure 4.14: AD-GMM-I solution time improvement

AD-GMM-I could find optimum solutions for all test problems in reasonable times. But for lower multi-manned task percentage and Os value, RM-I could find optimum solutions in very short solution times.

4.5.3.2 Discussion of RM-II and AD-GMM-II Formulations

Solution times, objective functions, gap percentages and performance gaps are given in Table B.13 for RM-II formulation. Table B.14 represents the objective functions, solution times and gap percentages for AD-GMM-II formulation.

For problems with parameter sets $\{(30,40),\phi,\delta,\gamma\}$ and $\{50,0.25,\delta,\gamma\}$ RM-II can be completed less than 3 hours. The solution time analysis of RM-II is in Figure 4.15.

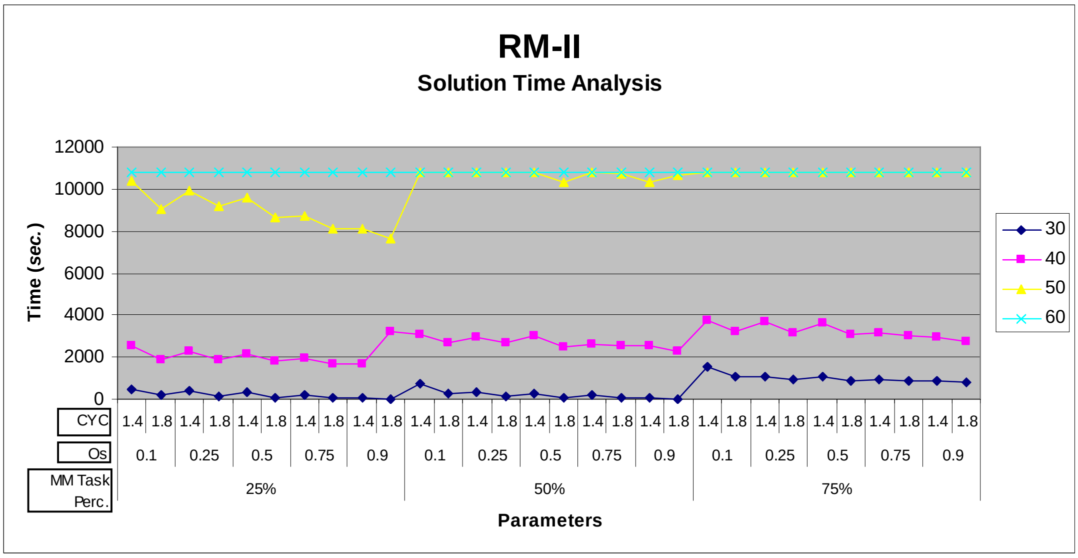

Figure 4.15: RM-II solution time analysis

When number of tasks and multi-manned task percentage increases separately, solution times are also increasing and for task numbers greater than 50, RM-II cannot be completed in 3 hours. If Os value is increased, solution times declined very slightly. Finally when number of station multiplier is changed from 1.4 to 1.8, solution times decrease about 14% in average.

Number of tasks actually does not effect the RM-II gap percentages. However, if multi-manned task percentage increases, gap percentages increase slightly. Average gap percentages are 33.42%, 37.60% and 42.39% for problems with multi-manned task percentages 25%, 50% and 75%, respectively. Moreover if Os value is increased, gap percentages increase clearly. Especially Os values greater than 0.75, gap percentages reach a very high level of approximately 50%-70%. Finally when number of station multiplier is changed from 1.4 to 1.8, solution times decrease about 7% in average. Gap percentage analysis of RM-II is given in Figure 4.16.

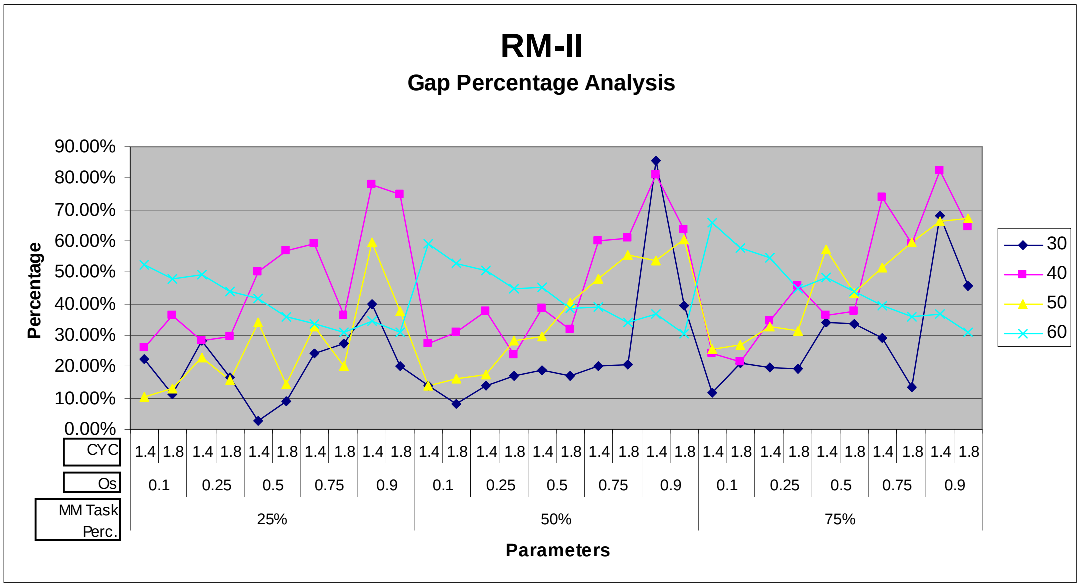

Figure 4.16: RM-II gap percentage analysis

AD-GMM-II formulation cannot be completed in 3 hours however gap percentages are improved about 49% in average. The graph of the improvement of gap percentages for GMM-II is presented in Figure 4.17.

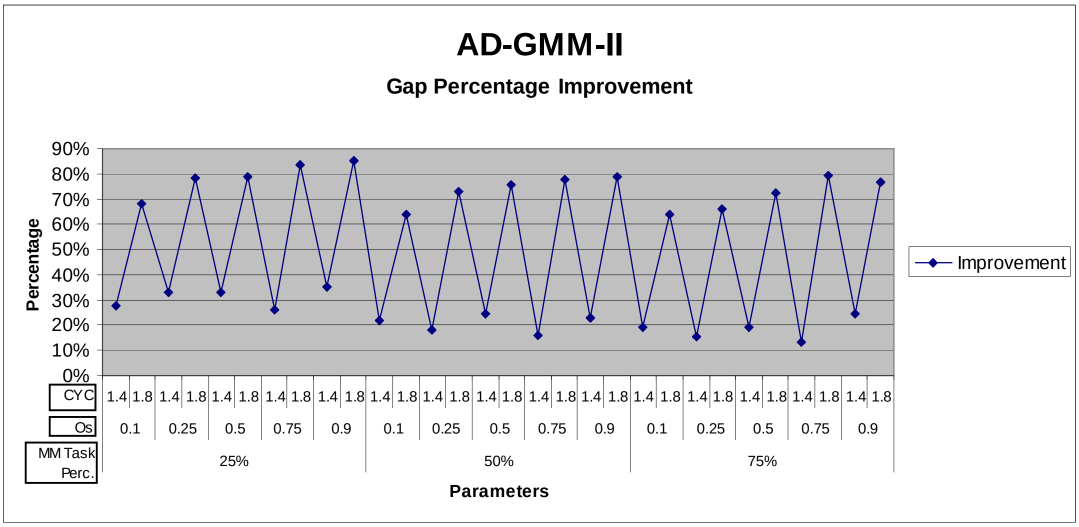

Figure 4.17: AD-GMM-II gap percentage improvement

Improvement is apparently higher for number of station multiplier value 1.8 than 1.4. This is because of the fact that the additional inequality is a lower bound for cycle time. Also improvement percentage increases for higher Os values.

AD-GMM-II could find optimum solutions in 3 hours only for problems with parameter set $\{30,25\%,\delta,\gamma\}$, $\{30,50\%,\delta,1.8\}$ and $\{30,55\%,(0.5,0.75,0.9),\gamma\}$. RM-II found solutions with parameter set $\{(30,40),\phi,\delta,\gamma\}$ and $\{50,25\%,\delta,\gamma\}$less than 3 hours but solutions of only problems with Os value less than 0.5 may be acceptable since the average gap percentage for parameters $\{(30,40),\phi,(0.1,0.25),\gamma\}$ is 23.68%.

4.6 Comparison of Type-I Formulations

Effects of parameters to performance measures are analyzed in the experiment performed in Chapter 3 and 4. Parameters affect performance measures of formulations in different ways. Comparison tables for solution times and gap percentages of Type-I formulations are given in Table B.15 and Table B.16 respectively.

AD-GMM-I could find optimum solutions for all test problems in reasonable times. But for lower multi-manned task percentage and Os value, RM-I could find optimum solutions in very short solution times. Furthermore, for large Os values (0.75 and 0.90), AD-CP-I's solution times are shorter than AD-GMM-I's solution times.

Therefore for problems which have multi-manned task percentages less than 25%, Os values less than 0.1 and 0.25, RM-I should be chosen. For problems which have Os values greater than 0.75, AD-CP-I should be chosen. For problems with other parameter sets, AD-GMM-I formulation should be chosen.

4.7 Comparison of Type-II Formulations

Comparison tables for solution times and gap percentages of Type-II formulations are given in Table B.17 and Table B.18 respectively.

AD-GMM-II could find optimum solutions in 3 hours only for problems with parameter set $\{30,25\%,\delta,\gamma\}$, $\{30,50\%,\delta,1.8\}$ and $\{30,55\%,(0.5,0.75,0.9),\gamma\}$. RM-II found solutions with parameter set $\{(30,40),\phi,\delta,\gamma\}$ and $\{50,25\%,\delta,\gamma\}$less than 3 hours but solutions of only problems with Os value less than 0.5 may be acceptable since the average gap percentage for parameters $\{(30,40),\phi,(0.1,0.25),\gamma\}$ is 23.68%.

AD-RCPSP-II could not find optimum solutions in 3 hours for any problem set and gap percentages are apparently far away than AD-GMM-II and RM-II.

The sensible formulation for Type-II problems is AD-CP-II. Problems with parameter set $\{(30,40,50),\phi,\delta,\gamma\}$can be solved in 3 hours by AD-CP-II. Moreover gap percentages for these problems are 0%. Indeed, average gap percentage for parameter set $\{60,\phi,\delta,\gamma\}$ is 3.50%.

Consequently, for small-sized problems with Os values less than 0.5, RM-II may be chosen, but for all other parameter sets of Type-II problems, AD-CP-II should be used.

CHAPTER 5

CONCLUSIONS

We develop two mathematical models and one constraint programming model for both Type-I and Type-II assembly line balancing problems with multi-manned tasks. Then valid inequalities, lower bounds are introduced to reduce the domain. Heuristic algorithms for both types are developed for large-sized problems. Totally four alternative formulations and three formulations with valid inequalities are presented for Type-I and Type-II problems.

These formulations are applied to a real case study which is an excavator assembly line. Next to see the effects of problem parameters on performance measures, computational runs are conducted. Since *assembly line balancing with multi-manned tasks* problem is firstly defined in this thesis, there does not exist any well-known test problem in the literature, therefore test problems are generated by the author of this thesis. Afterwards, formulations for both types are compared with each other and we determine acceptable models for different parameters.

For Type-I problems, heuristic algorithm should be used for few multi-manned tasks and few precedence relations. For problems with high number of precedence relations, constraint programming model with valid inequalities should be chosen. For other set of parameters, general mathematical model with a lower bound will be acceptable.

For Type-II problems, heuristic algorithm may be used for small-sized problems with low precedence relations. For other set of parameters, constraint programming model with valid inequalities is surely the most efficient formulation.

Since the problem is firstly defined in the literature, there are various future research directions. First of all, other types of objective functions should be of interest. The problem may be solved with an objective function of *minimize number of total operators*. Also the other assembly line balancing problem types such as Type-E and Type-F should be in consideration.

Secondly, only mathematical and constraint programming models are given in this thesis, other type of exact algorithms such as branch-and-cut and branch-and bound algorithms will be more efficient in terms of solution times and gap percentages.

Finally, in this thesis a heuristic algorithm including mathematical models is developed for large-sized assembly lines. However, this algorithm is not efficient for lines with more than hundred tasks. Therefore, effective heuristics especially genetic algorithm may be able to solve large-sized assembly lines. Furthermore, simulated annealing or tabu search methods should be of interest since there are various studies on assembly line balancing solutions with tabu search or simulated annealing.

REFERENCES

Akker J., Diepen G., and Hoogeveen J.A. (1991). A Column Generation Based Destructive Lower Bound for Resource Constrained Project Scheduling Problems. *1980 Mathematics Subject Classification (Revision 1991): 90B35.*

Arcus A.L. (1966). A computer method of sequencing operations for assembly lines. *International Journal of Production Research*, 4, 259-277

Askin. R.G. and Zhou. M. (1997). A parallel station heuristic for the mixed-model production line balancing problem. *International Journal of Production Research.* 35(11). 3095 3105.

Bard, J.F. (1989). Assembly line balancing with parallel workstations and dead time. *International Journal of Production Research,* 27(6). 1005-1018.

Bartholdi, J.J., (1993). Balancing two-sided assembly lines: A case study, *International Journal of Production Research* 31, 2447-2461.

Baybars, I., (1986). A survey of exact algorithms for the simple assembly line balancing problem. *Management Science* 32 (8), 909–932.

Bockmayr, A., Pisaruk, N., (2001). Solving assembly line balancing problems by combining IP and CP. In: Proceedings of the 6th Annual Workshop of the ERCIM Working Group on Constraints, Prague, Czech Republic.

Boysen N., Fliedner M, Scholl A. (2006). A classification of assembly line balancing problems. *Arbeits- und Diskussionspapiere der Wirtschaftswissenschaftlichen Fakultät der Friedrich-Schiller-Universität Jena* (12)

Bowman, E. H., (1960). Assembly Line Balancing by Linear programming, *Operations Research*, 8 (3), 385-389.

Brucker P. and Knust S. (2000). A linear programming and constraint propagation-based lower bound for the RCPSP. *European Journal of Operational Research* 127(2), 355-362

Brucker P. (2002). Scheduling and constraint propagation. *Discrete Applied Mathematics* 123(1-3), 227-256

Brucker P. and Knust S.(2003). Lower bounds for resource-constrained project scheduling problems. *European Journal of Operational Research* 149(2), 302-313

Bukchin J. and Rubinovitz J. (2003). A weighted approach for assembly line design with station paralleling and equipment selection. *IIE Transactions,* 35. 73-85.

Bukchin, Y., Rabinowitch, I., (2005).. A branch-and-bound based solution approach for the mixed-model assembly line-balancing problem for minimizing stations and task duplication costs. To appear: *European Journal of Operational Research*.

Buxey. G.M. (1974). Assembly line balancing with multiple stations. *Management Science.* 20(6). 1010-1021.

Carlier J. and Néron E. (2000). A new LP-based lower bound for the cumulative scheduling problem. *European Journal of Operational Research* 127(2), 363-382

Demassey S, Artigues C., Michelon P. (2005). Constraint-Propagation-Based Cutting Planes: An Application to the Resource-Constrained Project Scheduling Problem. *INFORMS Journal on Computing* 17(1), 52–65

De Reyck, B., Herroelen, W., (1997). Assembly line balancing by resource-constrained project scheduling—A critical appraisal. *Foundations of Computing and Control Engineering* 22, 143–167.

Ege, Y. (2001). Assembly line balancing with station paralleling M.S., *Thesis in Industrial Engineering, METU.*

Freeman D.R. and Jucker J.V., (1967). The line balancing problem. *Journal of Industrial Engineering*, 18, 361

Hooker, J. (2000). Logic-Based Methods for Optimization. *John Wiley & Sons.*

Ghosh. S. and Gagnon. R.J. (1989). A comprehensive literature review and analysis of the design, balancing and scheduling of assembly systems. *International Journal of Production Research.* 27(4), 637-670

Gökçen H, Agpak K, Benzer R., (2006). Balancing of parallel assembly lines. *International Journal of Production Economics.* 103, 600-609

Gutjhar A.L. and Nemhauser G.L.,(1964). An algorithm for line balancing. *Management Science*, 11, 308-315

Kaplan, Ö. (2004). Assembly line balancing with task paralleling, *M.S. Thesis in Industrial Engineering, METU.*

Kim, Y.K., Kim, Y., Kim, Y.J., (2000). Two-sided assembly line balancing: a genetic algorithm approach. *Production Planning and Control* 11, 44–53.

Klein, R., and Scholl, A., (1996). Maximizing the Production Rate in Simple Assembly Line Balancing-A Branch and Bound Procedure, *European Journal of Operational Research,* 62, 367-385.

Klein R. and Scholl A. (1999). Computing lower bounds by destructive improvement: An application to resource-constrained project scheduling. *European Journal of Operational Research* 112(2), 322-346 .

Lee, T.O., Kim, Y., Kim, Y.K., (2001). Two-sided assembly line balancing to maximize work relatedness and slackness. *Computers and Industrial Engineering* 40, 273–292.

Mastor. A. (1970). An experimental investigation and comparative evaluation of production line balancing techniques. *Management Science.* 16(22). 728-745.

McMullen, PR. and Frazier. G.V. (1998). Using simulated annealing to solve a multi- objective assembly line balancing problem with parallel workstations. *International Journal of Production Research.* 36{10), 2717-2741.

Pinto, P.. Dannenbring. D.G. and Khumawala. B.M. (1975). A branch and bound algorithm for assembly line balancing with paralleling. *International Journal of Production Research.* 13(2), 183-196.

Pinto. P.. Dartnenbring. D.G. and Khumawala. B.M. (1981). Branch and bound heuristic procedures for assembly line balancing with paralleling of stations. *International Journal of Production Research,* 19(4). 565-576.

Sarin S.C. and Erel E. (1998). A Survey of the Assembly line balancing procedures. *Production Planning and Control.* 9, 414-434

Sarker. B.R. and Shantikumar, J.G. (1983). Generalized approach for serial or parallel line balancing. *International Journal of Production Research.* 21(1). 109-133.

Sprecher, A., (1999). A competitive branch-and-bound algorithm for the simple assembly line balancing problem. *International Journal of Production Research* 37, 1787–1816.

Süer G. A., (1998). Designing parallel assembly lines. *Computer and Industrial Engineering* 35(3-4) 467-470.

Scholl A. Becker C. , (2006). State-of-the-art exact and heuristic solution procedures for simple assembly line balancing. *European Journal of Operational Research* 168, 666-693

Scholl A. Becker C. , (2006). A survey on problems and methods in generalized assembly line balancing. . *European Journal of Operational Research* 168, 694-715

Udomkesmalee. N. and Daganzo. C.F. (1989). Impact of parallel processing on job sequences in flexible assembly systems. *International Journal of Production Research,* 27(1), 73-89.

White, W. W., (1961). Comments on a Paper by Bowman, *Operations Research*, 9.

