6-1-2007

Mixed-integer linear programming approach to U-line balancing with objective of achieving proportional throughput per worker in a dynamic environment

Reyhan Erin

Rochester Institutes of Technology

Mixed-Integer Linear Programming Approach to U-Line Balancing with Objective of Achieving Proportional Throughput per Worker in a Dynamic Environment

A Thesis

Submitted in partial fulfillment of the

requirement for the degree of

Master of Science in Industrial Engineering

at the

Kate Gleason College of Engineering

Industrial and Systems Engineering Department

by

Reyhan Erin

B.S. Industrial Engineering, Kocaeli University

June, 2007

DEPARTMENT OF INDUSTRIAL AND SYSTEMS ENGINEERING

KATE GLEASON COLLEGE OF ENGINEERING

ROCHESTER INSTITUTE OF TECHNOLOGY

ROCHESTER, NEW YORK

CERTIFICATE OF APPROVAL

M.S. DEGREE THESIS

The M.S. Degree Thesis of Reyhan Erin

has been examined and approved by the

thesis committee as satisfactory for the

thesis requirement for the

Master of Science degree

Approved by:

____

Dr. Moises Sudit, Primary Advisor

____

Dr. Andres Carrano, Committee Member

Acknowledgement

I would like to thank my family, especially my sister Nuray Erin, for their continuous support throughout my life. I feel very blessed to have such a great family.

This thesis would never been possible nor successful without the help of my advisors Dr. Moises Sudit and Dr. Andres Carrano and Thomas Glaser from Delphi Corporation. Their valuable input and advices contributed significantly in the success of this paper. I truly appreciate their time, patience, and support.

I also would like to thank my friends and my second family in USA Hediye Ali and late Syed Haider Ali for providing the motivation, love and support I needed. Lastly, I would like to thank Dr. Elliot Vessel who helped me to get my first coop that opened the opportunities to advance in my education and career in USA.

Abstract

One of the major challenges of manufacturing companies is to remain competitive in a very dynamic environment dictated by fluctuations in production rate and customer demand. These challenges may be attributed to frequent changes in customer expectations, unsteady economic conditions or failure to reach the projected throughput due to inefficiencies in production systems.

Survival in such a dynamic environment is contingent on implementing manufacturing systems that are able to adapt to change quickly and economically. The *U-Shaped* production cell is considered to be one of the most flexible techniques for changing the number of workers in the cell to match cell cycle time to planned cycle time. However, companies currently use a trial-and-error method to develop walk-paths. It is a very iterative and time consuming process that does not always guarantee an optimal solution.

Walk-paths need to be performed for all possible number of workers. Fluctuations are adapted to by altering only the number of workers and the worker's walk-path without changing the number of stations and task allocations. Selecting the best configuration (i.e. optimal number of stations and task allocation) is dependant upon the *linearity metric* i.e. the measurement of the proportional throughput per worker. Designing the production cell by considering the linearity helps to keep direct labor costs per unit at a minimum for any number of workers employed.

This thesis proposes a *mixed integer linear model for U-shaped lines* that determines the best cell configuration for various number of workers with the objective function of achieving proportional throughput per worker and decreasing the iteration time. The problem originated at *Delphi Corporation* but has been generalized to be applicable to other Lean systems. The model has been constructed using *OPL Studio 3.7.*

Key Words: *U-Line, Line Balancing, Linearity, Proportional, Mixed-Integer Linear Programming, Mathematical Model, Walk-path*

Table of Contents

1 Introduction

The U-shaped production line is a special type of cellular manufacturing system used in just-in-time (JIT) production systems. Companies competing with cost, quality, and response time prefer JIT production systems. U-shaped cells are more flexible to changes in demand and production in comparison to traditional assembly lines. Moreover, the number of workers is always lower or the same in JIT since multifunctional operators can be assigned to multiple stations.

The production cell needs to be rebalanced whenever demand changes. The least costly and most practical rebalancing technique is to revise walk-paths for the required number of workers without modifying the current stations settings (i.e. number of stations, location of stations, and task assignments). A walk-path is the group of stations a worker is assigned to. The number of stations in a walk-path can range from 1 to M (i.e. total number of stations in the cell).

Industrial engineers use a heuristic trial and error method to design the walk-paths for every worker at each cell configuration. However, this method is iterative, tedious, and time consuming. Moreover, previous line balancing algorithm is developed for stable environments where the number of workers is fixed. Those approaches are not applicable if it is a dynamic environment in which cycle time changes frequently due to fluctuations in demand and production. Thus, the initial cell design (i.e. optimal number of stations and work load assignments at each station) should be created to provide the ability of responding to cycle time changes by only modifying the walk-paths.

The thesis problem originated at Delphi Corporation, the manufacturer of automotive parts. Delphi has challenges with high labor cost and variation in production and demand. In order to keep labor costs at a minimum in a dynamic environment, walk-path assignments are created to maintain a direct labor cost constant per unit for varying number of workers.

Hence, this research proposes a mixed-integer linear model for U-lines to decrease iteration time in line balancing, and minimize labor costs per unit by developing flexible walk-path assignments for dynamic environments. As it is a NP-Hard problem, the model is divided into two stages to overcome the solving time issues. The first stage determines the optimal task allocations to a given number of stations with the objective function of balancing the work load at each station. Then, workers are assigned to stations determined at the first stage in the second phase of the model with the aim of equalizing operator times. Since it is developed for dynamic environments, the model is repeated for all N, N= {2, 3,…, M-1}. N is the number of workers employed in a cell with M numbers of stations. In order to evaluate the results, a scatter plot is charted between workers and throughput. A linear line that will pass from the origin will be fit onto the points to measure the linearity. R^2 is used to measure the linearity. If the linearity is perfect (i.e. $R^2 = 1$), then the direct labor cost per unit is the same for any number of workers in the cell.

1.1 JIT Production System

Miltenburg (2001) describes JIT as an umbrella term for a number of techniques whose purpose is to improve product quality and reduce cost by eliminating all waste in the production system. Due to its wide range of advantages, JIT has gained increasing

popularity at many manufacturing companies. Tables 1 and 2 describe the techniques associated with JIT along with its advantages.

Table 1. Benefits of JIT Production Systems

1. WIP reduction	5. Increased Flexibility
2. Increased Quality	6. Raw materials/parts reduction
3. Increased productivity	7. Lower overheads
4. Reduced Spaced Requirements	8. Increased Employee motivation

Table 2. JIT Techniques and Approaches

Core Techniques	Supportive Techniques	
1. Cellular Manufacturing	1. Group Technology	4. Line Stopping
2. Set-Up Time Reduction	2. Smallest Machine	5. U-Lines and Flexible Labor
3. Pull Scheduling (Kanban)	3.Fool-Proofing (PokaYoke)	6. House Keeping Methods

Source: Voss, C.A. (1987). *Just In Time Manufacture*, IFS (Publications) Ltd., UK, Springer-Verlang.

The main focus area of this research is u-shaped cells with flexible worker assignments. Hence, the following sections describe the characteristics of u-shaped cells and the importance of flexible production systems.

1.2 U-Shaped Cells

U-shaped cells are the preferred layout for a one-piece flow production system in the JIT environment. The essence of the U-turn format is that the entrance and exit of a line are at the same position (Monden, 1993). Therefore, one operator can take care of the first and the last station. There are many different types of u-shaped layouts depending on the number of products produced, available manpower source, and layout constraints. According to Miltenberg (2001), some u-shaped lines are multi-lines in a single U, double-dependent U-lines, embedded U-lines, figure eight pattern U-lines, and multi U-line facilities. Those are called complex U-lines. However, the simple u-shaped line demonstrated in Figure 1 is the focus of this research.

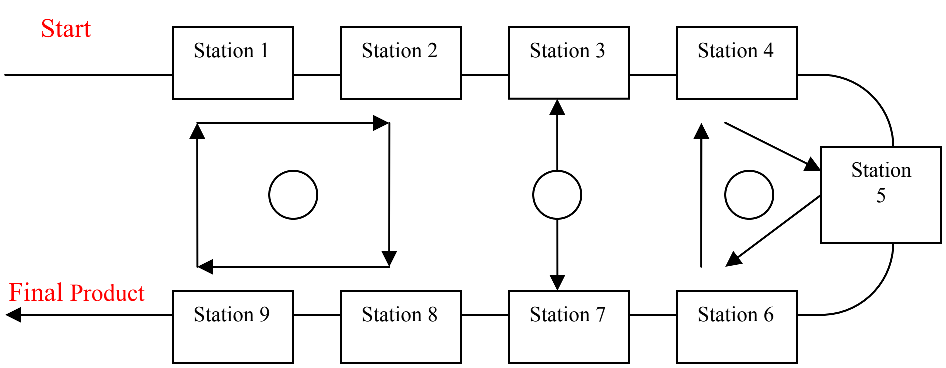


Figure 1. Simple U-Shaped line

The u-shaped line has many benefits over the traditional line. The traditional assembly line with a conveyor system was developed by Henry Ford and used in different

industries for years until the development of JIT and cellular manufacturing systems. Still, some industries continue to use traditional lines. The main benefit of the u-shaped line is the flexibility to increase and decrease the necessary number of workers when adapting to the changes in production quantities and changes in demand (Voss, 1987). As Figure 1 depicts, one operator can be responsible for more than one station. This requires multifunctional workers who are capable of undertaking different processes. Workers need to rotate in different jobs to learn each task. This research assumes that all workers are multifunctional and cross-trained. Other additional benefits are increase in quality, better communication between workers, less WIP and shorter lead times.