APPENDIX A

CONSTRAINT PROGRAMMING MODELS

A.1 Type-I Model

```
range Boolean 0..1;

int capacity = 2;

 int nbTasks = 38;

 int nbStations = 9;

 int settasks = 29;

 range Tasks 1..nbTasks;

 range Stations 1..nbStations;

 range Set 1..settasks;

int duration[Tasks] = [75, 40, 75, 90, 15, 25, 60, 40, 20, 70, 70, 100, 10, 30, 70, 40,
10, 25, 40, 10, 75, 70, 25, 10, 10, 10, 35, 30, 65, 0, 0, 0, 0, 0, 0, 0, 0, 0]

int totalDuration = sum(t in Tasks) duration[t];

 int demand[Tasks] = [2 , 1, 1, 1, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 1, 2, 1,
1, 1, 2, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 2];

 struct Precedences { int before; int after; };

 {Precedences} setOfPrecedences = {

 < 1 , 2 > ,

< 1 , 3 > ,

< 1 , 4 > ,

< 4 , 5 > ,

< 5 , 6 > ,
```

```
< 5 , 10 > ,
< 5 , 26 > ,
< 6 , 7 > ,
< 7 , 8 > ,
< 8 , 9 > ,
< 9 , 20 > ,
< 10 , 11 > ,
< 10 , 17 > ,
< 11 , 12 > ,
< 12 , 13 > ,
< 12 , 14 > ,
< 13 , 15 > ,
< 14 , 15 > ,
< 14 , 18 > ,
< 15 , 16 > ,
< 16 , 24 > ,
< 16 , 25 > ,
< 18 , 19 > ,
< 19 , 20 > ,
< 20 , 21 > ,
< 20 , 23 > ,
< 20 , 27 > ,
< 20 , 28 > ,
< 20 , 29 > ,
< 1 , 22 > };
scheduleHorizon = totalDuration;
```

```
 Activity a[t in Tasks](duration[t]);

 DiscreteResource res(2);

var Boolean open[Stations];

minimize

   sum(k in Stations) k*open[k]

 subject to {

   forall(t in Set)

     a[t] precedes a[38];

   forall(p in setOfPrecedences)

     a[p.before] precedes a[p.after];

   forall(t in Tasks)

     a[t] requires(demand[t]) res;

forall(t in Set)

      forall(k in Stations)

      2*totalDuration*open[k]-a[t].start+(k-1)*160 >= 0;

};
```

A.2 Type-II Model

```
int capacity = 2;

 int nbTasks = 33;

 range Tasks 1..nbTasks;

 int duration[Tasks] = [75, 40, 75, 90, 15, 25, 60, 40, 20, 70, 70, 100, 10, 30, 70, 40,
10, 25, 40, 10, 75, 70, 25, 10, 10, 10, 35, 30, 65, 0, 0, 0, 0];

int totalDuration = sum(t in Tasks) duration[t];

 int demand[Tasks] = [2 , 1, 1, 1, 2, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 1, 2, 1,
1, 1, 2, 1, 1, 2, 2, 2, 2];

 struct Precedences { int before; int after; };
```

```
{Precedences} setOfPrecedences = {
< 1 , 2 > ,
< 1 , 3 > ,
< 1 , 4 > ,
< 4 , 5 > ,
< 5 , 6 > ,
< 5 , 10 > ,
< 5 , 26 > ,
< 6 , 7 > ,
< 7 , 8 > ,
< 8 , 9 > ,
< 9 , 20 > ,
< 10 , 11 > ,
< 10 , 17 > ,
< 11 , 12 > ,
< 12 , 13 > ,
< 12 , 14 > ,
< 13 , 15 > ,
< 14 , 15 > ,
< 14 , 18 > ,
< 15 , 16 > ,
< 16 , 24 > ,
< 16 , 25 > ,
< 18 , 19 > ,
< 19 , 20 > ,
< 20 , 21 > ,
```

```
<  20  ,  23  >  ,
<  20  ,  27  >  ,
<  20  ,  28  >  ,
<  20  ,  29  >  ,
<  1  ,  22  >};
 scheduleHorizon = totalDuration;
 Activity a[t in Tasks](duration[t]);
 DiscreteResource res(2);
 Activity makespan(0);
 var
   int+   CYC in 0..200;
 minimize
   CYC
 subject to {
   forall(t in Tasks)
     a[t] precedes makespan;
   forall(p in setOfPrecedences)
     a[p.before] precedes a[p.after];
   forall(t in Tasks)
     a[t] requires(demand[t]) res;
     a[30].start - CYC = 0;
     a[31].start - ((2*CYC)+1) = 0;
     a[32].start - ((3*CYC)+2) = 0;
     a[33].start - ((4*C)+3) = 0;
     makespan.start - ((5*CYC)+4) = 0;
};
```

APPENDIX B

RESULTS OF ALL FORMULATIONS

In tables from B.1 to B.5, computational experiment results of formulations GMM-I, RCPSP-I, CP-I, AD-RCPSP-I and AD-CP-I, in tables from B.6 to B.10, computational experiment results of formulations GMM-II, RCPSP-II, CP-II, AD-RCPSP-II and AD-CP-II for 120 test problems are presented and in tables B. 11 to B.14, computational experiment results of formulations RM-I, AD-GMM-I, RM-II and AD-GMM-II are presented respectively.

Comparison of solution times and gap percentages of Type-I and Type-II problems are given in Table B.15,B.16, B.17 and B.18.

Table B. 1: Results of GMM-I Formulation

GMM-I						
Set of Tasks (N)	**Multi-manned Task Perc.(%)**	**Order Strength**	**Cycle Time Multiplier**	**Number of Stations Opened**	**Solution Time (*sec.*)**	**Gap**
30	25%	0,1	1,4	8	1212	0%
			1,8	6	425	0%
		0,25	1,4	8	884	0%
			1,8	6	356	0%
		0,5	1,4	8	672	0%
			1,8	6	224	0%
		0,75	1,4	9	450	0%
			1,8	7	124	0%
		0,9	1,4	10	227	0%
			1,8	7	56	0%
	50%	0,1	1,4	9	1985	0%
			1,8	7	949	0%
		0,25	1,4	10	1497	0%
			1,8	8	688	0%
		0,5	1,4	10	1218	0%
			1,8	8	455	0%
		0,75	1,4	11	957	0%
			1,8	8	321	0%
		0,9	1,4	11	647	0%
			1,8	9	288	0%
	75%	0,1	1,4	11	3649	0%
			1,8	8	2985	0%
		0,25	1,4	11	2672	0%
			1,8	9	1979	0%
		0,5	1,4	12	2064	0%
			1,8	9	991	0%
		0,75	1,4	13	1659	0%
			1,8	10	1092	0%
		0,9	1,4	13	1264	0%
			1,8	10	569	0%
40	25%	0,1	1,4	10	2260	0%
			1,8	8	864	0%
		0,25	1,4	11	1752	0%
			1,8	8	680	0%
		0,5	1,4	11	1224	0%
			1,8	8	402	0%
		0,75	1,4	11	698	0%
			1,8	9	203	0%
		0,9	1,4	12	381	0%
			1,8	9	92	0%
	50%	0,1	1,4	12	3386	0%
			1,8	10	1465	0%
		0,25	1,4	13	2498	0%
			1,8	10	1231	0%
		0,5	1,4	13	2089	0%
			1,8	10	778	0%
		0,75	1,4	14	1636	0%
			1,8	11	546	0%
		0,9	1,4	14	1099	0%
			1,8	12	479	0%
	75%	0,1	1,4	14	6289	0%
			1,8	12	5075	0%
		0,25	1,4	15	4689	0%
			1,8	13	3318	0%
		0,5	1,4	15	3462	0%
			1,8	13	1696	0%
		0,75	1,4	16	2951	0%
			1,8	13	1873	0%
		0,9	1,4	17	2109	0%
			1,8	14	989	0%

(Table B.1 Continued)

GMM-I						
Set of Tasks (N)	**Multi-manned Task Perc.(%)**	**Order Strength**	**Cycle Time Multiplier**	**Number of Stations Opened**	**Solution Time (*sec.*)**	**Gap**
50	25%	0,1	1,4	13	3519	0%
			1,8	10	1228	0%
		0,25	1,4	14	2601	0%
			1,8	10	1046	0%
		0,5	1,4	14	2001	0%
			1,8	11	658	0%
		0,75	1,4	15	1305	0%
			1,8	12	364	0%
		0,9	1,4	16	658	0%
			1,8	13	163	0%
	50%	0,1	1,4	15	5898	0%
			1,8	12	2753	0%
		0,25	1,4	16	4351	0%
			1,8	13	1898	0%
		0,5	1,4	16	3564	0%
			1,8	13	1324	0%
		0,75	1,4	17	2779	0%
			1,8	14	931	0%
		0,9	1,4	18	1903	0%
			1,8	14	842	0%
	75%	0,1	1,4	18	Timeout (3 hr)	0%
			1,8	14	8679	0%
		0,25	1,4	18	7845	0%
			1,8	14	5861	0%
		0,5	1,4	19	6001	0%
			1,8	15	2923	0%
		0,75	1,4	20	4795	0%
			1,8	15	3280	0%
		0,9	1,4	21	3667	0%
			1,8	16	1672	0%
60	25%	0,1	1,4	15	5268	0%
			1,8	12	1934	0%
		0,25	1,4	16	3821	0%
			1,8	13	1549	0%
		0,5	1,4	16	2994	0%
			1,8	13	973	0%
		0,75	1,4	17	1894	0%
			1,8	14	521	0%
		0,9	1,4	18	958	0%
			1,8	14	251	0%
	50%	0,1	1,4	18	8638	0%
			1,8	14	4092	0%
		0,25	1,4	19	6509	0%
			1,8	15	3007	0%
		0,5	1,4	20	5431	0%
			1,8	16	2109	0%
		0,75	1,4	21	4067	0%
			1,8	17	1289	0%
		0,9	1,4	22	2779	0%
			1,8	17	1219	0%
	75%	0,1	1,4	21	Timeout (3 hr)	0%
			1,8	17	Timeout (3 hr)	0%
		0,25	1,4	22	Timeout (3 hr)	0%
			1,8	18	8601	0%
		0,5	1,4	23	8875	0%
			1,8	19	4231	0%
		0,75	1,4	24	7209	0%
			1,8	20	4658	0%
		0,9	1,4	25	5391	0%
			1,8	21	2551	0%

Table B. 2: Results of RCPSP-I Formulation

RCPSP-I						
Set of Tasks (N)	Multi-manned Task Perc.(%)	Order Strength	Cycle Time Multiplier	Number of Stations Opened	Solution Time (*sec.*)	Gap
30	25%	0,1	1,4	15	Timeout (3 hr)	87,50%
			1,8	9	Timeout (3 hr)	50,00%
		0,25	1,4	13	Timeout (3 hr)	62,50%
			1,8	7	Timeout (3 hr)	16,67%
		0,5	1,4	12	Timeout (3 hr)	50,00%
			1,8	7	Timeout (3 hr)	16,67%
		0,75	1,4	12	Timeout (3 hr)	33,33%
			1,8	7	Timeout (3 hr)	0,00%
		0,9	1,4	12	Timeout (3 hr)	20,00%
			1,8	7	Timeout (3 hr)	0,00%
	50%	0,1	1,4	17	Timeout (3 hr)	88,89%
			1,8	10	Timeout (3 hr)	42,86%
		0,25	1,4	17	Timeout (3 hr)	70,00%
			1,8	10	Timeout (3 hr)	25,00%
		0,5	1,4	15	Timeout (3 hr)	50,00%
			1,8	9	Timeout (3 hr)	12,50%
		0,75	1,4	14	Timeout (3 hr)	27,27%
			1,8	9	Timeout (3 hr)	12,50%
		0,9	1,4	13	Timeout (3 hr)	18,18%
			1,8	9	Timeout (3 hr)	0,00%
	75%	0,1	1,4	21	Timeout (3 hr)	90,91%
			1,8	12	Timeout (3 hr)	50,00%
		0,25	1,4	18	Timeout (3 hr)	63,64%
			1,8	11	Timeout (3 hr)	22,22%
		0,5	1,4	18	Timeout (3 hr)	50,00%
			1,8	10	Timeout (3 hr)	11,11%
		0,75	1,4	17	Timeout (3 hr)	30,77%
			1,8	11	Timeout (3 hr)	10,00%
		0,9	1,4	15	Timeout (3 hr)	15,38%
			1,8	10	Timeout (3 hr)	0,00%
40	25%	0,1	1,4	19	Timeout (3 hr)	90,00%
			1,8	12	Timeout (3 hr)	50,00%
		0,25	1,4	18	Timeout (3 hr)	63,64%
			1,8	10	Timeout (3 hr)	25,00%
		0,5	1,4	17	Timeout (3 hr)	54,55%
			1,8	9	Timeout (3 hr)	12,50%
		0,75	1,4	14	Timeout (3 hr)	27,27%
			1,8	10	Timeout (3 hr)	11,11%
		0,9	1,4	14	Timeout (3 hr)	16,67%
			1,8	9	Timeout (3 hr)	0,00%
	50%	0,1	1,4	23	Timeout (3 hr)	91,67%
			1,8	15	Timeout (3 hr)	50,00%
		0,25	1,4	21	Timeout (3 hr)	61,54%
			1,8	12	Timeout (3 hr)	20,00%
		0,5	1,4	19	Timeout (3 hr)	46,15%
			1,8	11	Timeout (3 hr)	10,00%
		0,75	1,4	18	Timeout (3 hr)	28,57%
			1,8	12	Timeout (3 hr)	9,09%
		0,9	1,4	16	Timeout (3 hr)	14,29%
			1,8	12	Timeout (3 hr)	0,00%
	75%	0,1	1,4	27	Timeout (3 hr)	92,86%
			1,8	18	Timeout (3 hr)	50,00%
		0,25	1,4	25	Timeout (3 hr)	66,67%
			1,8	16	Timeout (3 hr)	23,08%
		0,5	1,4	22	Timeout (3 hr)	46,67%
			1,8	15	Timeout (3 hr)	15,38%
		0,75	1,4	20	Timeout (3 hr)	25,00%
			1,8	14	Timeout (3 hr)	7,69%
		0,9	1,4	20	Timeout (3 hr)	17,65%
			1,8	15	Timeout (3 hr)	7,14%

(Table B.2 Continued)

RCPSP-I						
Set of Tasks (N)	**Multi-manned Task Perc.(%)**	**Order Strength**	**Cycle Time Multiplier**	**Number of Stations Opened**	**Solution Time (*sec.*)**	**Gap**
50	25%	0,1	1,4	25	Timeout (3 hr)	92,31%
			1,8	15	Timeout (3 hr)	50,00%
		0,25	1,4	23	Timeout (3 hr)	64,29%
			1,8	12	Timeout (3 hr)	20,00%
		0,5	1,4	21	Timeout (3 hr)	50,00%
			1,8	12	Timeout (3 hr)	9,09%
		0,75	1,4	19	Timeout (3 hr)	26,67%
			1,8	13	Timeout (3 hr)	8,33%
		0,9	1,4	19	Timeout (3 hr)	18,75%
			1,8	13	Timeout (3 hr)	0,00%
	50%	0,1	1,4	29	Timeout (3 hr)	93,33%
			1,8	17	Timeout (3 hr)	41,67%
		0,25	1,4	26	Timeout (3 hr)	62,50%
			1,8	16	Timeout (3 hr)	23,08%
		0,5	1,4	24	Timeout (3 hr)	50,00%
			1,8	15	Timeout (3 hr)	15,38%
		0,75	1,4	21	Timeout (3 hr)	23,53%
			1,8	15	Timeout (3 hr)	7,14%
		0,9	1,4	21	Timeout (3 hr)	16,67%
			1,8	14	Timeout (3 hr)	0,00%
	75%	0,1	1,4	35	Timeout (3 hr)	94,44%
			1,8	20	Timeout (3 hr)	42,86%
		0,25	1,4	30	Timeout (3 hr)	66,67%
			1,8	17	Timeout (3 hr)	21,43%
		0,5	1,4	28	Timeout (3 hr)	47,37%
			1,8	17	Timeout (3 hr)	13,33%
		0,75	1,4	25	Timeout (3 hr)	25,00%
			1,8	16	Timeout (3 hr)	6,67%
		0,9	1,4	24	Timeout (3 hr)	14,29%
			1,8	16	Timeout (3 hr)	0,00%
60	25%	0,1	1,4	29	Timeout (3 hr)	93,33%
			1,8	18	Timeout (3 hr)	50,00%
		0,25	1,4	26	Timeout (3 hr)	62,50%
			1,8	16	Timeout (3 hr)	23,08%
		0,5	1,4	24	Timeout (3 hr)	50,00%
			1,8	15	Timeout (3 hr)	15,38%
		0,75	1,4	21	Timeout (3 hr)	23,53%
			1,8	15	Timeout (3 hr)	7,14%
		0,9	1,4	21	Timeout (3 hr)	16,67%
			1,8	15	Timeout (3 hr)	7,14%
	50%	0,1	1,4	35	Timeout (3 hr)	94,44%
			1,8	21	Timeout (3 hr)	50,00%
		0,25	1,4	31	Timeout (3 hr)	63,16%
			1,8	19	Timeout (3 hr)	26,67%
		0,5	1,4	30	Timeout (3 hr)	50,00%
			1,8	18	Timeout (3 hr)	12,50%
		0,75	1,4	27	Timeout (3 hr)	28,57%
			1,8	19	Timeout (3 hr)	11,76%
		0,9	1,4	25	Timeout (3 hr)	13,64%
			1,8	18	Timeout (3 hr)	5,88%
	75%	0,1	1,4	40	Timeout (3 hr)	90,48%
			1,8	25	Timeout (3 hr)	47,06%
		0,25	1,4	36	Timeout (3 hr)	63,64%
			1,8	22	Timeout (3 hr)	22,22%
		0,5	1,4	34	Timeout (3 hr)	47,83%
			1,8	21	Timeout (3 hr)	10,53%
		0,75	1,4	30	Timeout (3 hr)	25,00%
			1,8	22	Timeout (3 hr)	10,00%
		0,9	1,4	29	Timeout (3 hr)	16,00%
			1,8	22	Timeout (3 hr)	4,76%

Table B. 3: Results of CP-I Formulation

CP-I						
Set of Tasks (N)	Multi-manned Task Perc.(%)	Order Strength	Cycle Time Multiplier	Number of Stations Opened	Solution Time (*sec.*)	Gap
30	25%	0,1	1,4	8	758	0%
			1,8	6	256	0%
		0,25	1,4	8	553	0%
			1,8	6	228	0%
		0,5	1,4	8	420	0%
			1,8	6	138	0%
		0,75	1,4	9	282	0%
			1,8	7	78	0%
		0,9	1,4	10	142	0%
			1,8	7	34	0%
	50%	0,1	1,4	9	1241	0%
			1,8	7	594	0%
		0,25	1,4	10	926	0%
			1,8	8	427	0%
		0,5	1,4	10	762	0%
			1,8	8	276	0%
		0,75	1,4	11	599	0%
			1,8	8	189	0%
		0,9	1,4	11	389	0%
			1,8	9	149	0%
	75%	0,1	1,4	11	2269	0%
			1,8	8	1866	0%
		0,25	1,4	11	1664	0%
			1,8	9	1237	0%
		0,5	1,4	12	1257	0%
			1,8	9	587	0%
		0,75	1,4	13	1014	0%
			1,8	10	663	0%
		0,9	1,4	13	761	0%
			1,8	10	331	0%
40	25%	0,1	1,4	10	1413	0%
			1,8	8	540	0%
		0,25	1,4	11	1087	0%
			1,8	8	419	0%
		0,5	1,4	11	755	0%
			1,8	8	242	0%
		0,75	1,4	11	424	0%
			1,8	9	108	0%
		0,9	1,4	12	209	0%
			1,8	9	38	0%
	50%	0,1	1,4	12	2113	0%
			1,8	10	907	0%
		0,25	1,4	13	1562	0%
			1,8	10	770	0%
		0,5	1,4	13	1277	0%
			1,8	10	469	0%
		0,75	1,4	14	980	0%
			1,8	11	309	0%
		0,9	1,4	14	607	0%
			1,8	12	229	0%
	75%	0,1	1,4	14	3831	0%
			1,8	12	3072	0%
		0,25	1,4	15	2831	0%
			1,8	13	2074	0%
		0,5	1,4	15	2122	0%
			1,8	13	1040	0%
		0,75	1,4	16	1669	0%
			1,8	13	1071	0%
		0,9	1,4	17	1219	0%
			1,8	14	518	0%

(Table B.3 Continued)

CP-I						
Set of Tasks (N)	Multi-manned Task Perc.(%)	Order Strength	Cycle Time Multiplier	Number of Stations Opened	Solution Time (*sec.*)	Gap
50	25%	0,1	1,4	13	2189	0%
			1,8	10	768	0%
		0,25	1,4	14	1526	0%
			1,8	10	624	0%
		0,5	1,4	14	1231	0%
			1,8	11	392	0%
		0,75	1,4	15	786	0%
			1,8	12	208	0%
		0,9	1,4	16	372	0%
			1,8	13	72	0%
	50%	0,1	1,4	15	3587	0%
			1,8	12	1701	0%
		0,25	1,4	16	2620	0%
			1,8	13	1087	0%
		0,5	1,4	16	2128	0%
			1,8	13	808	0%
		0,75	1,4	17	1637	0%
			1,8	14	482	0%
		0,9	1,4	18	1020	0%
			1,8	14	407	0%
	75%	0,1	1,4	18	9196	0%
			1,8	14	5405	0%
		0,25	1,4	18	4884	0%
			1,8	14	3634	0%
		0,5	1,4	19	3721	0%
			1,8	15	1776	0%
		0,75	1,4	20	2896	0%
			1,8	15	1950	0%
		0,9	1,4	21	2202	0%
			1,8	16	875	0%
60	25%	0,1	1,4	15	3273	0%
			1,8	12	1989	0%
		0,25	1,4	16	2359	0%
			1,8	13	919	0%
		0,5	1,4	16	1802	0%
			1,8	13	569	0%
		0,75	1,4	17	1084	0%
			1,8	14	266	0%
		0,9	1,4	18	499	0%
			1,8	14	97	0%
	50%	0,1	1,4	18	5369	0%
			1,8	14	2258	0%
		0,25	1,4	19	3969	0%
			1,8	15	1819	0%
		0,5	1,4	20	3295	0%
			1,8	16	1259	0%
		0,75	1,4	21	2421	0%
			1,8	17	706	0%
		0,9	1,4	22	1627	0%
			1,8	17	592	0%
	75%	0,1	1,4	21	Timeout (3 hr)	0%
			1,8	17	7739	0%
		0,25	1,4	22	Timeout (3 hr)	0%
			1,8	18	5375	0%
		0,5	1,4	23	5527	0%
			1,8	19	2654	0%
		0,75	1,4	24	3995	0%
			1,8	20	2872	0%
		0,9	1,4	25	3270	0%
			1,8	21	1495	0%

Table B. 4: Results of AD-RCPSP-I Formulation

AD-RCPSP-I						
Set of Tasks (N)	**Multi-manned Task Perc.(%)**	**Order Strength**	**Cycle Time Multiplier**	**Number of Stations Opened**	**Solution Time (*sec.*)**	**Gap**
30	25%	0,1	1,4	15	Timeout (3 hr)	87,50%
			1,8	9	Timeout (3 hr)	50,00%
		0,25	1,4	13	Timeout (3 hr)	62,50%
			1,8	7	Timeout (3 hr)	16,67%
		0,5	1,4	12	Timeout (3 hr)	50,00%
			1,8	7	Timeout (3 hr)	16,67%
		0,75	1,4	11	Timeout (3 hr)	22,22%
			1,8	7	Timeout (3 hr)	0,00%
		0,9	1,4	11	Timeout (3 hr)	10,00%
			1,8	7	Timeout (3 hr)	0,00%
	50%	0,1	1,4	17	Timeout (3 hr)	88,89%
			1,8	10	Timeout (3 hr)	42,86%
		0,25	1,4	17	Timeout (3 hr)	70,00%
			1,8	10	Timeout (3 hr)	25,00%
		0,5	1,4	14	Timeout (3 hr)	40,00%
			1,8	9	Timeout (3 hr)	12,50%
		0,75	1,4	13	Timeout (3 hr)	18,18%
			1,8	9	Timeout (3 hr)	12,50%
		0,9	1,4	12	Timeout (3 hr)	9,09%
			1,8	9	Timeout (3 hr)	0,00%
	75%	0,1	1,4	21	Timeout (3 hr)	90,91%
			1,8	12	Timeout (3 hr)	50,00%
		0,25	1,4	18	Timeout (3 hr)	63,64%
			1,8	11	Timeout (3 hr)	22,22%
		0,5	1,4	17	Timeout (3 hr)	41,67%
			1,8	10	Timeout (3 hr)	11,11%
		0,75	1,4	15	Timeout (3 hr)	15,38%
			1,8	11	Timeout (3 hr)	10,00%
		0,9	1,4	14	Timeout (3 hr)	7,69%
			1,8	10	Timeout (3 hr)	0,00%
40	25%	0,1	1,4	19	Timeout (3 hr)	90,00%
			1,8	12	Timeout (3 hr)	50,00%
		0,25	1,4	18	Timeout (3 hr)	63,64%
			1,8	10	Timeout (3 hr)	25,00%
		0,5	1,4	16	Timeout (3 hr)	45,45%
			1,8	9	Timeout (3 hr)	12,50%
		0,75	1,4	13	Timeout (3 hr)	18,18%
			1,8	10	Timeout (3 hr)	11,11%
		0,9	1,4	12	Timeout (3 hr)	0,00%
			1,8	9	Timeout (3 hr)	0,00%
	50%	0,1	1,4	23	Timeout (3 hr)	91,67%
			1,8	15	Timeout (3 hr)	50,00%
		0,25	1,4	21	Timeout (3 hr)	61,54%
			1,8	12	Timeout (3 hr)	20,00%
		0,5	1,4	17	Timeout (3 hr)	30,77%
			1,8	11	Timeout (3 hr)	10,00%
		0,75	1,4	15	Timeout (3 hr)	7,14%
			1,8	12	Timeout (3 hr)	9,09%
		0,9	1,4	14	Timeout (3 hr)	0,00%
			1,8	12	Timeout (3 hr)	0,00%
	75%	0,1	1,4	27	Timeout (3 hr)	92,86%
			1,8	18	Timeout (3 hr)	50,00%
		0,25	1,4	25	Timeout (3 hr)	66,67%
			1,8	16	Timeout (3 hr)	23,08%
		0,5	1,4	21	Timeout (3 hr)	40,00%
			1,8	15	Timeout (3 hr)	15,38%
		0,75	1,4	18	Timeout (3 hr)	12,50%
			1,8	14	Timeout (3 hr)	7,69%
		0,9	1,4	18	Timeout (3 hr)	5,88%
			1,8	14	Timeout (3 hr)	0,00%

(Table B.4 Continued)

AD-RCPSP-I						
Set of Tasks (N)	**Multi-manned Task Perc.(%)**	**Order Strength**	**Cycle Time Multiplier**	**Number of Stations Opened**	**Solution Time (*sec.*)**	**Gap**
50	25%	0,1	1,4	25	Timeout (3 hr)	92,31%
			1,8	15	Timeout (3 hr)	50,00%
		0,25	1,4	23	Timeout (3 hr)	64,29%
			1,8	12	Timeout (3 hr)	20,00%
		0,5	1,4	21	Timeout (3 hr)	50,00%
			1,8	12	Timeout (3 hr)	9,09%
		0,75	1,4	19	Timeout (3 hr)	26,67%
			1,8	13	Timeout (3 hr)	8,33%
		0,9	1,4	18	Timeout (3 hr)	12,50%
			1,8	13	Timeout (3 hr)	0,00%
	50%	0,1	1,4	29	Timeout (3 hr)	93,33%
			1,8	17	Timeout (3 hr)	41,67%
		0,25	1,4	26	Timeout (3 hr)	62,50%
			1,8	16	Timeout (3 hr)	23,08%
		0,5	1,4	24	Timeout (3 hr)	50,00%
			1,8	15	Timeout (3 hr)	15,38%
		0,75	1,4	21	Timeout (3 hr)	23,53%
			1,8	15	Timeout (3 hr)	7,14%
		0,9	1,4	20	Timeout (3 hr)	11,11%
			1,8	14	Timeout (3 hr)	0,00%
	75%	0,1	1,4	35	Timeout (3 hr)	94,44%
			1,8	20	Timeout (3 hr)	42,86%
		0,25	1,4	30	Timeout (3 hr)	66,67%
			1,8	17	Timeout (3 hr)	21,43%
		0,5	1,4	28	Timeout (3 hr)	47,37%
			1,8	17	Timeout (3 hr)	13,33%
		0,75	1,4	25	Timeout (3 hr)	25,00%
			1,8	16	Timeout (3 hr)	6,67%
		0,9	1,4	23	Timeout (3 hr)	9,52%
			1,8	16	Timeout (3 hr)	0,00%
60	25%	0,1	1,4	29	Timeout (3 hr)	93,33%
			1,8	18	Timeout (3 hr)	50,00%
		0,25	1,4	26	Timeout (3 hr)	62,50%
			1,8	16	Timeout (3 hr)	23,08%
		0,5	1,4	24	Timeout (3 hr)	50,00%
			1,8	15	Timeout (3 hr)	15,38%
		0,75	1,4	21	Timeout (3 hr)	23,53%
			1,8	15	Timeout (3 hr)	7,14%
		0,9	1,4	20	Timeout (3 hr)	11,11%
			1,8	12	Timeout (3 hr)	-14,29%
	50%	0,1	1,4	35	Timeout (3 hr)	94,44%
			1,8	21	Timeout (3 hr)	50,00%
		0,25	1,4	31	Timeout (3 hr)	63,16%
			1,8	19	Timeout (3 hr)	26,67%
		0,5	1,4	30	Timeout (3 hr)	50,00%
			1,8	18	Timeout (3 hr)	12,50%
		0,75	1,4	27	Timeout (3 hr)	28,57%
			1,8	19	Timeout (3 hr)	11,76%
		0,9	1,4	24	Timeout (3 hr)	9,09%
			1,8	18	Timeout (3 hr)	5,88%
	75%	0,1	1,4	40	Timeout (3 hr)	90,48%
			1,8	25	Timeout (3 hr)	47,06%
		0,25	1,4	36	Timeout (3 hr)	63,64%
			1,8	22	Timeout (3 hr)	22,22%
		0,5	1,4	34	Timeout (3 hr)	47,83%
			1,8	21	Timeout (3 hr)	10,53%
		0,75	1,4	30	Timeout (3 hr)	25,00%
			1,8	22	Timeout (3 hr)	10,00%
		0,9	1,4	28	Timeout (3 hr)	12,00%
			1,8	22	Timeout (3 hr)	4,76%

Table B. 5: Results of AD-CP-I Formulation

AD-CP-I						
Set of Tasks (N)	**Multi-manned Task Perc.(%)**	**Order Strength**	**Cycle Time Multiplier**	**Number of Stations Opened**	**Solution Time (*sec.*)**	**Gap**
30	25%	0,1	1,4	8	754	0%
			1,8	6	249	0%
		0,25	1,4	8	550	0%
			1,8	6	226	0%
		0,5	1,4	8	405	0%
			1,8	6	118	0%
		0,75	1,4	9	259	0%
			1,8	7	46	0%
		0,9	1,4	10	96	0%
			1,8	7	5	0%
	50%	0,1	1,4	9	1241	0%
			1,8	7	590	0%
		0,25	1,4	10	926	0%
			1,8	8	418	0%
		0,5	1,4	10	747	0%
			1,8	8	258	0%
		0,75	1,4	11	574	0%
			1,8	8	158	0%
		0,9	1,4	11	346	0%
			1,8	9	111	0%
	75%	0,1	1,4	11	2269	0%
			1,8	8	1858	0%
		0,25	1,4	11	1662	0%
			1,8	9	1230	0%
		0,5	1,4	12	1237	0%
			1,8	9	570	0%
		0,75	1,4	13	983	0%
			1,8	10	639	0%
		0,9	1,4	13	719	0%
			1,8	10	290	0%
40	25%	0,1	1,4	10	1405	0%
			1,8	8	532	0%
		0,25	1,4	11	1081	0%
			1,8	8	419	0%
		0,5	1,4	11	735	0%
			1,8	8	227	0%
		0,75	1,4	11	393	0%
			1,8	9	84	0%
		0,9	1,4	12	164	0%
			1,8	9	8	0%
	50%	0,1	1,4	12	2108	0%
			1,8	10	900	0%
		0,25	1,4	13	1555	0%
			1,8	10	766	0%
		0,5	1,4	13	1257	0%
			1,8	10	450	0%
		0,75	1,4	14	953	0%
			1,8	11	281	0%
		0,9	1,4	14	564	0%
			1,8	12	184	0%
	75%	0,1	1,4	14	3825	0%
			1,8	12	3072	0%
		0,25	1,4	15	2829	0%
			1,8	13	2067	0%
		0,5	1,4	15	2110	0%
			1,8	13	1023	0%
		0,75	1,4	16	1642	0%
			1,8	13	1048	0%
		0,9	1,4	17	1178	0%
			1,8	14	476	0%

(Table B.5 Continued)

AD-CP-I						
Set of Tasks (N)	**Multi-manned Task Perc.(%)**	**Order Strength**	**Cycle Time Multiplier**	**Number of Stations Opened**	**Solution Time (*sec.*)**	**Gap**
50	25%	0,1	1,4	13	2185	0%
			1,8	10	762	0%
		0,25	1,4	14	1519	0%
			1,8	10	616	0%
		0,5	1,4	14	1214	0%
			1,8	11	378	0%
		0,75	1,4	15	757	0%
			1,8	12	177	0%
		0,9	1,4	16	330	0%
			1,8	13	27	0%
	50%	0,1	1,4	15	3578	0%
			1,8	12	1695	0%
		0,25	1,4	16	2612	0%
			1,8	13	1085	0%
		0,5	1,4	16	2115	0%
			1,8	13	790	0%
		0,75	1,4	17	1613	0%
			1,8	14	457	0%
		0,9	1,4	18	981	0%
			1,8	14	366	0%
	75%	0,1	1,4	18	9188	0%
			1,8	14	5396	0%
		0,25	1,4	18	4876	0%
			1,8	14	3634	0%
		0,5	1,4	19	3709	0%
			1,8	15	1763	0%
		0,75	1,4	20	2872	0%
			1,8	15	1923	0%
		0,9	1,4	21	2155	0%
			1,8	16	831	0%
60	25%	0,1	1,4	15	3267	0%
			1,8	12	1982	0%
		0,25	1,4	16	2354	0%
			1,8	13	919	0%
		0,5	1,4	16	1786	0%
			1,8	13	550	0%
		0,75	1,4	17	1059	0%
			1,8	14	238	0%
		0,9	1,4	18	453	0%
			1,8	14	56	0%
	50%	0,1	1,4	18	5364	0%
			1,8	14	2253	0%
		0,25	1,4	19	3961	0%
			1,8	15	1816	0%
		0,5	1,4	20	3280	0%
			1,8	16	1240	0%
		0,75	1,4	21	2390	0%
			1,8	17	682	0%
		0,9	1,4	22	1583	0%
			1,8	17	550	0%
	75%	0,1	1,4	21	Timeout (3 hr)	0%
			1,8	17	7735	0%
		0,25	1,4	22	Timeout (3 hr)	0%
			1,8	18	5367	0%
		0,5	1,4	23	5516	0%
			1,8	19	2637	0%
		0,75	1,4	24	3965	0%
			1,8	20	2845	0%
		0,9	1,4	25	3223	0%
			1,8	21	1454	0%

Table B. 6: Results of GMM-II Formulation

GMM-II						
Set of Tasks (N)	**Multi-manned Task Perc.(%)**	**Order Strength**	**Number of Stations Multiplier**	**Cycle Time**	**Solution Time (*sec.*)**	**Gap**
30	25%	0,1	1,4	150	Timeout (3 hr)	11,94%
			1,8	187	Timeout (3 hr)	8,09%
		0,25	1,4	150	Timeout (3 hr)	11,94%
			1,8	187	Timeout (3 hr)	8,09%
		0,5	1,4	151	Timeout (3 hr)	7,86%
			1,8	185	Timeout (3 hr)	2,78%
		0,75	1,4	160	Timeout (3 hr)	2,56%
			1,8	197	Timeout (3 hr)	0,00%
		0,9	1,4	160	Timeout (3 hr)	2,56%
			1,8	197	Timeout (3 hr)	0,00%
	50%	0,1	1,4	167	Timeout (3 hr)	20,14%
			1,8	206	Timeout (3 hr)	15,73%
		0,25	1,4	167	Timeout (3 hr)	15,17%
			1,8	204	Timeout (3 hr)	9,68%
		0,5	1,4	167	Timeout (3 hr)	15,17%
			1,8	204	Timeout (3 hr)	9,68%
		0,75	1,4	166	Timeout (3 hr)	5,06%
			1,8	209	Timeout (3 hr)	2,96%
		0,9	1,4	166	Timeout (3 hr)	5,06%
			1,8	203	Timeout (3 hr)	0,00%
	75%	0,1	1,4	170	Timeout (3 hr)	25,00%
			1,8	213	Timeout (3 hr)	18,33%
		0,25	1,4	163	Timeout (3 hr)	19,85%
			1,8	216	Timeout (3 hr)	13,09%
		0,5	1,4	162	Timeout (3 hr)	10,96%
			1,8	204	Timeout (3 hr)	6,81%
		0,75	1,4	168	Timeout (3 hr)	7,01%
			1,8	222	Timeout (3 hr)	6,22%
		0,9	1,4	163	Timeout (3 hr)	3,82%
			1,8	209	Timeout (3 hr)	0,00%
40	25%	0,1	1,4	173	Timeout (3 hr)	24,46%
			1,8	208	Timeout (3 hr)	20,93%
		0,25	1,4	183	Timeout (3 hr)	23,65%
			1,8	213	Timeout (3 hr)	20,34%
		0,5	1,4	188	Timeout (3 hr)	20,51%
			1,8	210	Timeout (3 hr)	16,67%
		0,75	1,4	179	Timeout (3 hr)	14,74%
			1,8	221	Timeout (3 hr)	14,51%
		0,9	1,4	197	Timeout (3 hr)	15,20%
			1,8	220	Timeout (3 hr)	13,99%
	50%	0,1	1,4	179	Timeout (3 hr)	29,71%
			1,8	209	Timeout (3 hr)	25,90%
		0,25	1,4	182	Timeout (3 hr)	26,39%
			1,8	210	Timeout (3 hr)	21,39%
		0,5	1,4	185	Timeout (3 hr)	23,33%
			1,8	213	Timeout (3 hr)	18,99%
		0,75	1,4	185	Timeout (3 hr)	19,35%
			1,8	231	Timeout (3 hr)	18,46%
		0,9	1,4	186	Timeout (3 hr)	16,98%
			1,8	236	Timeout (3 hr)	15,69%
	75%	0,1	1,4	189	Timeout (3 hr)	35,00%
			1,8	215	Timeout (3 hr)	29,52%
		0,25	1,4	185	Timeout (3 hr)	29,37%
			1,8	215	Timeout (3 hr)	18,78%
		0,5	1,4	185	Timeout (3 hr)	24,16%
			1,8	226	Timeout (3 hr)	21,51%
		0,75	1,4	191	Timeout (3 hr)	21,66%
			1,8	241	Timeout (3 hr)	24,23%
		0,9	1,4	211	Timeout (3 hr)	24,85%
			1,8	241	Timeout (3 hr)	15,31%