1.3 Flexibility in Production Cells

Flexibility of a production line gains importance if rebalancing is required often. Rebalancing once a month is common (Miltenburg, 2001). Three types of rebalancing techniques are proposed by Miltenburg.

1. Adding operators to increase the output or removing operators to decrease output
2. Moving machines
3. Changing the standard operation chart

The first technique is more common for manned cells. The second technique is more appropriate for automated production systems.

The mathematical model is developed by considering these three techniques. The model assigns the location of machines and work elements in a way so that rebalancing can be performed by only changing the number of workers inside the cell. The following is a list of the possible causes of variations:

1. **Fluctuations in Demand:** Demand rate is not always constant. Possible causes may be the competitive market, recession, or high customer expectations that require frequent model changes. This may result in greater demand, lower demand or alteration of the model.

2. **Fluctuations in Production:** Although demand may remain stable, companies cannot always meet production demands. There are numerous contributors to lower production such as:

   - *Absenteeism:* It is particularly common in the auto-industry due to strong union environment.
   - *Quality problems:* Frequent quality problems cause increase in scrap and decrease in the throughput.
   - *Equipment breakdown:* Downtime in production increases with an increase in breakdowns, and prevents reaching the projected throughput.
   - *Employee motivation:* This is essential for manned cells where the throughput rate is determined by performance of the workers.

The Cell design should be flexible to overcome these problems. In a manned cell, flexibility is accomplished by increasing or decreasing the number of employees. Production cells are a better choice for flexible environments. Stockton, Ardon-Finch, and Khalil (2005) studied walk cycle design for flexible manpower lines. They state "Change in the number of operators requires the redesign of the individual walk cycles …Walk cycle is a repetitive sequence in which to load and unload machine tools… Current methods of

designing flexible manpower lines are essentially manual and require both past experience and many design iterations before acceptable line designs can be identified".

1.4 Dynamic Environments vs. Stable Environments

Stable environment does not require frequent rebalancing due to less variation in demand and production. In that case, the line is balanced for an optimal number of workers for a given takt time. If rebalancing is required, the production line is redesigned with a relocation of stations and tasks as well as reallocation of workers. This type of rebalancing is costly and time consuming.

A dynamic environment, on the other hand, requires frequent rebalancing due to greater changes in demand or production rate. Although u-shaped cells are great for dynamic environments, the initial line balancing determines the flexibility of the line. If line balancing is performed by only considering the optimal number of workers, having a U-shaped cell will still not be sufficient enough to respond to the changes quickly and economically. Line balancing should be performed by considering the different number of workers. This thesis is designed for dynamic environments. Hence, the task allocations to stations will be performed in a way such that by changing only the worker allocations to stations, a proportional throughput, approximating the throughput of the optimal number of workers, will be achieved.

2 Problem Statement

The idea of the thesis problem was provided by Delphi Corporation. Delphi is a manufacturer of automotive parts. Lean manufacturing and JIT principles were implemented by the company to deliver high quality products faster and economically. However, Delphi has been facing two major problems lately:

(i) Greater competition in the auto industry that causes fluctuation in demand; and

(ii) A high direct labor cost.

These obstacles generate variations in throughput which require redesigning or rebalancing of the production processes. Rebalancing is used to adjust the production volume to match planned cycle time.

Delphi has many labor intense production systems. Hence, the most practical approach for rebalancing is to modify walk-paths to meet the desired cycle time. The new cycle time is calculated using required demand and available production hours. Once planned cycle time is determined, industrial engineers use manual based, heuristic trial and error methods to balance the cell as close to the required cycle time. This is an iterative and time consuming process. Moreover, the manual method cannot always ensure the optimal solution due to human errors and time limitations.

Another issue that companies encounter is variation in throughput per person. Most companies create a walk-path design for an optimal number of workers determined by planned cycle time. However, whenever variations occur in throughput, the cell needs to be rebalanced for a different number of workers. While rebalancing the cell for different

number of workers, engineers attempt to keep the labor cost per unit constant by keeping the throughput per worker constant. In reality, achieving lower unit cost than optimal is desired; however, the unit cost is usually higher for any other configurations.

In conclusion, the thesis problem is to develop a u-shaped line balancing algorithm with flexible walk-paths that will determine the best configuration in terms of optimal number of stations and workers required, in order to achieve maximum linear output across workers. Consequently, it will help to shorten cell design time and provide flexibility to adapt the changes faster and more easily without changing the direct labor cost per unit. Although the idea of the problem is originated from Delphi, the model is generalized in order to be applicable to other companies and industries.

3 Literature Review

The literature review is completed for both U-Line Balancing and Flexible Work Force assignments since the thesis problem considers these two areas at the same time:

1. Optimal task assignments to given number of stations (Research area: Line Balancing)
2. Flexible work-force assignment to given number of workstations (Research area: Worker Allocation to U-Lines)

3.1 Line Balancing

Martinez and Duff (2004) define line balancing as the assignment of approximately the same amount of workload to each workstation (worker) in an assembly line. Previous research mainly considers two types of line balancing problems. The Type I problem is

intended to minimize the required number of work-stations for a fixed cycle time. The Type II problem, on the other hand, aims to minimize the maximum cycle time for the fixed number of work-stations. Immense research has been conducted for traditional straight lines since 1955 for both these types of problems. However, U-shaped line balancing problems have been investigated since 1994 with the evolution of JIT. Previous research areas can be grouped as shown in Figure 2.

This thesis problem is deterministic and a single model line balancing for multiple workers. Thus, the literature review only focuses on the highlighted area in Figure 2.

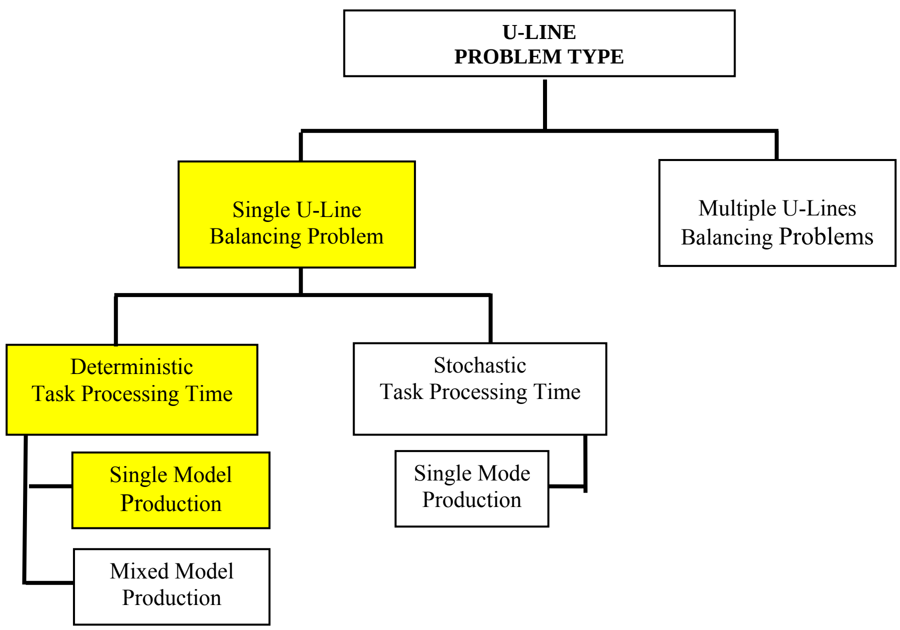


Figure 2. Classification of previous research studies

Miltenberg and Wijangard (1994) were the first to investigate line balancing problems for u-shaped lines. A dynamic programming model was developed for small size problems consisting of 11 tasks to minimize the required number of stations for a given cycle time. For medium size problems, a "maximum ranked positional weight" heuristic procedure, which is the modification of Hoffman's traditional line heuristic method, was

proposed. The purpose of this study was two fold. One is to offer a new line balancing algorithm for u-shaped lines, and the second is to prove that traditional line algorithms can be successfully adapted to new problems.

Urban (1998) extended Miltenberg and Wijangard’s study to solve larger problems by using integer programming. To overcome the solving time issue, an in depth search in branch and bound method was conducted first. He also set lower bounds on some variables to reduce the problem size. Therefore, Urban’s proposed model was able to solve problems with 21 or more tasks with better optimal solutions than the “maximum ranked positional weight” heuristic.

Wainwright and Ajenblit (1998) studied the heuristic approach to a Type I u-line balancing problem. A Genetic Algorithm (GA) approach with six different task assignment methods was applied to minimize the number of stations for fixed cycle time. They also considered total idle time and work balance between stations while distributing tasks to stations. This was the first time that GA was used to solve a U-shaped line balancing problem. The contribution of this procedure was to be able to solve larger problems like 111 tasks in less than 10 minutes processing time with similar or superior results compared to previous heuristic methods.