(Table B.6 Continued)

GMM-II						
Set of Tasks (N)	Multi-manned Task Perc.(%)	Order Strength	Number of Stations Multiplier	Cycle Time	Solution Time (*sec.*)	Gap
50	25%	0,1	1,4	205	Timeout (3 hr)	48,55%
			1,8	249	Timeout (3 hr)	42,29%
		0,25	1,4	204	Timeout (3 hr)	44,68%
			1,8	249	Timeout (3 hr)	38,33%
		0,5	1,4	205	Timeout (3 hr)	39,46%
			1,8	256	Timeout (3 hr)	31,96%
		0,75	1,4	207	Timeout (3 hr)	32,69%
			1,8	275	Timeout (3 hr)	29,11%
		0,9	1,4	221	Timeout (3 hr)	31,55%
			1,8	289	Timeout (3 hr)	29,02%
	50%	0,1	1,4	215	Timeout (3 hr)	54,68%
			1,8	262	Timeout (3 hr)	48,86%
		0,25	1,4	210	Timeout (3 hr)	46,85%
			1,8	261	Timeout (3 hr)	43,41%
		0,5	1,4	210	Timeout (3 hr)	41,89%
			1,8	253	Timeout (3 hr)	34,57%
		0,75	1,4	209	Timeout (3 hr)	35,71%
			1,8	261	Timeout (3 hr)	31,82%
		0,9	1,4	219	Timeout (3 hr)	32,73%
			1,8	269	Timeout (3 hr)	29,33%
	75%	0,1	1,4	217	Timeout (3 hr)	61,94%
			1,8	259	Timeout (3 hr)	51,46%
		0,25	1,4	213	Timeout (3 hr)	52,14%
			1,8	254	Timeout (3 hr)	42,70%
		0,5	1,4	210	Timeout (3 hr)	44,83%
			1,8	254	Timeout (3 hr)	38,04%
		0,75	1,4	210	Timeout (3 hr)	37,25%
			1,8	263	Timeout (3 hr)	37,70%
		0,9	1,4	227	Timeout (3 hr)	34,32%
			1,8	267	Timeout (3 hr)	28,99%
60	25%	0,1	1,4	210	Timeout (3 hr)	52,17%
			1,8	265	Timeout (3 hr)	48,04%
		0,25	1,4	215	Timeout (3 hr)	49,31%
			1,8	272	Timeout (3 hr)	43,92%
		0,5	1,4	211	Timeout (3 hr)	41,61%
			1,8	265	Timeout (3 hr)	35,90%
		0,75	1,4	210	Timeout (3 hr)	33,76%
			1,8	266	Timeout (3 hr)	31,03%
		0,9	1,4	230	Timeout (3 hr)	34,50%
			1,8	276	Timeout (3 hr)	30,81%
	50%	0,1	1,4	221	Timeout (3 hr)	58,99%
			1,8	275	Timeout (3 hr)	52,78%
		0,25	1,4	220	Timeout (3 hr)	50,68%
			1,8	274	Timeout (3 hr)	44,97%
		0,5	1,4	222	Timeout (3 hr)	45,10%
			1,8	278	Timeout (3 hr)	38,31%
		0,75	1,4	222	Timeout (3 hr)	38,75%
			1,8	280	Timeout (3 hr)	33,97%
		0,9	1,4	231	Timeout (3 hr)	36,69%
			1,8	280	Timeout (3 hr)	30,23%
	75%	0,1	1,4	232	Timeout (3 hr)	65,71%
			1,8	271	Timeout (3 hr)	57,56%
		0,25	1,4	224	Timeout (3 hr)	54,48%
			1,8	271	Timeout (3 hr)	44,92%
		0,5	1,4	224	Timeout (3 hr)	48,34%
			1,8	282	Timeout (3 hr)	43,88%
		0,75	1,4	220	Timeout (3 hr)	39,24%
			1,8	284	Timeout (3 hr)	35,89%
		0,9	1,4	224	Timeout (3 hr)	36,59%
			1,8	284	Timeout (3 hr)	30,88%

Table B. 7: Results of RCPSP-II Formulation

RCPSP-II						
Set of Tasks (N)	Multi-manned Task Perc.(%)	Order Strength	Number of Stations Multiplier	Cycle Time	Solution Time (*sec.*)	Gap
30	25%	0,1	1,4	160	Timeout (3 hr)	19,40%
			1,8	204	Timeout (3 hr)	17,92%
		0,25	1,4	160	Timeout (3 hr)	19,40%
			1,8	204	Timeout (3 hr)	17,92%
		0,5	1,4	162	Timeout (3 hr)	15,71%
			1,8	213	Timeout (3 hr)	18,33%
		0,75	1,4	181	Timeout (3 hr)	16,03%
			1,8	226	Timeout (3 hr)	14,72%
		0,9	1,4	181	Timeout (3 hr)	16,03%
			1,8	226	Timeout (3 hr)	14,72%
	50%	0,1	1,4	190	Timeout (3 hr)	36,69%
			1,8	221	Timeout (3 hr)	24,16%
		0,25	1,4	190	Timeout (3 hr)	31,03%
			1,8	216	Timeout (3 hr)	16,13%
		0,5	1,4	190	Timeout (3 hr)	31,03%
			1,8	216	Timeout (3 hr)	16,13%
		0,75	1,4	190	Timeout (3 hr)	20,25%
			1,8	235	Timeout (3 hr)	15,76%
		0,9	1,4	190	Timeout (3 hr)	20,25%
			1,8	228	Timeout (3 hr)	12,32%
	75%	0,1	1,4	196	Timeout (3 hr)	44,12%
			1,8	234	Timeout (3 hr)	30,00%
		0,25	1,4	183	Timeout (3 hr)	34,56%
			1,8	229	Timeout (3 hr)	19,90%
		0,5	1,4	184	Timeout (3 hr)	26,03%
			1,8	217	Timeout (3 hr)	13,61%
		0,75	1,4	189	Timeout (3 hr)	20,38%
			1,8	246	Timeout (3 hr)	17,70%
		0,9	1,4	188	Timeout (3 hr)	19,75%
			1,8	224	Timeout (3 hr)	7,18%
40	25%	0,1	1,4	200	Timeout (3 hr)	43,88%
			1,8	226	Timeout (3 hr)	31,40%
		0,25	1,4	206	Timeout (3 hr)	39,19%
			1,8	244	Timeout (3 hr)	37,85%
		0,5	1,4	203	Timeout (3 hr)	30,13%
			1,8	243	Timeout (3 hr)	35,00%
		0,75	1,4	203	Timeout (3 hr)	30,13%
			1,8	243	Timeout (3 hr)	25,91%
		0,9	1,4	218	Timeout (3 hr)	27,49%
			1,8	249	Timeout (3 hr)	29,02%
	50%	0,1	1,4	192	Timeout (3 hr)	39,13%
			1,8	221	Timeout (3 hr)	33,13%
		0,25	1,4	199	Timeout (3 hr)	38,19%
			1,8	241	Timeout (3 hr)	39,31%
		0,5	1,4	209	Timeout (3 hr)	39,33%
			1,8	229	Timeout (3 hr)	27,93%
		0,75	1,4	204	Timeout (3 hr)	31,61%
			1,8	262	Timeout (3 hr)	34,36%
		0,9	1,4	209	Timeout (3 hr)	31,45%
			1,8	254	Timeout (3 hr)	24,51%
	75%	0,1	1,4	205	Timeout (3 hr)	46,43%
			1,8	232	Timeout (3 hr)	39,76%
		0,25	1,4	209	Timeout (3 hr)	46,15%
			1,8	238	Timeout (3 hr)	31,49%
		0,5	1,4	211	Timeout (3 hr)	41,61%
			1,8	252	Timeout (3 hr)	35,48%
		0,75	1,4	212	Timeout (3 hr)	35,03%
			1,8	269	Timeout (3 hr)	38,66%
		0,9	1,4	232	Timeout (3 hr)	37,28%
			1,8	260	Timeout (3 hr)	24,40%

(Table B.7 Continued)

RCPSP-II						
Set of Tasks (N)	**Multi-manned Task Perc.(%)**	**Order Strength**	**Number of Stations Multiplier**	**Cycle Time**	**Solution Time (*sec.*)**	**Gap**
50	25%	0,1	1,4	222	Timeout (3 hr)	60,87%
			1,8	274	Timeout (3 hr)	56,57%
		0,25	1,4	230	Timeout (3 hr)	63,12%
			1,8	285	Timeout (3 hr)	58,33%
		0,5	1,4	228	Timeout (3 hr)	55,10%
			1,8	292	Timeout (3 hr)	50,52%
		0,75	1,4	224	Timeout (3 hr)	43,59%
			1,8	300	Timeout (3 hr)	40,85%
		0,9	1,4	239	Timeout (3 hr)	42,26%
			1,8	307	Timeout (3 hr)	37,05%
	50%	0,1	1,4	242	Timeout (3 hr)	74,10%
			1,8	283	Timeout (3 hr)	60,80%
		0,25	1,4	233	Timeout (3 hr)	62,94%
			1,8	301	Timeout (3 hr)	65,38%
		0,5	1,4	226	Timeout (3 hr)	52,70%
			1,8	269	Timeout (3 hr)	43,09%
		0,75	1,4	222	Timeout (3 hr)	44,16%
			1,8	292	Timeout (3 hr)	47,47%
		0,9	1,4	234	Timeout (3 hr)	41,82%
			1,8	307	Timeout (3 hr)	47,60%
	75%	0,1	1,4	251	Timeout (3 hr)	87,31%
			1,8	275	Timeout (3 hr)	60,82%
		0,25	1,4	230	Timeout (3 hr)	64,29%
			1,8	294	Timeout (3 hr)	65,17%
		0,5	1,4	230	Timeout (3 hr)	58,62%
			1,8	294	Timeout (3 hr)	59,78%
		0,75	1,4	227	Timeout (3 hr)	48,37%
			1,8	302	Timeout (3 hr)	58,12%
		0,9	1,4	253	Timeout (3 hr)	49,70%
			1,8	296	Timeout (3 hr)	43,00%
60	25%	0,1	1,4	223	Timeout (3 hr)	61,59%
			1,8	303	Timeout (3 hr)	69,27%
		0,25	1,4	245	Timeout (3 hr)	70,14%
			1,8	307	Timeout (3 hr)	62,43%
		0,5	1,4	232	Timeout (3 hr)	55,70%
			1,8	299	Timeout (3 hr)	53,33%
		0,75	1,4	226	Timeout (3 hr)	43,95%
			1,8	296	Timeout (3 hr)	45,81%
		0,9	1,4	267	Timeout (3 hr)	56,14%
			1,8	309	Timeout (3 hr)	46,45%
	50%	0,1	1,4	243	Timeout (3 hr)	74,82%
			1,8	314	Timeout (3 hr)	74,44%
		0,25	1,4	249	Timeout (3 hr)	70,55%
			1,8	299	Timeout (3 hr)	58,20%
		0,5	1,4	255	Timeout (3 hr)	66,67%
			1,8	324	Timeout (3 hr)	61,19%
		0,75	1,4	242	Timeout (3 hr)	51,25%
			1,8	325	Timeout (3 hr)	55,50%
		0,9	1,4	247	Timeout (3 hr)	46,15%
			1,8	304	Timeout (3 hr)	41,40%
	75%	0,1	1,4	246	Timeout (3 hr)	75,71%
			1,8	314	Timeout (3 hr)	82,56%
		0,25	1,4	259	Timeout (3 hr)	78,62%
			1,8	311	Timeout (3 hr)	66,31%
		0,5	1,4	250	Timeout (3 hr)	65,56%
			1,8	318	Timeout (3 hr)	62,24%
		0,75	1,4	250	Timeout (3 hr)	58,23%
			1,8	313	Timeout (3 hr)	49,76%
		0,9	1,4	246	Timeout (3 hr)	50,00%
			1,8	326	Timeout (3 hr)	50,23%

Table B. 8: Results of CP-II Formulation

CP-II						
Set of Tasks (N)	**Multi-manned Task Perc.(%)**	**Order Strength**	**Number of Stations Multiplier**	**Cycle Time**	**Solution Time (*sec.*)**	**Gap**
30	25%	0,1	1,4	150	Timeout (3 hr)	11,94%
			1,8	187	Timeout (3 hr)	8,09%
		0,25	1,4	150	Timeout (3 hr)	11,94%
			1,8	187	Timeout (3 hr)	8,09%
		0,5	1,4	151	Timeout (3 hr)	7,86%
			1,8	180	Timeout (3 hr)	0,00%
		0,75	1,4	156	Timeout (3 hr)	0,00%
			1,8	197	Timeout (3 hr)	0,00%
		0,9	1,4	156	Timeout (3 hr)	0,00%
			1,8	197	Timeout (3 hr)	0,00%
	50%	0,1	1,4	167	Timeout (3 hr)	20,14%
			1,8	206	Timeout (3 hr)	15,73%
		0,25	1,4	167	Timeout (3 hr)	15,17%
			1,8	204	Timeout (3 hr)	9,68%
		0,5	1,4	167	Timeout (3 hr)	15,17%
			1,8	204	Timeout (3 hr)	9,68%
		0,75	1,4	166	Timeout (3 hr)	5,06%
			1,8	203	Timeout (3 hr)	0,00%
		0,9	1,4	158	Timeout (3 hr)	0,00%
			1,8	203	Timeout (3 hr)	0,00%
	75%	0,1	1,4	170	Timeout (3 hr)	25,00%
			1,8	213	Timeout (3 hr)	18,33%
		0,25	1,4	163	Timeout (3 hr)	19,85%
			1,8	216	Timeout (3 hr)	13,09%
		0,5	1,4	162	Timeout (3 hr)	10,96%
			1,8	204	Timeout (3 hr)	6,81%
		0,75	1,4	168	Timeout (3 hr)	7,01%
			1,8	209	Timeout (3 hr)	0,00%
		0,9	1,4	160	Timeout (3 hr)	1,91%
			1,8	209	Timeout (3 hr)	0,00%
40	25%	0,1	1,4	166	Timeout (3 hr)	19,42%
			1,8	204	Timeout (3 hr)	18,60%
		0,25	1,4	174	Timeout (3 hr)	17,57%
			1,8	202	Timeout (3 hr)	14,12%
		0,5	1,4	176	Timeout (3 hr)	12,82%
			1,8	197	Timeout (3 hr)	9,44%
		0,75	1,4	164	Timeout (3 hr)	5,13%
			1,8	202	Timeout (3 hr)	4,66%
		0,9	1,4	183	Timeout (3 hr)	7,02%
			1,8	193	Timeout (3 hr)	0,00%
	50%	0,1	1,4	174	Timeout (3 hr)	26,09%
			1,8	203	Timeout (3 hr)	22,29%
		0,25	1,4	175	Timeout (3 hr)	21,53%
			1,8	204	Timeout (3 hr)	17,92%
		0,5	1,4	175	Timeout (3 hr)	16,67%
			1,8	203	Timeout (3 hr)	13,41%
		0,75	1,4	172	Timeout (3 hr)	10,97%
			1,8	211	Timeout (3 hr)	8,21%
		0,9	1,4	164	Timeout (3 hr)	3,14%
			1,8	212	Timeout (3 hr)	3,92%
	75%	0,1	1,4	185	Timeout (3 hr)	32,14%
			1,8	209	Timeout (3 hr)	25,90%
		0,25	1,4	178	Timeout (3 hr)	24,48%
			1,8	205	Timeout (3 hr)	13,26%
		0,5	1,4	171	Timeout (3 hr)	14,77%
			1,8	210	Timeout (3 hr)	12,90%
		0,75	1,4	170	Timeout (3 hr)	8,28%
			1,8	218	Timeout (3 hr)	12,37%
		0,9	1,4	186	Timeout (3 hr)	10,06%
			1,8	216	Timeout (3 hr)	3,35%

(Table B.8 Continued)

CP-II						
Set of Tasks (N)	**Multi-manned Task Perc.(%)**	**Order Strength**	**Number of Stations Multiplier**	**Cycle Time**	**Solution Time (*sec.*)**	**Gap**
50	25%	0,1	1,4	201	Timeout (3 hr)	45,65%
			1,8	244	Timeout (3 hr)	39,43%
		0,25	1,4	199	Timeout (3 hr)	41,13%
			1,8	242	Timeout (3 hr)	34,44%
		0,5	1,4	199	Timeout (3 hr)	35,37%
			1,8	246	Timeout (3 hr)	26,80%
		0,75	1,4	196	Timeout (3 hr)	25,64%
			1,8	261	Timeout (3 hr)	22,54%
		0,9	1,4	203	Timeout (3 hr)	20,83%
			1,8	261	Timeout (3 hr)	16,52%
	50%	0,1	1,4	210	Timeout (3 hr)	51,08%
			1,8	256	Timeout (3 hr)	45,45%
		0,25	1,4	203	Timeout (3 hr)	41,96%
			1,8	254	Timeout (3 hr)	39,56%
		0,5	1,4	202	Timeout (3 hr)	36,49%
			1,8	240	Timeout (3 hr)	27,66%
		0,75	1,4	190	Timeout (3 hr)	23,38%
			1,8	239	Timeout (3 hr)	20,71%
		0,9	1,4	193	Timeout (3 hr)	16,97%
			1,8	238	Timeout (3 hr)	14,42%
	75%	0,1	1,4	209	Timeout (3 hr)	55,97%
			1,8	250	Timeout (3 hr)	46,20%
		0,25	1,4	202	Timeout (3 hr)	44,29%
			1,8	241	Timeout (3 hr)	35,39%
		0,5	1,4	195	Timeout (3 hr)	34,48%
			1,8	238	Timeout (3 hr)	29,35%
		0,75	1,4	188	Timeout (3 hr)	22,88%
			1,8	236	Timeout (3 hr)	23,56%
		0,9	1,4	195	Timeout (3 hr)	15,38%
			1,8	229	Timeout (3 hr)	10,63%
60	25%	0,1	1,4	205	Timeout (3 hr)	48,55%
			1,8	258	Timeout (3 hr)	44,13%
		0,25	1,4	208	Timeout (3 hr)	44,44%
			1,8	265	Timeout (3 hr)	40,21%
		0,5	1,4	202	Timeout (3 hr)	35,57%
			1,8	254	Timeout (3 hr)	30,26%
		0,75	1,4	198	Timeout (3 hr)	26,11%
			1,8	252	Timeout (3 hr)	24,14%
		0,9	1,4	214	Timeout (3 hr)	25,15%
			1,8	250	Timeout (3 hr)	18,48%
	50%	0,1	1,4	214	Timeout (3 hr)	53,96%
			1,8	268	Timeout (3 hr)	48,89%
		0,25	1,4	212	Timeout (3 hr)	45,21%
			1,8	266	Timeout (3 hr)	40,74%
		0,5	1,4	211	Timeout (3 hr)	37,91%
			1,8	265	Timeout (3 hr)	31,84%
		0,75	1,4	205	Timeout (3 hr)	28,13%
			1,8	259	Timeout (3 hr)	23,92%
		0,9	1,4	205	Timeout (3 hr)	21,30%
			1,8	249	Timeout (3 hr)	15,81%
	75%	0,1	1,4	222	Timeout (3 hr)	58,57%
			1,8	259	Timeout (3 hr)	50,58%
		0,25	1,4	212	Timeout (3 hr)	46,21%
			1,8	257	Timeout (3 hr)	37,43%
		0,5	1,4	207	Timeout (3 hr)	37,09%
			1,8	263	Timeout (3 hr)	34,18%
		0,75	1,4	198	Timeout (3 hr)	25,32%
			1,8	260	Timeout (3 hr)	24,40%
		0,9	1,4	192	Timeout (3 hr)	17,07%
			1,8	247	Timeout (3 hr)	13,82%