Scholl and Klein (1999) developed a branch and bound procedure for simple assembly line balancing (SALOME) for Type I problems in 1997. SALOME was developed to find the minimum number of stations for a given cycle time by applying local lower bound and bi-directional branching techniques. It is thought to be most effective algorithms for traditional line balancing problems. Scholl and Klein extended their previous study by modifying SALOME for u-lines. In addition to the Type I problem, they

proposed a model for Type II, which determines the optimal cycle time for a given number of stations, and another formulation for line maximization when cycle time and number of stations are variables. Their algorithms were capable of solving up to 297 tasks in a considerably short time with optimal or best solutions. However, the line maximization algorithm requires further improvement.

Sabuncuoglu and Aksu (2001) solved the ULB problem using a simulated annealing (SA) based algorithm. This algorithm is divided into two main stages. In the first stage, the solution generator module generates a new or better solution from the previous solution by relaxing the cycle time constraint. In the second stage, the SA module takes this solution and reallocates the tasks to stations with the objective of minimizing the maximum station time. The algorithm was compared to RPWT, and it showed that the SA algorithm gave the same or superior results for small and medium sizes problems. Moreover, the algorithm was tested with Scholl and Klein's (1999) data set. The majority of the results were the same and computation time requirements for both algorithms were comparable.

Martinez and Duff (2004) proposed the GA approach to improve different heuristics results obtained from previous simple assembly line balancing (SALB) research work for Type I problems. An initial solution was created with 10 different heuristics for u-lines, and then GA was applied to improve the solution. Since the study was conducted only for small sized problems, neither of the methods showed superior results. Thus, they recommended repeating this study with larger and more diverse u-line problems.

Gokcen and Agpak (2004) proposed a goal programming approach for simple u-line balancing problems. The model is based on the integer programming formulation developed by Urban for the u-line balancing problem and the goal model of Deckro and

Rangachari developed for the traditional line. Since it is a goal programming approach, the model provides "satisfactory" solutions rather than "optimal" solutions due to conflicting goals. They presented up to 30 tasks with 3 conflicting goals, which are the minimization of; (i) the number of work-stations, (ii) cycle time and (iii) the number of tasks at each station. This research is valuable if more than one conflicting goal needs to be considered in u-shaped line balance. This area of the research is new to the literature.

Most of the research proposes to minimize the number of stations (workers). In fact, they minimize the number of workers instead of physical work stations, since a worker can work at more than one station in u-line. However, the word "station" has been used in the literature for u-lines due to its inheritance from simple assembly line problems. All line balancing research has been done for stable environments. The difference between a stable and dynamic environment is the frequency of rebalancing. If rebalancing is often required due to variations in production or demand, it is considered to be a dynamic environment. Thus, balancing technique should be easy and economical. Proposed algorithms from previous research require the redesigning of work stations and task assignments at each station. Redesigning the cell from the beginning is costly and time-consuming. Moreover, previous line balancing research does not consider the flow issues such as walking time and crossover. The model for the thesis is designed for dynamic environments by including flow issues (i.e. walking time and crossover). Consequently, rebalancing will be achieved only by modifying walk-paths (worker to stations assignments) without redesigning the work stations and task assignments at each station.

3.2 Worker Allocation to U-Lines

Research conducted in worker allocation aims to prove that optimizing workforce assignment is as essential as line balancing. The majority of the research in line balancing focuses on the minimization of the number of workers (stations) without considering walking times and waiting times. However, it is not a practical approach in real life. Ohno and Nakade were the first to criticize Miltenberg and Wijingard's (1994) approach to the u-line balancing problem, which only takes into account the minimization of the number of stations and ignores the crossover issues in walk-paths. Their main research interest was optimal operator allocation in u-shaped cells and four papers have been published by them in this area.

Ohno and Nakade (1996) proposed mathematical functions to calculate the cycle time under an optimal worker allocation scenario for the established line. Firstly, an optimal worker allocation problem was examined for one and two workers in a u-shaped production line with k machines located at the same distance under deterministic process and walking times. Subsequently, a formula was derived for n workers and m machines ($n<m$) that minimized the overall cycle time. Finally, one worker in the line scenario was simulated under stochastic processing and walking time. It was concluded that cycle time calculations vary based on either the number of workers in the line or the total waiting time of the workers in one cycle.

Ohno and Nakade (1997) proposed Petri Net and GSMP theories to prove that reduction in variances of operation and walking times of workers increases the throughput. In addition, throughput is the same for the reverse system, as well.

Ohno and Nakade (1999) proposed a model for deterministic walking and process time cases. In this paper, first the minimum number of workers is determined under a given cycle time, and then an optimal worker allocation with a minimum number of operators is proposed.

Nakade and Ohno (2003) worked on separate and carousel type of worker allocation using deterministic and stochastic times. In separate allocation, every worker is responsible for specific machine groups; however, all workers are charge of all the machines in carousel type of allocation. This study demonstrated that if the workload between operators is the same in separate type allocation, the system cycle time is smaller than in carousel allocation. Conversely, the carousel allocation is preferred if it is unable to balance operator times in a separate allocation.

Stockton, Ardon-Finch, and Khalil (2005) proposed flexible walk cycle designs using genetic algorithms. This paper also supports that designing the walk cycles manually is iterative and time consuming. Thus, a GA algorithm is proposed for operator walk cycle design including the walking times. Two objectives are studied. The first objective is to minimize the number of operators and the second is to reduce the smallest operator cycle time. This paper argues that minimizing the idle time between operators is not always the best practice as previously believed. It is argued that having ample capacity can be helpful when extra capacity is needed for additional tasks.

Table 3 summarizes the previous research and shows the gaps intended to be fulfilled by this study.

Table 3. Summary of Previous Research Work

Paper	**Miltenberg and Wijangard, 1994**	**Urban, 1998**
Objective	(1) Decrease the number of workers for given cycle time	Decrease the number of workers for given cycle time
Method	(1) Dynamic Programming (2) Maximum Ranked Positional Weight Heuristic	Integer Programming
Result	(1) Achieved optimal solutions for up to 11 tasks (2) Proved that traditional line balancing algorithm can be successfully adapted to U-lines	Achieved optimal solutions for up to 21 tasks
GAP	(1) Applicable to stable environments (2) Walking time and crossover are not considered (3) Inefficient for medium and large scale problems	(1) Applicable to stable environments (2) Walking time and crossover are not considered (3) Inefficient for medium and large scale problems
Paper	**Wainwright and Ajenblit, 1998**	**Scholl and Klein, 1999**
Objective	Minimize the number of workers for given cycle time	(1)Minimize the number of workers for given cycle time (2) Minimize the cycle time for given number of workers (3) Minimize both at the same time
Method	Genetic Algorithm	Branch and Bound
Result	Capable of solving up to 111 tasks with good results	(1) Capable of solving up to 297 tasks with good results (2) Third objective requires further research
GAP	(1) Applicable to stable environments (2) Walking time and crossover are not considered	(1) Applicable to stable environments (2) Walking time and crossover are not considered
Paper	**Ohno and Nakade, 1999**	**Stockton, Ardon-Finch, and Khalil (2005)**
Objective	Minimize the cycle time under minimum number of workers including waiting and walking times.	(1) Developed walk-cycle designs for flexible manpower lines (2) Minimize the number of worker (FP) (3) Minimize the smallest operator cycle time
Method	Program written in FORTRAN 77	Genetic Algorithm
Result	Able to solve problems up to 100 work stations in seconds.	Developed flexible walk cycles considering walking time and crossovers
GAP	Applicable to stable environments	Applicable to stable environments

4 Methodology

4.1 Production System Description

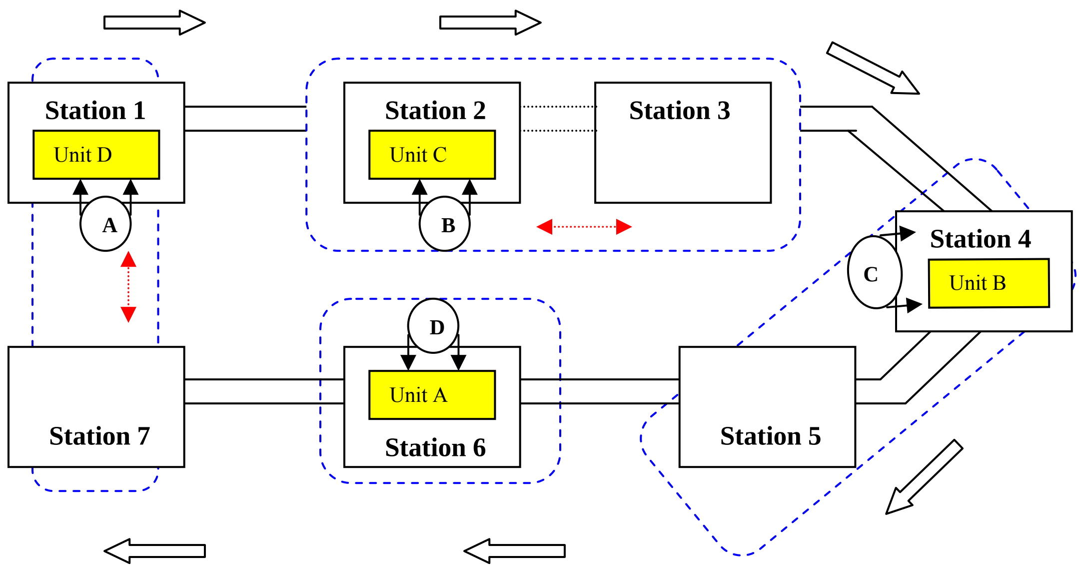


Figure 3. Function of Simple U-line

Figure 3 illustrates a simple u-shaped production cell. Multiple stations are closely located in a u-shape to form a production cell. Tasks are allocated to stations according to precedence relations.