Table B. 9: Results of AD-RCPSP-II Formulation

AD-RCPSP-II						
Set of Tasks (N)	Multi-manned Task Perc.(%)	Order Strength	Number of Stations Multiplier	Cycle Time	Solution Time (*sec.*)	Gap
30	25%	0,1	1,4	160	Timeout (3 hr)	19,40%
			1,8	204	Timeout (3 hr)	17,92%
		0,25	1,4	160	Timeout (3 hr)	19,40%
			1,8	204	Timeout (3 hr)	17,92%
		0,5	1,4	162	Timeout (3 hr)	15,71%
			1,8	213	Timeout (3 hr)	18,33%
		0,75	1,4	181	Timeout (3 hr)	16,03%
			1,8	219	Timeout (3 hr)	11,17%
		0,9	1,4	178	Timeout (3 hr)	14,10%
			1,8	208	Timeout (3 hr)	5,58%
	50%	0,1	1,4	190	Timeout (3 hr)	36,69%
			1,8	221	Timeout (3 hr)	24,16%
		0,25	1,4	190	Timeout (3 hr)	31,03%
			1,8	216	Timeout (3 hr)	16,13%
		0,5	1,4	190	Timeout (3 hr)	31,03%
			1,8	216	Timeout (3 hr)	16,13%
		0,75	1,4	188	Timeout (3 hr)	18,99%
			1,8	226	Timeout (3 hr)	11,33%
		0,9	1,4	188	Timeout (3 hr)	18,99%
			1,8	207	Timeout (3 hr)	1,97%
	75%	0,1	1,4	196	Timeout (3 hr)	44,12%
			1,8	234	Timeout (3 hr)	30,00%
		0,25	1,4	183	Timeout (3 hr)	34,56%
			1,8	229	Timeout (3 hr)	19,90%
		0,5	1,4	184	Timeout (3 hr)	26,03%
			1,8	217	Timeout (3 hr)	13,61%
		0,75	1,4	189	Timeout (3 hr)	20,38%
			1,8	238	Timeout (3 hr)	13,88%
		0,9	1,4	186	Timeout (3 hr)	18,47%
			1,8	212	Timeout (3 hr)	1,44%
40	25%	0,1	1,4	200	Timeout (3 hr)	43,88%
			1,8	226	Timeout (3 hr)	31,40%
		0,25	1,4	206	Timeout (3 hr)	39,19%
			1,8	244	Timeout (3 hr)	37,85%
		0,5	1,4	203	Timeout (3 hr)	30,13%
			1,8	243	Timeout (3 hr)	35,00%
		0,75	1,4	203	Timeout (3 hr)	30,13%
			1,8	237	Timeout (3 hr)	22,80%
		0,9	1,4	214	Timeout (3 hr)	25,15%
			1,8	224	Timeout (3 hr)	16,06%
	50%	0,1	1,4	192	Timeout (3 hr)	39,13%
			1,8	221	Timeout (3 hr)	33,13%
		0,25	1,4	199	Timeout (3 hr)	38,19%
			1,8	241	Timeout (3 hr)	39,31%
		0,5	1,4	209	Timeout (3 hr)	39,33%
			1,8	229	Timeout (3 hr)	27,93%
		0,75	1,4	202	Timeout (3 hr)	30,32%
			1,8	254	Timeout (3 hr)	30,26%
		0,9	1,4	203	Timeout (3 hr)	27,67%
			1,8	228	Timeout (3 hr)	11,76%
	75%	0,1	1,4	205	Timeout (3 hr)	46,43%
			1,8	232	Timeout (3 hr)	39,76%
		0,25	1,4	209	Timeout (3 hr)	46,15%
			1,8	238	Timeout (3 hr)	31,49%
		0,5	1,4	211	Timeout (3 hr)	41,61%
			1,8	252	Timeout (3 hr)	35,48%
		0,75	1,4	212	Timeout (3 hr)	35,03%
			1,8	262	Timeout (3 hr)	35,05%
		0,9	1,4	225	Timeout (3 hr)	33,14%
			1,8	242	Timeout (3 hr)	15,79%

(Table B.9 Continued)

AD-RCPSP-II						
Set of Tasks (N)	**Multi-manned Task Perc.(%)**	**Order Strength**	**Number of Stations Multiplier**	**Cycle Time**	**Solution Time (*sec.*)**	**Gap**
50	25%	0,1	1,4	222	Timeout (3 hr)	60,87%
			1,8	274	Timeout (3 hr)	56,57%
		0,25	1,4	230	Timeout (3 hr)	63,12%
			1,8	285	Timeout (3 hr)	58,33%
		0,5	1,4	228	Timeout (3 hr)	55,10%
			1,8	292	Timeout (3 hr)	50,52%
		0,75	1,4	224	Timeout (3 hr)	43,59%
			1,8	298	Timeout (3 hr)	39,91%
		0,9	1,4	234	Timeout (3 hr)	39,29%
			1,8	294	Timeout (3 hr)	31,25%
	50%	0,1	1,4	242	Timeout (3 hr)	74,10%
			1,8	283	Timeout (3 hr)	60,80%
		0,25	1,4	233	Timeout (3 hr)	62,94%
			1,8	301	Timeout (3 hr)	65,38%
		0,5	1,4	226	Timeout (3 hr)	52,70%
			1,8	269	Timeout (3 hr)	43,09%
		0,75	1,4	222	Timeout (3 hr)	44,16%
			1,8	292	Timeout (3 hr)	47,47%
		0,9	1,4	232	Timeout (3 hr)	40,61%
			1,8	298	Timeout (3 hr)	43,27%
	75%	0,1	1,4	251	Timeout (3 hr)	87,31%
			1,8	275	Timeout (3 hr)	60,82%
		0,25	1,4	230	Timeout (3 hr)	64,29%
			1,8	294	Timeout (3 hr)	65,17%
		0,5	1,4	230	Timeout (3 hr)	58,62%
			1,8	294	Timeout (3 hr)	59,78%
		0,75	1,4	227	Timeout (3 hr)	48,37%
			1,8	302	Timeout (3 hr)	58,12%
		0,9	1,4	253	Timeout (3 hr)	49,70%
			1,8	288	Timeout (3 hr)	39,13%
60	25%	0,1	1,4	223	Timeout (3 hr)	61,59%
			1,8	303	Timeout (3 hr)	69,27%
		0,25	1,4	245	Timeout (3 hr)	70,14%
			1,8	307	Timeout (3 hr)	62,43%
		0,5	1,4	232	Timeout (3 hr)	55,70%
			1,8	299	Timeout (3 hr)	53,33%
		0,75	1,4	226	Timeout (3 hr)	43,95%
			1,8	296	Timeout (3 hr)	45,81%
		0,9	1,4	267	Timeout (3 hr)	56,14%
			1,8	304	Timeout (3 hr)	44,08%
	50%	0,1	1,4	243	Timeout (3 hr)	74,82%
			1,8	314	Timeout (3 hr)	74,44%
		0,25	1,4	249	Timeout (3 hr)	70,55%
			1,8	299	Timeout (3 hr)	58,20%
		0,5	1,4	255	Timeout (3 hr)	66,67%
			1,8	324	Timeout (3 hr)	61,19%
		0,75	1,4	242	Timeout (3 hr)	51,25%
			1,8	325	Timeout (3 hr)	55,50%
		0,9	1,4	247	Timeout (3 hr)	46,15%
			1,8	302	Timeout (3 hr)	40,47%
	75%	0,1	1,4	246	Timeout (3 hr)	75,71%
			1,8	314	Timeout (3 hr)	82,56%
		0,25	1,4	259	Timeout (3 hr)	78,62%
			1,8	311	Timeout (3 hr)	66,31%
		0,5	1,4	250	Timeout (3 hr)	65,56%
			1,8	318	Timeout (3 hr)	62,24%
		0,75	1,4	250	Timeout (3 hr)	58,23%
			1,8	313	Timeout (3 hr)	49,76%
		0,9	1,4	246	Timeout (3 hr)	50,00%
			1,8	326	Timeout (3 hr)	50,23%

Table B. 10: Results of AD-CP-II Formulation

AD-CP-II						
Set of Tasks (N)	Multi-manned Task Perc.(%)	Order Strength	Number of Stations Multiplier	Cycle Time	Solution Time (*sec.*)	Gap
30	25%	0,1	1,4	134	410	0,00%
			1,8	173	168	0,00%
		0,25	1,4	134	341	0,00%
			1,8	173	118	0,00%
		0,5	1,4	140	307	0,00%
			1,8	180	79	0,00%
		0,75	1,4	156	155	0,00%
			1,8	197	66	0,00%
		0,9	1,4	156	73	0,00%
			1,8	197	17	0,00%
	50%	0,1	1,4	139	651	0,00%
			1,8	178	248	0,00%
		0,25	1,4	145	389	0,00%
			1,8	186	132	0,00%
		0,5	1,4	145	325	0,00%
			1,8	186	88	0,00%
		0,75	1,4	158	199	0,00%
			1,8	203	75	0,00%
		0,9	1,4	158	95	0,00%
			1,8	203	28	0,00%
	75%	0,1	1,4	136	889	0,00%
			1,8	180	419	0,00%
		0,25	1,4	136	492	0,00%
			1,8	191	381	0,00%
		0,5	1,4	146	404	0,00%
			1,8	191	269	0,00%
		0,75	1,4	157	308	0,00%
			1,8	209	235	0,00%
		0,9	1,4	157	273	0,00%
			1,8	209	186	0,00%
40	25%	0,1	1,4	139	2298	0,00%
			1,8	172	1805	0,00%
		0,25	1,4	148	1973	0,00%
			1,8	177	1725	0,00%
		0,5	1,4	156	1867	0,00%
			1,8	180	1630	0,00%
		0,75	1,4	156	1777	0,00%
			1,8	193	1437	0,00%
		0,9	1,4	171	1426	0,00%
			1,8	193	1322	0,00%
	50%	0,1	1,4	138	2618	0,00%
			1,8	166	2322	0,00%
		0,25	1,4	144	2560	0,00%
			1,8	173	2268	0,00%
		0,5	1,4	150	2470	0,00%
			1,8	179	2215	0,00%
		0,75	1,4	155	2316	0,00%
			1,8	195	2157	0,00%
		0,9	1,4	159	2223	0,00%
			1,8	204	2030	0,00%
	75%	0,1	1,4	140	3330	0,00%
			1,8	166	2860	0,00%
		0,25	1,4	143	3174	0,00%
			1,8	181	2742	0,00%
		0,5	1,4	149	3220	0,00%
			1,8	186	2699	0,00%
		0,75	1,4	157	2810	0,00%
			1,8	194	2591	0,00%
		0,9	1,4	169	2668	0,00%
			1,8	209	2376	0,00%

(Table B.10 Continued)

AD-CP-II						
Set of Tasks (N)	**Multi-manned Task Perc.(%)**	**Order Strength**	**Number of Stations Multiplier**	**Cycle Time**	**Solution Time (*sec.*)**	**Gap**
50	25%	0,1	1,4	138	9453	0,00%
			1,8	175	8227	0,00%
		0,25	1,4	141	9064	0,00%
			1,8	180	8359	0,00%
		0,5	1,4	147	8761	0,00%
			1,8	194	7879	0,00%
		0,75	1,4	156	7931	0,00%
			1,8	213	7381	0,00%
		0,9	1,4	168	7909	0,00%
			1,8	224	6676	0,00%
	50%	0,1	1,4	139	10628	0,00%
			1,8	176	8967	0,00%
		0,25	1,4	143	10157	0,00%
			1,8	182	8703	0,00%
		0,5	1,4	148	9948	0,00%
			1,8	188	8591	0,00%
		0,75	1,4	154	9761	0,00%
			1,8	198	8406	0,00%
		0,9	1,4	165	9638	0,00%
			1,8	208	8224	0,00%
	75%	0,1	1,4	134	Timeout (3 hr)	0,00%
			1,8	171	9237	0,00%
		0,25	1,4	140	Timeout (3 hr)	0,00%
			1,8	178	9086	0,00%
		0,5	1,4	145	10533	0,00%
			1,8	184	8831	0,00%
		0,75	1,4	153	10038	0,00%
			1,8	191	8549	0,00%
		0,9	1,4	169	9891	0,00%
			1,8	207	8343	0,00%
60	25%	0,1	1,4	148	Timeout (3 hr)	7,25%
			1,8	179	Timeout (3 hr)	0,00%
		0,25	1,4	153	Timeout (3 hr)	6,25%
			1,8	189	Timeout (3 hr)	0,00%
		0,5	1,4	155	Timeout (3 hr)	4,03%
			1,8	195	Timeout (3 hr)	0,00%
		0,75	1,4	161	Timeout (3 hr)	2,55%
			1,8	203	9961	0,00%
		0,9	1,4	171	10549	0,00%
			1,8	211	9228	0,00%
	50%	0,1	1,4	152	Timeout (3 hr)	9,35%
			1,8	180	Timeout (3 hr)	0,00%
		0,25	1,4	158	Timeout (3 hr)	8,22%
			1,8	189	Timeout (3 hr)	0,00%
		0,5	1,4	164	Timeout (3 hr)	7,19%
			1,8	201	Timeout (3 hr)	0,00%
		0,75	1,4	169	Timeout (3 hr)	5,63%
			1,8	209	10157	0,00%
		0,9	1,4	176	Timeout (3 hr)	4,14%
			1,8	215	9524	0,00%
	75%	0,1	1,4	156	Timeout (3 hr)	11,43%
			1,8	178	Timeout (3 hr)	3,49%
		0,25	1,4	160	Timeout (3 hr)	10,34%
			1,8	190	Timeout (3 hr)	1,60%
		0,5	1,4	164	Timeout (3 hr)	8,61%
			1,8	196	Timeout (3 hr)	0,00%
		0,75	1,4	170	Timeout (3 hr)	7,59%
			1,8	209	10624	0,00%
		0,9	1,4	176	Timeout (3 hr)	7,32%
			1,8	217	9867	0,00%

Table B. 11: Results of RM-I Formulation

RM-I						
Set of Tasks (N)	Multi-manned Task Perc.(%)	Order Strength	Cycle Time Multiplier	Avg. Solution Time (*sec.*)	Avg. Gap	Avg. Performance
30	25%	0.1	1.4	8.24	1.42%	35.76%
30	25%	0.1	1.8	3.72	2.00%	30.91%
30	25%	0.25	1.4	6.71	1.10%	35.76%
30	25%	0.25	1.8	3.15	18.01%	52.73%
30	25%	0.5	1.4	6.39	1.99%	35.76%
30	25%	0.5	1.8	2.21	18.48%	52.73%
30	25%	0.75	1.4	4.07	1.41%	52.73%
30	25%	0.75	1.8	2.50	1.07%	52.73%
30	25%	0.9	1.4	2.73	12.00%	86.67%
30	25%	0.9	1.8	2.02	15.39%	74.55%
30	50%	0.1	1.4	10.93	1.31%	52.73%
30	50%	0.1	1.8	5.44	15.69%	74.55%
30	50%	0.25	1.4	5.83	11.71%	86.67%
30	50%	0.25	1.8	3.91	13.59%	96.36%
30	50%	0.5	1.4	4.85	11.45%	86.67%
30	50%	0.5	1.8	2.59	13.65%	96.36%
30	50%	0.75	1.4	4.57	10.69%	103.64%
30	50%	0.75	1.8	2.32	26.48%	118.18%
30	50%	0.9	1.4	2.24	20.17%	120.61%
30	50%	0.9	1.8	1.83	34.95%	161.82%
30	75%	0.1	1.4	20.00	19.29%	120.61%
30	75%	0.1	1.8	15.72	13.89%	96.36%
30	75%	0.25	1.4	15.04	20.15%	120.61%
30	75%	0.25	1.8	14.13	23.95%	140.00%
30	75%	0.5	1.4	14.76	26.25%	154.55%
30	75%	0.5	1.8	13.23	34.86%	161.82%
30	75%	0.75	1.4	13.61	31.81%	188.48%
30	75%	0.75	1.8	12.44	41.52%	205.45%
30	75%	0.9	1.4	12.72	39.47%	205.45%
30	75%	0.9	1.8	11.74	41.60%	205.45%
40	25%	0.1	1.4	33.44	1.81%	27.27%
40	25%	0.1	1.8	26.26	1.40%	30.91%
40	25%	0.25	1.4	29.80	1.62%	40.00%
40	25%	0.25	1.8	24.58	1.66%	30.91%
40	25%	0.5	1.4	29.57	10.70%	52.73%
40	25%	0.5	1.8	23.77	13.99%	47.27%
40	25%	0.75	1.4	25.13	10.78%	52.73%
40	25%	0.75	1.8	23.67	12.40%	63.64%
40	25%	0.9	1.4	23.27	18.33%	78.18%
40	25%	0.9	1.8	42.18	24.01%	80.00%
40	50%	0.1	1.4	41.82	9.85%	65.45%
40	50%	0.1	1.8	37.29	11.97%	80.00%
40	50%	0.25	1.4	39.47	9.17%	78.18%
40	50%	0.25	1.8	35.60	21.85%	96.36%
40	50%	0.5	1.4	39.49	17.20%	90.91%
40	50%	0.5	1.8	33.58	31.84%	112.73%
40	50%	0.75	1.4	35.78	16.09%	103.64%
40	50%	0.75	1.8	32.85	28.96%	129.09%
40	50%	0.9	1.4	33.24	29.94%	129.09%
40	50%	0.9	1.8	31.73	34.73%	161.82%
40	75%	0.1	1.4	50.12	15.34%	103.64%
40	75%	0.1	1.8	44.45	9.40%	112.73%
40	75%	0.25	1.4	48.03	21.35%	129.09%
40	75%	0.25	1.8	41.64	9.39%	129.09%
40	75%	0.5	1.4	46.62	28.48%	141.82%
40	75%	0.5	1.8	40.56	24.39%	161.82%
40	75%	0.75	1.4	42.48	32.33%	167.27%
40	75%	0.75	1.8	39.93	39.82%	194.55%
40	75%	0.9	1.4	40.05	37.26%	192.73%
40	75%	0.9	1.8	38.24	43.89%	227.27%