Once tasks are allocated to stations, then assignments of workers are established. Multiple stations can be assigned to the same worker as long as crossover and cycle time constraints are satisfied. In this thesis, the term "Walk-path" is used to define each worker's area of responsibility. According to Figure 3, there are 4 walk-paths for 4

workers. Those walk-paths are “Stations 1 and 7” “Stations 2 and 3”, “ Stations 4 and 5” and “Station 6” for workers A, B,C, and D respectively.

Significant constraints to be considered during the line balancing process are (1) task precedence relations, (2) crossover, and (3) cycle time. All tasks have to be assigned to given stations by complying with precedence constraints. Crossover occurs if the worker occupies another worker’s walk-path when moving from one station to another one. For example, worker A is assigned to both stations 1 and 3 while worker B is only assigned to station 2. Worker A disrupts worker B when moving from Station 1 to Station 3. Lastly, overall processing and walking times a worker spends at all stations cannot exceed the planned cycle time.

Parts move to each station sequentially (i.e. station 1, 2,...M). In Figure 3, the part enters station 1, then moves to station 2. Once all processes are completed at station 2, the part is sent to the next station. Since worker B is responsible for both stations 2 and 3, he takes over the part from worker A and starts processing at station 2. When the part is completed, worker B moves to station 3 with the part to start processing at station 3. When he finishes processing at station 3, he sends the part to station 4 and returns to his first station (i.e. Station 2).

Some companies allow multiple workers per walk-path with a worker chasing mode. Because it makes the model more complicated, only one worker is allowed per walk-path in this research.

4.2 Trial and Error Method in Line Balancing

When a new product requirement is received from marketing, the engineer defines (1) work elements and the time to complete each element (2) precedence relationship between tasks, (3) demand, and (4) available working hours. Based on these inputs,

Step 1: Calculate the customer takt time. Takt time is the pace of production to meet customer demand. Then, compute planned cycle time.

$$Takt\ Time = \frac{Available\ \ Time}{Demand}$$

$$PCT = Takt\ Time\ *\ \%\ Quality$$

Step 2: Determine the minimum number of workers needed.

$$Min.\ Number\ of\ Workers = \frac{Total\ \ Processing\ Time}{PCT}$$

Step 3: Decide the number of stations. The number of stations is a critical factor in line balancing. Delphi uses factor method in the calculation. In factor method, a number is selected between 1 and 2 based on experience. Minimum number of workers is multiplied with a factor to calculate the number of stations.

$$Number\ of\ \ Stations = \text{Minimum Number of Workers * Factor}$$

Step 4: Allocate the tasks to stations complying with precedence requirements. Initial assignments are performed randomly. Then, workers are assigned to stations. Workers are loaded up to PCT level. Walk-path time includes tasks processing time at each station and walking time.

$$\textit{Walk-path time}_{(worker)} = \textit{Processing time at each station} + \textit{Walking time}$$