(Table B.11 Continued)

RM-I						
Set of Tasks (N)	Multi-manned Task Perc.(%)	Order Strength	Cycle Time Multiplier	Solution Time (*sec.*)	Gap	Performance
50	25%	0.1	1.4	138.53	1.57%	32.36%
			1.8	118.09	1.22%	30.91%
		0.25	1.4	127.05	8.83%	52.73%
			1.8	115.89	11.02%	44.00%
		0.5	1.4	123.52	8.84%	52.73%
			1.8	109.39	10.44%	57.09%
		0.75	1.4	115.58	1.39%	52.73%
			1.8	107.59	10.22%	70.18%
		0.9	1.4	108.39	8.07%	73.09%
			1.8	103.65	8.94%	83.27%
	50%	0.1	1.4	175.57	8.64%	62.91%
			1.8	150.32	10.14%	70.18%
		0.25	1.4	152.66	7.43%	73.09%
			1.8	142.07	9.34%	83.27%
		0.5	1.4	151.51	7.99%	73.09%
			1.8	139.15	17.22%	96.36%
		0.75	1.4	146.89	13.51%	93.45%
			1.8	137.25	16.03%	109.45%
		0.9	1.4	137.94	17.97%	113.82%
			1.8	134.19	29.86%	135.64%
	75%	0.1	1.4	191.63	12.85%	103.64%
			1.8	172.46	15.49%	109.45%
		0.25	1.4	178.72	18.11%	113.82%
			1.8	166.10	23.38%	122.55%
		0.5	1.4	178.01	17.00%	124.00%
			1.8	158.62	21.09%	135.64%
		0.75	1.4	162.74	16.50%	134.18%
			1.8	159.25	28.26%	148.73%
		0.9	1.4	157.54	20.65%	154.55%
			1.8	154.56	33.06%	174.91%
60	25%	0.1	1.4	533.36	1.18%	27.27%
			1.8	504.76	1.76%	30.91%
		0.25	1.4	523.46	1.40%	35.76%
			1.8	496.46	1.65%	41.82%
		0.5	1.4	518.46	1.95%	35.76%
			1.8	470.26	9.44%	52.73%
		0.75	1.4	501.35	7.38%	52.73%
			1.8	459.38	8.26%	63.64%
		0.9	1.4	490.00	12.14%	69.70%
			1.8	423.53	15.82%	74.55%
	50%	0.1	1.4	563.58	1.49%	52.73%
			1.8	528.41	1.19%	52.73%
		0.25	1.4	530.91	1.18%	61.21%
			1.8	517.26	7.80%	74.55%
		0.5	1.4	529.26	11.55%	86.67%
			1.8	512.28	13.55%	96.36%
		0.75	1.4	523.53	11.07%	95.15%
			1.8	510.10	13.69%	107.27%
		0.9	1.4	511.52	15.24%	112.12%
			1.8	505.18	24.63%	129.09%
	75%	0.1	1.4	593.16	10.71%	95.15%
			1.8	558.41	12.92%	107.27%
		0.25	1.4	578.25	15.36%	112.12%
			1.8	555.34	18.46%	129.09%
		0.5	1.4	572.15	18.88%	129.09%
			1.8	546.55	27.45%	161.82%
		0.75	1.4	554.30	21.92%	146.06%
			1.8	546.71	31.67%	183.64%
		0.9	1.4	546.53	25.78%	163.03%
			1.8	540.57	35.01%	205.45%

Table B. 12: Results of AD-GMM-I Formulation

AD-GMM-I						
Set of Tasks (N)	**Multi-manned Task Perc.(%)**	**Order Strength**	**Cycle Time Multiplier**	**Number of Stations Opened**	**Solution Time (*sec.*)**	**Gap**
30	25%	0,1	1,4	8	386	0%
			1,8	6	137	0%
		0,25	1,4	8	289	0%
			1,8	6	121	0%
		0,5	1,4	8	248	0%
			1,8	6	87	0%
		0,75	1,4	9	176	0%
			1,8	7	48	0%
		0,9	1,4	10	101	0%
			1,8	7	25	0%
	50%	0,1	1,4	9	651	0%
			1,8	7	307	0%
		0,25	1,4	10	541	0%
			1,8	8	253	0%
		0,5	1,4	10	470	0%
			1,8	8	178	0%
		0,75	1,4	11	406	0%
			1,8	8	144	0%
		0,9	1,4	11	304	0%
			1,8	9	136	0%
	75%	0,1	1,4	11	1268	0%
			1,8	8	1039	0%
		0,25	1,4	11	1036	0%
			1,8	9	771	0%
		0,5	1,4	12	861	0%
			1,8	9	414	0%
		0,75	1,4	13	797	0%
			1,8	10	523	0%
		0,9	1,4	13	668	0%
			1,8	10	309	0%
40	25%	0,1	1,4	10	717	0%
			1,8	8	274	0%
		0,25	1,4	11	575	0%
			1,8	8	229	0%
		0,5	1,4	11	440	0%
			1,8	8	152	0%
		0,75	1,4	11	275	0%
			1,8	9	88	0%
		0,9	1,4	12	163	0%
			1,8	9	43	0%
	50%	0,1	1,4	12	1095	0%
			1,8	10	481	0%
		0,25	1,4	13	900	0%
			1,8	10	441	0%
		0,5	1,4	13	810	0%
			1,8	10	309	0%
		0,75	1,4	14	690	0%
			1,8	11	230	0%
		0,9	1,4	14	505	0%
			1,8	12	222	0%
	75%	0,1	1,4	14	2178	0%
			1,8	12	1756	0%
		0,25	1,4	15	1809	0%
			1,8	13	1287	0%
		0,5	1,4	15	1447	0%
			1,8	13	717	0%
		0,75	1,4	16	1413	0%
			1,8	13	901	0%
		0,9	1,4	17	1116	0%
			1,8	14	528	0%

(Table B.12 Continued)

AD-GMM-I						
Set of Tasks (N)	**Multi-manned Task Perc.(%)**	**Order Strength**	**Cycle Time Multiplier**	**Number of Stations Opened**	**Solution Time (*sec.*)**	**Gap**
50	25%	0,1	1,4	13	1108	0%
			1,8	10	387	0%
		0,25	1,4	14	841	0%
			1,8	10	342	0%
		0,5	1,4	14	717	0%
			1,8	11	239	0%
		0,75	1,4	15	507	0%
			1,8	12	149	0%
		0,9	1,4	16	280	0%
			1,8	13	77	0%
	50%	0,1	1,4	15	1903	0%
			1,8	12	892	0%
		0,25	1,4	16	1562	0%
			1,8	13	687	0%
		0,5	1,4	16	1378	0%
			1,8	13	520	0%
		0,75	1,4	17	1166	0%
			1,8	14	397	0%
		0,9	1,4	18	871	0%
			1,8	14	384	0%
	75%	0,1	1,4	18	5092	0%
			1,8	14	2995	0%
		0,25	1,4	18	3024	0%
			1,8	14	2258	0%
		0,5	1,4	19	2503	0%
			1,8	15	1220	0%
		0,75	1,4	20	2290	0%
			1,8	15	1564	0%
		0,9	1,4	21	1935	0%
			1,8	16	889	0%
60	25%	0,1	1,4	15	1652	0%
			1,8	12	609	0%
		0,25	1,4	16	1242	0%
			1,8	13	503	0%
		0,5	1,4	16	1078	0%
			1,8	13	354	0%
		0,75	1,4	17	729	0%
			1,8	14	202	0%
		0,9	1,4	18	409	0%
			1,8	14	108	0%
	50%	0,1	1,4	18	2792	0%
			1,8	14	1329	0%
		0,25	1,4	19	2326	0%
			1,8	15	1081	0%
		0,5	1,4	20	2093	0%
			1,8	16	816	0%
		0,75	1,4	21	1699	0%
			1,8	17	547	0%
		0,9	1,4	22	1272	0%
			1,8	17	562	0%
	75%	0,1	1,4	21	8135	0%
			1,8	17	6922	0%
		0,25	1,4	22	7391	0%
			1,8	18	3312	0%
		0,5	1,4	23	3700	0%
			1,8	19	1766	0%
		0,75	1,4	24	3438	0%
			1,8	20	2221	0%
		0,9	1,4	25	2841	0%
			1,8	21	1351	0%

Table B. 13: Results of RM-II Formulation

RM-II							
Set of Tasks (N)	Multi-manned Task Perc.(%)	Order Strength	Number of Stations Multiplier	Cycle Time	Solution Time (sec.)	Gap	Performance
30	25%	0.1	1.4	164	478	22.39%	59.03%
			1.8	192	199	10.98%	39.64%
		0.25	1.4	172	383	28.36%	66.79%
			1.8	202	137	16.76%	46.91%
		0.5	1.4	144	352	2.86%	39.64%
			1.8	196	94	8.89%	42.55%
		0.75	1.4	194	175	24.36%	88.12%
			1.8	251	74	27.41%	82.55%
		0.9	1.4	218	86	39.74%	111.39%
			1.8	237	18	20.30%	72.36%
	50%	0.1	1.4	158	714	13.67%	53.21%
			1.8	192	278	7.87%	39.64%
		0.25	1.4	165	349	13.79%	60.00%
			1.8	218	148	17.20%	58.55%
		0.5	1.4	172	244	18.62%	66.79%
			1.8	218	96	17.20%	58.55%
		0.75	1.4	190	215	20.25%	84.24%
			1.8	245	85	20.69%	78.18%
		0.9	1.4	293	65	85.44%	184.12%
			1.8	283	31	39.41%	105.82%
	75%	0.1	1.4	152	1512	11.76%	47.39%
			1.8	218	1093	21.11%	58.55%
		0.25	1.4	163	1099	19.85%	58.06%
			1.8	228	962	19.37%	65.82%
		0.5	1.4	196	1064	34.25%	90.06%
			1.8	255	893	33.51%	85.45%
		0.75	1.4	203	959	29.30%	96.85%
			1.8	237	857	13.40%	72.36%
		0.9	1.4	264	895	68.15%	156.00%
			1.8	304	794	45.45%	121.09%
40	25%	0.1	1.4	175	2567	25.90%	59.09%
			1.8	234	1905	36.05%	70.18%
		0.25	1.4	190	2279	28.38%	72.73%
			1.8	229	1893	29.38%	66.55%
		0.5	1.4	234	2138	50.00%	112.73%
			1.8	282	1814	56.67%	105.09%
		0.75	1.4	248	1924	58.97%	125.45%
			1.8	263	1691	36.27%	91.27%
		0.9	1.4	304	1702	77.78%	176.36%
			1.8	337	3221	74.61%	145.09%
	50%	0.1	1.4	176	3071	27.54%	60.00%
			1.8	217	2659	30.72%	57.82%
		0.25	1.4	198	2962	37.50%	80.00%
			1.8	214	2713	23.70%	55.64%
		0.5	1.4	208	3011	38.67%	89.09%
			1.8	236	2452	31.84%	71.64%
		0.75	1.4	248	2641	60.00%	125.45%
			1.8	314	2536	61.03%	128.36%
		0.9	1.4	288	2536	81.13%	161.82%
			1.8	334	2303	63.73%	142.91%
	75%	0.1	1.4	174	3727	24.29%	58.18%
			1.8	202	3244	21.69%	46.91%
		0.25	1.4	192	3657	34.27%	74.55%
			1.8	264	3151	45.86%	92.00%
		0.5	1.4	203	3653	36.24%	84.55%
			1.8	256	3059	37.63%	86.18%
		0.75	1.4	273	3178	73.89%	148.18%
			1.8	309	2997	59.28%	124.73%
		0.9	1.4	308	2978	82.25%	180.00%
			1.8	344	2769	64.59%	150.18%

(Table B.13 Continued)

RM-II							
Set of Tasks (N)	Multi-manned Task Perc.(%)	Order Strength	Number of Stations Multiplier	Cycle Time	Solution Time (*sec.*)	Gap	Performance
50	25%	0.1	1.4	152	10414	10.14%	43.71%
			1.8	198	9020	13.14%	44.00%
		0.25	1.4	173	9932	22.70%	63.56%
			1.8	208	9211	15.56%	51.27%
		0.5	1.4	197	9596	34.01%	86.25%
			1.8	222	8618	14.43%	61.45%
		0.75	1.4	207	8683	32.69%	95.71%
			1.8	256	8109	20.19%	86.18%
		0.9	1.4	268	8139	59.52%	153.38%
			1.8	308	7618	37.50%	124.00%
	50%	0.1	1.4	158	Timeout (3 hr)	13.67%	49.38%
			1.8	204	Timeout (3 hr)	15.91%	48.36%
		0.25	1.4	168	Timeout (3 hr)	17.48%	58.84%
			1.8	233	Timeout (3 hr)	28.02%	69.45%
		0.5	1.4	192	Timeout (3 hr)	29.73%	81.53%
			1.8	264	10311	40.43%	92.00%
		0.75	1.4	228	Timeout (3 hr)	48.05%	115.56%
			1.8	308	10718	55.56%	124.00%
		0.9	1.4	254	10295	53.94%	140.15%
			1.8	334	10649	60.58%	142.91%
	75%	0.1	1.4	168	Timeout (3 hr)	25.37%	58.84%
			1.8	217	Timeout (3 hr)	26.90%	57.82%
		0.25	1.4	186	Timeout (3 hr)	32.86%	75.85%
			1.8	234	Timeout (3 hr)	31.46%	70.18%
		0.5	1.4	228	Timeout (3 hr)	57.24%	115.56%
			1.8	264	Timeout (3 hr)	43.48%	92.00%
		0.75	1.4	232	Timeout (3 hr)	51.63%	119.35%
			1.8	305	Timeout (3 hr)	59.69%	121.82%
		0.9	1.4	281	Timeout (3 hr)	66.27%	165.67%
			1.8	346	Timeout (3 hr)	67.15%	151.64%
60	25%	0.1	1.4	168	Timeout (3 hr)	52.17%	52.73%
			1.8	210	Timeout (3 hr)	48.04%	52.73%
		0.25	1.4	196	Timeout (3 hr)	49.31%	78.18%
			1.8	225	Timeout (3 hr)	43.92%	63.64%
		0.5	1.4	214	Timeout (3 hr)	41.61%	94.55%
			1.8	236	Timeout (3 hr)	35.90%	71.64%
		0.75	1.4	224	Timeout (3 hr)	33.76%	103.64%
			1.8	287	Timeout (3 hr)	31.03%	108.73%
		0.9	1.4	259	Timeout (3 hr)	34.50%	135.45%
			1.8	348	Timeout (3 hr)	30.81%	153.09%
	50%	0.1	1.4	167	Timeout (3 hr)	58.99%	51.82%
			1.8	224	Timeout (3 hr)	52.78%	62.91%
		0.25	1.4	205	Timeout (3 hr)	50.68%	86.36%
			1.8	248	Timeout (3 hr)	44.97%	80.36%
		0.5	1.4	236	Timeout (3 hr)	45.10%	114.55%
			1.8	275	Timeout (3 hr)	38.31%	100.00%
		0.75	1.4	290	Timeout (3 hr)	38.75%	163.64%
			1.8	344	Timeout (3 hr)	33.97%	150.18%
		0.9	1.4	346	Timeout (3 hr)	36.69%	214.55%
			1.8	388	Timeout (3 hr)	30.23%	182.18%
	75%	0.1	1.4	204	Timeout (3 hr)	65.71%	85.45%
			1.8	246	Timeout (3 hr)	57.56%	78.91%
		0.25	1.4	212	Timeout (3 hr)	54.48%	92.73%
			1.8	263	Timeout (3 hr)	44.92%	91.27%
		0.5	1.4	229	Timeout (3 hr)	48.34%	108.18%
			1.8	292	Timeout (3 hr)	43.88%	112.36%
		0.75	1.4	255	Timeout (3 hr)	39.24%	131.82%
			1.8	337	Timeout (3 hr)	35.89%	145.09%
		0.9	1.4	269	Timeout (3 hr)	36.59%	144.55%
			1.8	342	Timeout (3 hr)	30.88%	148.73%

Table B. 14: Results of AD-GMM-II Formulation

AD-GMM-II						
Set of Tasks (N)	**Multi-manned Task Perc.(%)**	**Order Strength**	**Number of Stations Multiplier**	**Cycle Time**	**Solution Time (*sec.*)**	**Gap**
30	25%	0,1	1,4	134	Timeout (3 hr)	0,00%
			1,8	173	Timeout (3 hr)	0,00%
		0,25	1,4	134	Timeout (3 hr)	0,00%
			1,8	173	Timeout (3 hr)	0,00%
		0,5	1,4	140	Timeout (3 hr)	0,00%
			1,8	180	Timeout (3 hr)	0,00%
		0,75	1,4	156	Timeout (3 hr)	0,00%
			1,8	197	Timeout (3 hr)	0,00%
		0,9	1,4	156	Timeout (3 hr)	0,00%
			1,8	197	Timeout (3 hr)	0,00%
	50%	0,1	1,4	148	Timeout (3 hr)	6,47%
			1,8	178	Timeout (3 hr)	0,00%
		0,25	1,4	152	Timeout (3 hr)	4,83%
			1,8	186	Timeout (3 hr)	0,00%
		0,5	1,4	145	Timeout (3 hr)	0,00%
			1,8	186	Timeout (3 hr)	0,00%
		0,75	1,4	158	Timeout (3 hr)	0,00%
			1,8	203	Timeout (3 hr)	0,00%
		0,9	1,4	158	Timeout (3 hr)	0,00%
			1,8	203	Timeout (3 hr)	0,00%
	75%	0,1	1,4	156	Timeout (3 hr)	14,71%
			1,8	184	Timeout (3 hr)	2,22%
		0,25	1,4	145	Timeout (3 hr)	6,62%
			1,8	194	Timeout (3 hr)	1,57%
		0,5	1,4	146	Timeout (3 hr)	0,00%
			1,8	191	Timeout (3 hr)	0,00%
		0,75	1,4	157	Timeout (3 hr)	0,00%
			1,8	209	Timeout (3 hr)	0,00%
		0,9	1,4	157	Timeout (3 hr)	0,00%
			1,8	209	Timeout (3 hr)	0,00%
40	25%	0,1	1,4	164	Timeout (3 hr)	17,99%
			1,8	180	Timeout (3 hr)	4,65%
		0,25	1,4	165	Timeout (3 hr)	11,49%
			1,8	181	Timeout (3 hr)	2,26%
		0,5	1,4	169	Timeout (3 hr)	8,33%
			1,8	182	Timeout (3 hr)	1,11%
		0,75	1,4	164	Timeout (3 hr)	5,13%
			1,8	193	Timeout (3 hr)	0,00%
		0,9	1,4	177	Timeout (3 hr)	3,51%
			1,8	193	Timeout (3 hr)	0,00%
	50%	0,1	1,4	170	Timeout (3 hr)	23,19%
			1,8	180	Timeout (3 hr)	8,43%
		0,25	1,4	174	Timeout (3 hr)	20,83%
			1,8	180	Timeout (3 hr)	4,05%
		0,5	1,4	177	Timeout (3 hr)	18,00%
			1,8	184	Timeout (3 hr)	2,79%
		0,75	1,4	180	Timeout (3 hr)	16,13%
			1,8	200	Timeout (3 hr)	2,56%
		0,9	1,4	180	Timeout (3 hr)	13,21%
			1,8	207	Timeout (3 hr)	1,47%
	75%	0,1	1,4	180	Timeout (3 hr)	28,57%
			1,8	185	Timeout (3 hr)	11,45%
		0,25	1,4	180	Timeout (3 hr)	25,87%
			1,8	191	Timeout (3 hr)	5,52%
		0,5	1,4	179	Timeout (3 hr)	20,13%
			1,8	194	Timeout (3 hr)	4,30%
		0,75	1,4	186	Timeout (3 hr)	18,47%
			1,8	198	Timeout (3 hr)	2,06%
		0,9	1,4	187	Timeout (3 hr)	10,65%
			1,8	212	Timeout (3 hr)	1,44%

(Table B.14 Continued)

AD-GMM-II						
Set of Tasks (N)	Multi-manned Task Perc.(%)	Order Strength	Number of Stations Multiplier	Cycle Time	Solution Time (*sec.*)	Gap
50	25%	0,1	1,4	191	Timeout (3 hr)	38,41%
			1,8	200	Timeout (3 hr)	14,29%
		0,25	1,4	191	Timeout (3 hr)	35,46%
			1,8	197	Timeout (3 hr)	9,44%
		0,5	1,4	192	Timeout (3 hr)	30,61%
			1,8	209	Timeout (3 hr)	7,73%
		0,75	1,4	199	Timeout (3 hr)	27,56%
			1,8	224	Timeout (3 hr)	5,16%
		0,9	1,4	210	Timeout (3 hr)	25,00%
			1,8	235	Timeout (3 hr)	4,91%
	50%	0,1	1,4	202	Timeout (3 hr)	45,32%
			1,8	210	Timeout (3 hr)	19,32%
		0,25	1,4	203	Timeout (3 hr)	41,96%
			1,8	206	Timeout (3 hr)	13,19%
		0,5	1,4	200	Timeout (3 hr)	35,14%
			1,8	207	Timeout (3 hr)	10,11%
		0,75	1,4	201	Timeout (3 hr)	30,52%
			1,8	214	Timeout (3 hr)	8,08%
		0,9	1,4	208	Timeout (3 hr)	26,06%
			1,8	223	Timeout (3 hr)	7,21%
	75%	0,1	1,4	201	Timeout (3 hr)	50,00%
			1,8	203	Timeout (3 hr)	18,71%
		0,25	1,4	202	Timeout (3 hr)	44,29%
			1,8	204	Timeout (3 hr)	14,61%
		0,5	1,4	198	Timeout (3 hr)	36,55%
			1,8	207	Timeout (3 hr)	12,50%
		0,75	1,4	202	Timeout (3 hr)	32,03%
			1,8	212	Timeout (3 hr)	10,99%
		0,9	1,4	215	Timeout (3 hr)	27,22%
			1,8	223	Timeout (3 hr)	7,73%
60	25%	0,1	1,4	197	Timeout (3 hr)	42,75%
			1,8	213	Timeout (3 hr)	18,99%
		0,25	1,4	201	Timeout (3 hr)	39,58%
			1,8	212	Timeout (3 hr)	12,17%
		0,5	1,4	200	Timeout (3 hr)	34,23%
			1,8	214	Timeout (3 hr)	9,74%
		0,75	1,4	203	Timeout (3 hr)	29,30%
			1,8	217	Timeout (3 hr)	6,90%
		0,9	1,4	215	Timeout (3 hr)	25,73%
			1,8	224	Timeout (3 hr)	6,16%
	50%	0,1	1,4	213	Timeout (3 hr)	53,24%
			1,8	223	Timeout (3 hr)	23,89%
		0,25	1,4	214	Timeout (3 hr)	46,58%
			1,8	217	Timeout (3 hr)	14,81%
		0,5	1,4	217	Timeout (3 hr)	41,83%
			1,8	225	Timeout (3 hr)	11,94%
		0,75	1,4	218	Timeout (3 hr)	36,25%
			1,8	227	Timeout (3 hr)	8,61%
		0,9	1,4	222	Timeout (3 hr)	31,36%
			1,8	231	Timeout (3 hr)	7,44%
	75%	0,1	1,4	222	Timeout (3 hr)	58,57%
			1,8	214	Timeout (3 hr)	24,42%
		0,25	1,4	225	Timeout (3 hr)	55,17%
			1,8	222	Timeout (3 hr)	18,72%
		0,5	1,4	222	Timeout (3 hr)	47,02%
			1,8	223	Timeout (3 hr)	13,78%
		0,75	1,4	222	Timeout (3 hr)	40,51%
			1,8	227	Timeout (3 hr)	8,61%
		0,9	1,4	225	Timeout (3 hr)	37,20%
			1,8	235	Timeout (3 hr)	8,29%

Table B. 15: Comparison of Solution Times for Type-I

Solution Time (sec.)							
Set of Tasks (N)	Multi-manned	Order Strength	Cycle Time	AD-GMM-I	RM-I	RCPSP-I	AD-CP-I
30	25%	0.1	1.4	386	6.35	Timeout (3 hr)	754
			1.8	137	2.59	Timeout (3 hr)	249
		0.25	1.4	289	4.89	Timeout (3 hr)	550
			1.8	121	1.84	Timeout (3 hr)	226
		0.5	1.4	248	4.67	Timeout (3 hr)	405
			1.8	87	1.16	Timeout (3 hr)	118
		0.75	1.4	176	2.28	Timeout (3 hr)	259
			1.8	48	0.92	Timeout (3 hr)	46
		0.9	1.4	101	1.08	Timeout (3 hr)	96
			1.8	25	0.23	Timeout (3 hr)	5
	50%	0.1	1.4	651	8.95	Timeout (3 hr)	1241
			1.8	307	3.69	Timeout (3 hr)	590
		0.25	1.4	541	4.34	Timeout (3 hr)	926
			1.8	253	1.98	Timeout (3 hr)	418
		0.5	1.4	470	3.23	Timeout (3 hr)	747
			1.8	178	1.27	Timeout (3 hr)	258
		0.75	1.4	406	2.67	Timeout (3 hr)	574
			1.8	144	1.07	Timeout (3 hr)	158
		0.9	1.4	304	0.85	Timeout (3 hr)	346
			1.8	136	0.39	Timeout (3 hr)	111
	75%	0.1	1.4	1268	18.65	Timeout (3 hr)	2269
			1.8	1039	14.12	Timeout (3 hr)	1858
		0.25	1.4	1036	13.89	Timeout (3 hr)	1662
			1.8	771	12.89	Timeout (3 hr)	1230
		0.5	1.4	861	13.35	Timeout (3 hr)	1237
			1.8	414	11.91	Timeout (3 hr)	570
		0.75	1.4	797	12.09	Timeout (3 hr)	983
			1.8	523	10.99	Timeout (3 hr)	639
		0.9	1.4	668	11.04	Timeout (3 hr)	719
			1.8	309	10.53	Timeout (3 hr)	290
40	25%	0.1	1.4	717	31.54	Timeout (3 hr)	1405
			1.8	274	24.89	Timeout (3 hr)	532
		0.25	1.4	575	28.73	Timeout (3 hr)	1081
			1.8	229	23.49	Timeout (3 hr)	419
		0.5	1.4	440	28.09	Timeout (3 hr)	735
			1.8	152	22.27	Timeout (3 hr)	227
		0.75	1.4	275	24.1	Timeout (3 hr)	393
			1.8	88	21.84	Timeout (3 hr)	84
		0.9	1.4	163	21.89	Timeout (3 hr)	164
			1.8	43	40.51	Timeout (3 hr)	8
	50%	0.1	1.4	1095	40.68	Timeout (3 hr)	2108
			1.8	481	35.67	Timeout (3 hr)	900
		0.25	1.4	900	38.43	Timeout (3 hr)	1555
			1.8	441	33.67	Timeout (3 hr)	766
		0.5	1.4	810	37.86	Timeout (3 hr)	1257
			1.8	309	32.48	Timeout (3 hr)	450
		0.75	1.4	690	33.98	Timeout (3 hr)	953
			1.8	230	31.79	Timeout (3 hr)	281
		0.9	1.4	505	32.07	Timeout (3 hr)	564
			1.8	222	30.39	Timeout (3 hr)	184
	75%	0.1	1.4	2178	48.82	Timeout (3 hr)	3825
			1.8	1756	42.80	Timeout (3 hr)	3072
		0.25	1.4	1809	46.12	Timeout (3 hr)	2829
			1.8	1287	40.40	Timeout (3 hr)	2067
		0.5	1.4	1447	45.43	Timeout (3 hr)	2110
			1.8	717	38.98	Timeout (3 hr)	1023
		0.75	1.4	1413	40.78	Timeout (3 hr)	1642
			1.8	901	38.15	Timeout (3 hr)	1048
		0.9	1.4	1116	38.48	Timeout (3 hr)	1178
			1.8	528	36.47	Timeout (3 hr)	476

(Table B.15 Continued)

Solution Time (sec.)							
Set of Tasks (N)	**Multi-manned**	**Order Strength**	**Cycle Time**	**AD-GMM-I**	**RM-I**	**RCPSP-I**	**AD-CP-I**
50	25%	0.1	1.4	1108	136.89	Timeout (3 hr)	2185
			1.8	387	116.37	Timeout (3 hr)	762
		0.25	1.4	841	125.24	Timeout (3 hr)	1519
			1.8	342	114.56	Timeout (3 hr)	616
		0.5	1.4	717	122.37	Timeout (3 hr)	1214
			1.8	239	107.56	Timeout (3 hr)	378
		0.75	1.4	507	114.08	Timeout (3 hr)	757
			1.8	149	105.68	Timeout (3 hr)	177
		0.9	1.4	280	106.97	Timeout (3 hr)	330
			1.8	77	101.77	Timeout (3 hr)	27
	50%	0.1	1.4	1903	173.65	Timeout (3 hr)	3578
			1.8	892	148.64	Timeout (3 hr)	1695
		0.25	1.4	1562	150.87	Timeout (3 hr)	2612
			1.8	687	140.98	Timeout (3 hr)	1085
		0.5	1.4	1378	149.8	Timeout (3 hr)	2115
			1.8	520	137.91	Timeout (3 hr)	790
		0.75	1.4	1166	145.61	Timeout (3 hr)	1613
			1.8	397	135.97	Timeout (3 hr)	457
		0.9	1.4	871	136.81	Timeout (3 hr)	981
			1.8	384	132.78	Timeout (3 hr)	366
	75%	0.1	1.4	5092	190.39	Timeout (3 hr)	9188
			1.8	2995	170.73	Timeout (3 hr)	5396
		0.25	1.4	3024	177.64	Timeout (3 hr)	4876
			1.8	2258	164.85	Timeout (3 hr)	3634
		0.5	1.4	2503	176.37	Timeout (3 hr)	3709
			1.8	1220	157.08	Timeout (3 hr)	1763
		0.75	1.4	2290	161.18	Timeout (3 hr)	2872
			1.8	1564	157.38	Timeout (3 hr)	1923
		0.9	1.4	1935	156.28	Timeout (3 hr)	2155
			1.8	889	153.01	Timeout (3 hr)	831
60	25%	0.1	1.4	1652	531.94	Timeout (3 hr)	3267
			1.8	609	503.18	Timeout (3 hr)	1982
		0.25	1.4	1242	522.04	Timeout (3 hr)	2354
			1.8	503	495.39	Timeout (3 hr)	919
		0.5	1.4	1078	517.13	Timeout (3 hr)	1786
			1.8	354	469.04	Timeout (3 hr)	550
		0.75	1.4	729	499.72	Timeout (3 hr)	1059
			1.8	202	457.64	Timeout (3 hr)	238
		0.9	1.4	409	488.34	Timeout (3 hr)	453
			1.8	108	421.99	Timeout (3 hr)	56
	50%	0.1	1.4	2792	562.3	Timeout (3 hr)	5364
			1.8	1329	527.38	Timeout (3 hr)	2253
		0.25	1.4	2326	528.92	Timeout (3 hr)	3961
			1.8	1081	515.43	Timeout (3 hr)	1816
		0.5	1.4	2093	527.88	Timeout (3 hr)	3280
			1.8	816	511.14	Timeout (3 hr)	1240
		0.75	1.4	1699	521.92	Timeout (3 hr)	2390
			1.8	547	508.45	Timeout (3 hr)	682
		0.9	1.4	1272	509.68	Timeout (3 hr)	1583
			1.8	562	503.91	Timeout (3 hr)	550
	75%	0.1	1.4	8135	591.97	Timeout (3 hr)	Timeout (3 hr)
			1.8	6922	557.34	Timeout (3 hr)	7735
		0.25	1.4	7391	577.18	Timeout (3 hr)	Timeout (3 hr)
			1.8	3312	554.04	Timeout (3 hr)	5367
		0.5	1.4	3700	571.11	Timeout (3 hr)	5516
			1.8	1766	545.03	Timeout (3 hr)	2637
		0.75	1.4	3438	552.67	Timeout (3 hr)	3965
			1.8	2221	544.82	Timeout (3 hr)	2845
		0.9	1.4	2841	545.34	Timeout (3 hr)	3223
			1.8	1351	538.69	Timeout (3 hr)	1454

Table B. 16: Comparison of Gap Percentages for Type-I

Set of Tasks (N)	Multi-manned	Order Strength	Cycle Time	AD-GMM-I	RM-I	RCPSP-I	AD-CP-I
30	25%	0.1	1.4	0%	0.00%	87.50%	0%
30	25%	0.1	1.8	0%	0.00%	50.00%	0%
30	25%	0.25	1.4	0%	0.00%	62.50%	0%
30	25%	0.25	1.8	0%	16.67%	16.67%	0%
30	25%	0.5	1.4	0%	0.00%	50.00%	0%
30	25%	0.5	1.8	0%	16.67%	16.67%	0%
30	25%	0.75	1.4	0%	0.00%	22.22%	0%
30	25%	0.75	1.8	0%	0.00%	0.00%	0%
30	25%	0.9	1.4	0%	10.00%	10.00%	0%
30	25%	0.9	1.8	0%	14.29%	0.00%	0%
30	50%	0.1	1.4	0%	0.00%	88.89%	0%
30	50%	0.1	1.8	0%	14.29%	42.86%	0%
30	50%	0.25	1.4	0%	10.00%	70.00%	0%
30	50%	0.25	1.8	0%	12.50%	25.00%	0%
30	50%	0.5	1.4	0%	10.00%	40.00%	0%
30	50%	0.5	1.8	0%	12.50%	12.50%	0%
30	50%	0.75	1.4	0%	9.09%	18.18%	0%
30	50%	0.75	1.8	0%	25.00%	12.50%	0%
30	50%	0.9	1.4	0%	18.18%	9.09%	0%
30	50%	0.9	1.8	0%	33.33%	0.00%	0%
30	75%	0.1	1.4	0%	18.18%	90.91%	0%
30	75%	0.1	1.8	0%	12.50%	50.00%	0%
30	75%	0.25	1.4	0%	18.18%	63.64%	0%
30	75%	0.25	1.8	0%	22.22%	22.22%	0%
30	75%	0.5	1.4	0%	25.00%	41.67%	0%
30	75%	0.5	1.8	0%	33.33%	11.11%	0%
30	75%	0.75	1.4	0%	30.77%	15.38%	0%
30	75%	0.75	1.8	0%	40.00%	10.00%	0%
30	75%	0.9	1.4	0%	38.46%	7.69%	0%
30	75%	0.9	1.8	0%	40.00%	0.00%	0%
40	25%	0.1	1.4	0%	0.00%	90.00%	0%
40	25%	0.1	1.8	0%	0.00%	50.00%	0%
40	25%	0.25	1.4	0%	0.00%	63.64%	0%
40	25%	0.25	1.8	0%	0.00%	25.00%	0%
40	25%	0.5	1.4	0%	9.09%	45.45%	0%
40	25%	0.5	1.8	0%	12.50%	12.50%	0%
40	25%	0.75	1.4	0%	9.09%	18.18%	0%
40	25%	0.75	1.8	0%	11.11%	11.11%	0%
40	25%	0.9	1.4	0%	16.67%	0.00%	0%
40	25%	0.9	1.8	0%	22.22%	0.00%	0%
40	50%	0.1	1.4	0%	8.33%	91.67%	0%
40	50%	0.1	1.8	0%	10.00%	50.00%	0%
40	50%	0.25	1.4	0%	7.69%	61.54%	0%
40	50%	0.25	1.8	0%	20.00%	20.00%	0%
40	50%	0.5	1.4	0%	15.38%	30.77%	0%
40	50%	0.5	1.8	0%	30.00%	10.00%	0%
40	50%	0.75	1.4	0%	14.29%	7.14%	0%
40	50%	0.75	1.8	0%	27.27%	9.09%	0%
40	50%	0.9	1.4	0%	28.57%	0.00%	0%
40	50%	0.9	1.8	0%	33.33%	0.00%	0%
40	75%	0.1	1.4	0%	14.29%	92.86%	0%
40	75%	0.1	1.8	0%	8.33%	50.00%	0%
40	75%	0.25	1.4	0%	20.00%	66.67%	0%
40	75%	0.25	1.8	0%	7.69%	23.08%	0%
40	75%	0.5	1.4	0%	26.67%	40.00%	0%
40	75%	0.5	1.8	0%	23.08%	15.38%	0%
40	75%	0.75	1.4	0%	31.25%	12.50%	0%
40	75%	0.75	1.8	0%	38.46%	7.69%	0%
40	75%	0.9	1.4	0%	35.29%	5.88%	0%
40	75%	0.9	1.8	0%	42.86%	0.00%	0%

(Table B.16 Continued)

Gap Percentage							
Set of Tasks (N)	Multi-manned	Order Strength	Cycle Time	AD-GMM-I	RM-I	RCPSP-I	AD-CP-I
50	25%	0.1	1.4	0%	0.00%	92.31%	0%
			1.8	0%	0.00%	50.00%	0%
		0.25	1.4	0%	7.14%	64.29%	0%
			1.8	0%	10.00%	20.00%	0%
		0.5	1.4	0%	7.14%	50.00%	0%
			1.8	0%	9.09%	9.09%	0%
		0.75	1.4	0%	0.00%	26.67%	0%
			1.8	0%	8.33%	8.33%	0%
		0.9	1.4	0%	6.25%	12.50%	0%
			1.8	0%	7.69%	0.00%	0%
	50%	0.1	1.4	0%	6.67%	93.33%	0%
			1.8	0%	8.33%	41.67%	0%
		0.25	1.4	0%	6.25%	62.50%	0%
			1.8	0%	7.69%	23.08%	0%
		0.5	1.4	0%	6.25%	50.00%	0%
			1.8	0%	15.38%	15.38%	0%
		0.75	1.4	0%	11.76%	23.53%	0%
			1.8	0%	14.29%	7.14%	0%
		0.9	1.4	0%	16.67%	11.11%	0%
			1.8	0%	28.57%	0.00%	0%
	75%	0.1	1.4	0%	11.11%	94.44%	0%
			1.8	0%	14.29%	42.86%	0%
		0.25	1.4	0%	16.67%	66.67%	0%
			1.8	0%	21.43%	21.43%	0%
		0.5	1.4	0%	15.79%	47.37%	0%
			1.8	0%	20.00%	13.33%	0%
		0.75	1.4	0%	15.00%	25.00%	0%
			1.8	0%	26.67%	6.67%	0%
		0.9	1.4	0%	19.05%	9.52%	0%
			1.8	0%	31.25%	0.00%	0%
60	25%	0.1	1.4	0%	0.00%	93.33%	0%
			1.8	0%	0.00%	50.00%	0%
		0.25	1.4	0%	0.00%	62.50%	0%
			1.8	0%	0.00%	23.08%	0%
		0.5	1.4	0%	0.00%	50.00%	0%
			1.8	0%	7.69%	15.38%	0%
		0.75	1.4	0%	5.88%	23.53%	0%
			1.8	0%	7.14%	7.14%	0%
		0.9	1.4	0%	11.11%	11.11%	0%
			1.8	0%	14.29%	7.14%	0%
	50%	0.1	1.4	0%	0.00%	94.44%	0%
			1.8	0%	0.00%	50.00%	0%
		0.25	1.4	0%	0.00%	63.16%	0%
			1.8	0%	6.67%	26.67%	0%
		0.5	1.4	0%	10.00%	50.00%	0%
			1.8	0%	12.50%	12.50%	0%
		0.75	1.4	0%	9.52%	28.57%	0%
			1.8	0%	11.76%	11.76%	0%
		0.9	1.4	0%	13.64%	9.09%	0%
			1.8	0%	23.53%	5.88%	0%
	75%	0.1	1.4	0%	9.52%	90.48%	0%
			1.8	0%	11.76%	47.06%	0%
		0.25	1.4	0%	13.64%	63.64%	0%
			1.8	0%	16.67%	22.22%	0%
		0.5	1.4	0%	17.39%	47.83%	0%
			1.8	0%	26.32%	10.53%	0%
		0.75	1.4	0%	20.83%	25.00%	0%
			1.8	0%	30.00%	10.00%	0%
		0.9	1.4	0%	24.00%	12.00%	0%
			1.8	0%	33.33%	4.76%	0%

Table B. 17: Comparison of Solution Times for Type-II

Solution Time (sec.)							
Set of Tasks (N)	**Multi-manned**	**Order Strength**	**Cycle Time**	**AD-GMM-II**	**RM-II**	**RCPSP-II**	**AD-CP-II**
30	25%	0.1	1.4	Timeout (3 hr)	478	Timeout (3 hr)	410
			1.8	Timeout (3 hr)	199	Timeout (3 hr)	168
		0.25	1.4	Timeout (3 hr)	383	Timeout (3 hr)	341
			1.8	Timeout (3 hr)	137	Timeout (3 hr)	118
		0.5	1.4	Timeout (3 hr)	352	Timeout (3 hr)	307
			1.8	Timeout (3 hr)	94	Timeout (3 hr)	79
		0.75	1.4	Timeout (3 hr)	175	Timeout (3 hr)	155
			1.8	Timeout (3 hr)	74	Timeout (3 hr)	66
		0.9	1.4	Timeout (3 hr)	86	Timeout (3 hr)	73
			1.8	Timeout (3 hr)	18	Timeout (3 hr)	17
	50%	0.1	1.4	Timeout (3 hr)	714	Timeout (3 hr)	651
			1.8	Timeout (3 hr)	278	Timeout (3 hr)	248
		0.25	1.4	Timeout (3 hr)	349	Timeout (3 hr)	389
			1.8	Timeout (3 hr)	148	Timeout (3 hr)	132
		0.5	1.4	Timeout (3 hr)	244	Timeout (3 hr)	325
			1.8	Timeout (3 hr)	96	Timeout (3 hr)	88
		0.75	1.4	Timeout (3 hr)	215	Timeout (3 hr)	199
			1.8	Timeout (3 hr)	85	Timeout (3 hr)	75
		0.9	1.4	Timeout (3 hr)	65	Timeout (3 hr)	95
			1.8	Timeout (3 hr)	31	Timeout (3 hr)	28
	75%	0.1	1.4	Timeout (3 hr)	1512	Timeout (3 hr)	889
			1.8	Timeout (3 hr)	1093	Timeout (3 hr)	419
		0.25	1.4	Timeout (3 hr)	1099	Timeout (3 hr)	492
			1.8	Timeout (3 hr)	962	Timeout (3 hr)	381
		0.5	1.4	Timeout (3 hr)	1064	Timeout (3 hr)	404
			1.8	Timeout (3 hr)	893	Timeout (3 hr)	269
		0.75	1.4	Timeout (3 hr)	959	Timeout (3 hr)	308
			1.8	Timeout (3 hr)	857	Timeout (3 hr)	235
		0.9	1.4	Timeout (3 hr)	895	Timeout (3 hr)	273
			1.8	Timeout (3 hr)	794	Timeout (3 hr)	186
40	25%	0.1	1.4	Timeout (3 hr)	2567	Timeout (3 hr)	2298
			1.8	Timeout (3 hr)	1905	Timeout (3 hr)	1805
		0.25	1.4	Timeout (3 hr)	2279	Timeout (3 hr)	1973
			1.8	Timeout (3 hr)	1893	Timeout (3 hr)	1725
		0.5	1.4	Timeout (3 hr)	2138	Timeout (3 hr)	1867
			1.8	Timeout (3 hr)	1814	Timeout (3 hr)	1630
		0.75	1.4	Timeout (3 hr)	1924	Timeout (3 hr)	1777
			1.8	Timeout (3 hr)	1691	Timeout (3 hr)	1437
		0.9	1.4	Timeout (3 hr)	1702	Timeout (3 hr)	1426
			1.8	Timeout (3 hr)	3221	Timeout (3 hr)	1322
	50%	0.1	1.4	Timeout (3 hr)	3071	Timeout (3 hr)	2618
			1.8	Timeout (3 hr)	2659	Timeout (3 hr)	2322
		0.25	1.4	Timeout (3 hr)	2962	Timeout (3 hr)	2560
			1.8	Timeout (3 hr)	2713	Timeout (3 hr)	2268
		0.5	1.4	Timeout (3 hr)	3011	Timeout (3 hr)	2470
			1.8	Timeout (3 hr)	2452	Timeout (3 hr)	2215
		0.75	1.4	Timeout (3 hr)	2641	Timeout (3 hr)	2316
			1.8	Timeout (3 hr)	2536	Timeout (3 hr)	2157
		0.9	1.4	Timeout (3 hr)	2536	Timeout (3 hr)	2223
			1.8	Timeout (3 hr)	2303	Timeout (3 hr)	2030
	75%	0.1	1.4	Timeout (3 hr)	3727	Timeout (3 hr)	3330
			1.8	Timeout (3 hr)	3244	Timeout (3 hr)	2860
		0.25	1.4	Timeout (3 hr)	3657	Timeout (3 hr)	3174
			1.8	Timeout (3 hr)	3151	Timeout (3 hr)	2742
		0.5	1.4	Timeout (3 hr)	3653	Timeout (3 hr)	3220
			1.8	Timeout (3 hr)	3059	Timeout (3 hr)	2699
		0.75	1.4	Timeout (3 hr)	3178	Timeout (3 hr)	2810
			1.8	Timeout (3 hr)	2997	Timeout (3 hr)	2591
		0.9	1.4	Timeout (3 hr)	2978	Timeout (3 hr)	2668
			1.8	Timeout (3 hr)	2769	Timeout (3 hr)	2376

(Table B.17 Continued)

Solution Time (sec.)							
Set of Tasks (N)	Multi-manned	Order Strength	Cycle Time	AD-GMM-II	RM-II	RCPSP-II	AD-CP-II
50	25%	0.1	1.4	Timeout (3 hr)	10414	Timeout (3 hr)	9453
			1.8	Timeout (3 hr)	9020	Timeout (3 hr)	8227
		0.25	1.4	Timeout (3 hr)	9932	Timeout (3 hr)	9064
			1.8	Timeout (3 hr)	9211	Timeout (3 hr)	8359
		0.5	1.4	Timeout (3 hr)	9596	Timeout (3 hr)	8761
			1.8	Timeout (3 hr)	8618	Timeout (3 hr)	7879
		0.75	1.4	Timeout (3 hr)	8683	Timeout (3 hr)	7931
			1.8	Timeout (3 hr)	8109	Timeout (3 hr)	7381
		0.9	1.4	Timeout (3 hr)	8139	Timeout (3 hr)	7909
			1.8	Timeout (3 hr)	7618	Timeout (3 hr)	6676
	50%	0.1	1.4	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	10628
			1.8	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	8967
		0.25	1.4	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	10157
			1.8	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	8703
		0.5	1.4	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	9948
			1.8	Timeout (3 hr)	10311	Timeout (3 hr)	8591
		0.75	1.4	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	9761
			1.8	Timeout (3 hr)	10718	Timeout (3 hr)	8406
		0.9	1.4	Timeout (3 hr)	10295	Timeout (3 hr)	9638
			1.8	Timeout (3 hr)	10649	Timeout (3 hr)	8224
	75%	0.1	1.4	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)
			1.8	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	9237
		0.25	1.4	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)
			1.8	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	9086
		0.5	1.4	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	10533
			1.8	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	8831
		0.75	1.4	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	10038
			1.8	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	8549
		0.9	1.4	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	9891
			1.8	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	8343
60	25%	0.1	1.4	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)
			1.8	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)
		0.25	1.4	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)
			1.8	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)
		0.5	1.4	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)
			1.8	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)
		0.75	1.4	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)
			1.8	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	9961
		0.9	1.4	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	10549
			1.8	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	9228
	50%	0.1	1.4	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)
			1.8	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)
		0.25	1.4	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)
			1.8	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)
		0.5	1.4	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)
			1.8	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)
		0.75	1.4	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)
			1.8	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	10157
		0.9	1.4	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)
			1.8	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	9524
	75%	0.1	1.4	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)
			1.8	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)
		0.25	1.4	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)
			1.8	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)
		0.5	1.4	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)
			1.8	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)
		0.75	1.4	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)
			1.8	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	10624
		0.9	1.4	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)
			1.8	Timeout (3 hr)	Timeout (3 hr)	Timeout (3 hr)	9867

Table B. 18: Comparison of Gap Percentages for Type-II

Gap Percentage							
Set of Tasks (N)	**Multi-manned**	**Order Strength**	**Cycle Time**	**AD-GMM-II**	**RM-II**	**RCPSP-II**	**AD-CP-II**
30	25%	0.1	1.4	0.00%	22.39%	19.40%	0.00%
			1.8	0.00%	10.98%	17.92%	0.00%
		0.25	1.4	0.00%	28.36%	19.40%	0.00%
			1.8	0.00%	16.76%	17.92%	0.00%
		0.5	1.4	0.00%	2.86%	15.71%	0.00%
			1.8	0.00%	8.89%	18.33%	0.00%
		0.75	1.4	0.00%	24.36%	16.03%	0.00%
			1.8	0.00%	27.41%	11.17%	0.00%
		0.9	1.4	0.00%	39.74%	14.10%	0.00%
			1.8	0.00%	20.30%	5.58%	0.00%
	50%	0.1	1.4	6.47%	13.67%	36.69%	0.00%
			1.8	0.00%	7.87%	24.16%	0.00%
		0.25	1.4	4.83%	13.79%	31.03%	0.00%
			1.8	0.00%	17.20%	16.13%	0.00%
		0.5	1.4	0.00%	18.62%	31.03%	0.00%
			1.8	0.00%	17.20%	16.13%	0.00%
		0.75	1.4	0.00%	20.25%	18.99%	0.00%
			1.8	0.00%	20.69%	11.33%	0.00%
		0.9	1.4	0.00%	85.44%	18.99%	0.00%
			1.8	0.00%	39.41%	1.97%	0.00%
	75%	0.1	1.4	14.71%	11.76%	44.12%	0.00%
			1.8	2.22%	21.11%	30.00%	0.00%
		0.25	1.4	6.62%	19.85%	34.56%	0.00%
			1.8	1.57%	19.37%	19.90%	0.00%
		0.5	1.4	0.00%	34.25%	26.03%	0.00%
			1.8	0.00%	33.51%	13.61%	0.00%
		0.75	1.4	0.00%	29.30%	20.38%	0.00%
			1.8	0.00%	13.40%	13.88%	0.00%
		0.9	1.4	0.00%	68.15%	18.47%	0.00%
			1.8	0.00%	45.45%	1.44%	0.00%
40	25%	0.1	1.4	17.99%	25.90%	43.88%	0.00%
			1.8	4.65%	36.05%	31.40%	0.00%
		0.25	1.4	11.49%	28.38%	39.19%	0.00%
			1.8	2.26%	29.38%	37.85%	0.00%
		0.5	1.4	8.33%	50.00%	30.13%	0.00%
			1.8	1.11%	56.67%	35.00%	0.00%
		0.75	1.4	5.13%	58.97%	30.13%	0.00%
			1.8	0.00%	36.27%	22.80%	0.00%
		0.9	1.4	3.51%	77.78%	25.15%	0.00%
			1.8	0.00%	74.61%	16.06%	0.00%
	50%	0.1	1.4	23.19%	27.54%	39.13%	0.00%
			1.8	8.43%	30.72%	33.13%	0.00%
		0.25	1.4	20.83%	37.50%	38.19%	0.00%
			1.8	4.05%	23.70%	39.31%	0.00%
		0.5	1.4	18.00%	38.67%	39.33%	0.00%
			1.8	2.79%	31.84%	27.93%	0.00%
		0.75	1.4	16.13%	60.00%	30.32%	0.00%
			1.8	2.56%	61.03%	30.26%	0.00%
		0.9	1.4	13.21%	81.13%	27.67%	0.00%
			1.8	1.47%	63.73%	11.76%	0.00%
	75%	0.1	1.4	28.57%	24.29%	46.43%	0.00%
			1.8	11.45%	21.69%	39.76%	0.00%
		0.25	1.4	25.87%	34.27%	46.15%	0.00%
			1.8	5.52%	45.86%	31.49%	0.00%
		0.5	1.4	20.13%	36.24%	41.61%	0.00%
			1.8	4.30%	37.63%	35.48%	0.00%
		0.75	1.4	18.47%	73.89%	35.03%	0.00%
			1.8	2.06%	59.28%	35.05%	0.00%
		0.9	1.4	10.65%	82.25%	33.14%	0.00%
			1.8	1.44%	64.59%	15.79%	0.00%

(Table B.18 Continued)

Gap Percentage							
Set of Tasks (N)	Multi-manned	Order Strength	Cycle Time	AD-GMM-II	RM-II	RCPSP-II	AD-CP-II
50	25%	0.1	1.4	38.41%	10.14%	60.87%	0.00%
			1.8	14.29%	13.14%	56.57%	0.00%
		0.25	1.4	35.46%	22.70%	63.12%	0.00%
			1.8	9.44%	15.56%	58.33%	0.00%
		0.5	1.4	30.61%	34.01%	55.10%	0.00%
			1.8	7.73%	14.43%	50.52%	0.00%
		0.75	1.4	27.56%	32.69%	43.59%	0.00%
			1.8	5.16%	20.19%	39.91%	0.00%
		0.9	1.4	25.00%	59.52%	39.29%	0.00%
			1.8	4.91%	37.50%	31.25%	0.00%
	50%	0.1	1.4	45.32%	13.67%	74.10%	0.00%
			1.8	19.32%	15.91%	60.80%	0.00%
		0.25	1.4	41.96%	17.48%	62.94%	0.00%
			1.8	13.19%	28.02%	65.38%	0.00%
		0.5	1.4	35.14%	29.73%	52.70%	0.00%
			1.8	10.11%	40.43%	43.09%	0.00%
		0.75	1.4	30.52%	48.05%	44.16%	0.00%
			1.8	8.08%	55.56%	47.47%	0.00%
		0.9	1.4	26.06%	53.94%	40.61%	0.00%
			1.8	7.21%	60.58%	43.27%	0.00%
	75%	0.1	1.4	50.00%	25.37%	87.31%	0.00%
			1.8	18.71%	26.90%	60.82%	0.00%
		0.25	1.4	44.29%	32.86%	64.29%	0.00%
			1.8	14.61%	31.46%	65.17%	0.00%
		0.5	1.4	36.55%	57.24%	58.62%	0.00%
			1.8	12.50%	43.48%	59.78%	0.00%
		0.75	1.4	32.03%	51.63%	48.37%	0.00%
			1.8	10.99%	59.69%	58.12%	0.00%
		0.9	1.4	27.22%	66.27%	49.70%	0.00%
			1.8	7.73%	67.15%	39.13%	0.00%
60	25%	0.1	1.4	42.75%	52.17%	61.59%	7.25%
			1.8	18.99%	48.04%	69.27%	0.00%
		0.25	1.4	39.58%	49.31%	70.14%	6.25%
			1.8	12.17%	43.92%	62.43%	0.00%
		0.5	1.4	34.23%	41.61%	55.70%	4.03%
			1.8	9.74%	35.90%	53.33%	0.00%
		0.75	1.4	29.30%	33.76%	43.95%	2.55%
			1.8	6.90%	31.03%	45.81%	0.00%
		0.9	1.4	25.73%	34.50%	56.14%	0.00%
			1.8	6.16%	30.81%	44.08%	0.00%
	50%	0.1	1.4	53.24%	58.99%	74.82%	9.35%
			1.8	23.89%	52.78%	74.44%	0.00%
		0.25	1.4	46.58%	50.68%	70.55%	8.22%
			1.8	14.81%	44.97%	58.20%	0.00%
		0.5	1.4	41.83%	45.10%	66.67%	7.19%
			1.8	11.94%	38.31%	61.19%	0.00%
		0.75	1.4	36.25%	38.75%	51.25%	5.63%
			1.8	8.61%	33.97%	55.50%	0.00%
		0.9	1.4	31.36%	36.69%	46.15%	4.14%
			1.8	7.44%	30.23%	40.47%	0.00%
	75%	0.1	1.4	58.57%	65.71%	75.71%	11.43%
			1.8	24.42%	57.56%	82.56%	3.49%
		0.25	1.4	55.17%	54.48%	78.62%	10.34%
			1.8	18.72%	44.92%	66.31%	1.60%
		0.5	1.4	47.02%	48.34%	65.56%	8.61%
			1.8	13.78%	43.88%	62.24%	0.00%
		0.75	1.4	40.51%	39.24%	58.23%	7.59%
			1.8	8.61%	35.89%	49.76%	0.00%
		0.9	1.4	37.20%	36.59%	50.00%	7.32%
			1.8	8.29%	30.88%	50.23%	0.00%