$$\textit{Walk-path time} \leq PCT$$

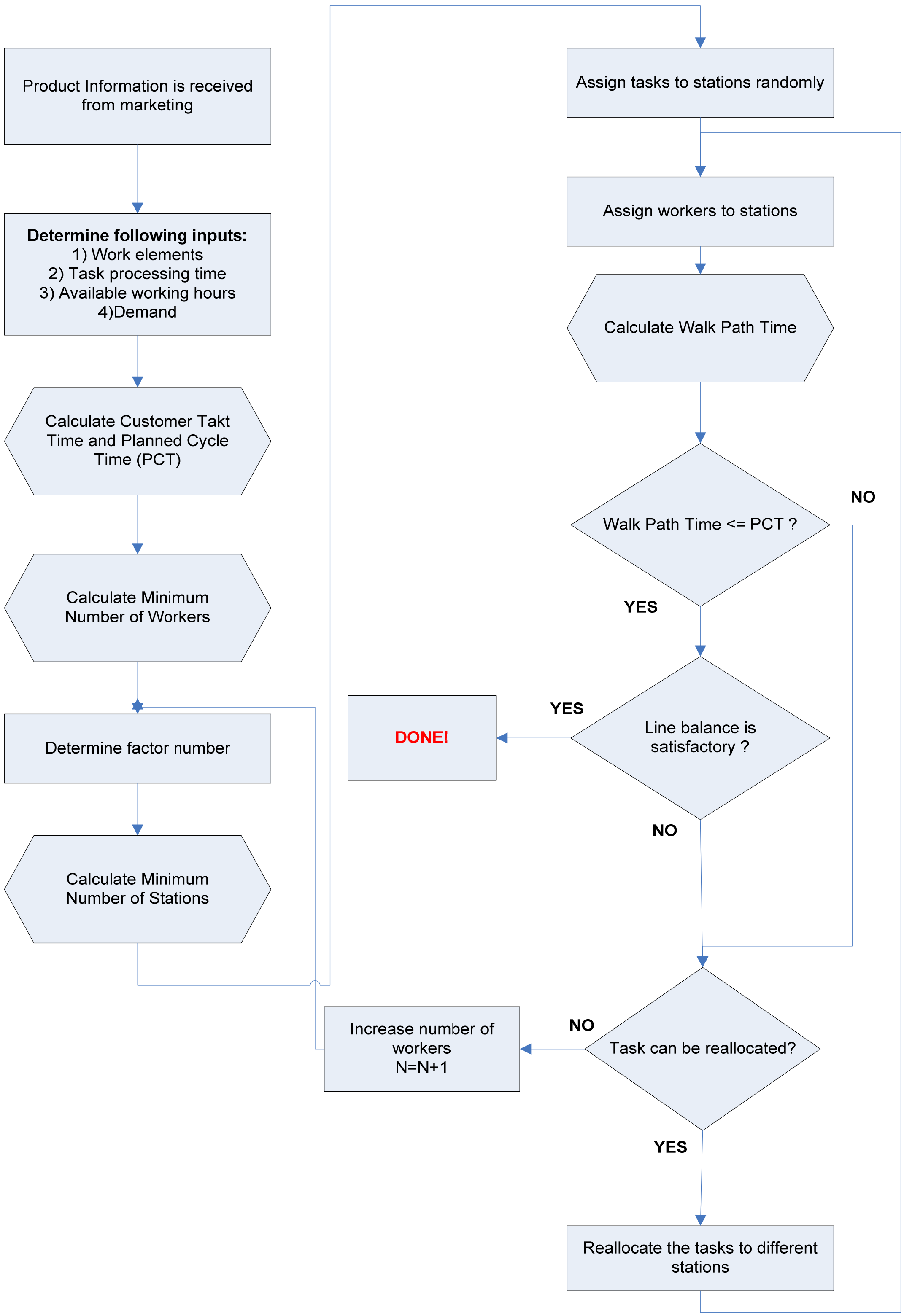
Step 5: Verify if line balance is satisfactory. If not, reallocate the tasks to stations until the desired balanced level is reached. It may require additional workers if tasks cannot be moved further and the line is not balanced.

Step 6: Once the cell is balanced for target volume, the engineer keep on searching the best assignments for a different number of workers. PCT is determined for each configuration. The maximum of the walk-path time determines the pace of production (i.e. cell cycle time).

$$\textit{Cell Cycle Time} = \textit{Maximum}\ \{\textit{Walk-path1}, \textit{Walk-path2}, \ldots, \textit{Walk-pathN}\}$$

$$\textit{Cell Cycle Time} \leq PCT$$

Diagram 1. The Heuristic Trial and Error Method used in Delphi Corporation



4.3 Mathematical Model Description

The model is constructed using *"Mixed Integer-Linear Programming"*. The problem detailed in this research is NP-Hard, thus some assumptions are made to ease the model.

1. Task processing time and walking time between stations are deterministic.
2. Operators are able to multi-task and are cross-trained.
3. Production is one-piece flow.
4. Production is designed for a single model.
5. Operators do not chase each other, and do not share the stations.
6. The breakdown in any machine is not considered in this research.
7. No waiting time for machines is considered.
8. First station has infinite raw materials in front of the station.
9. Space constraint for cell layout is not considered.

Mixed integer linear programming is used to develop the model using ILOG OPL Studio 3.7 optimization software. Experiments are conducted on Intel Pentium 4CPU, 3.20GHz, and 1 GB of RAM computers.

The mathematical model is divided into two stages due to the complex characteristics of the problem. Since it is a NP-Hard problem, the model reaches to solution in polynomial time. Thus, developing different stages helps to minimize solving time. At Stage-1, tasks are allocated to a given number of stations with the objective function of balancing the workload at stations. Then, workers are assigned to stations with the goal of

balancing the walk-path times. The second stage is run for each number of workers starting from 2 to the maximum number of workers allowed to calculate the linearity metric. If the linearity is not satisfactory, then the process must be iterated by altering Stage-1. Diagram 2 explains the logic of the model.

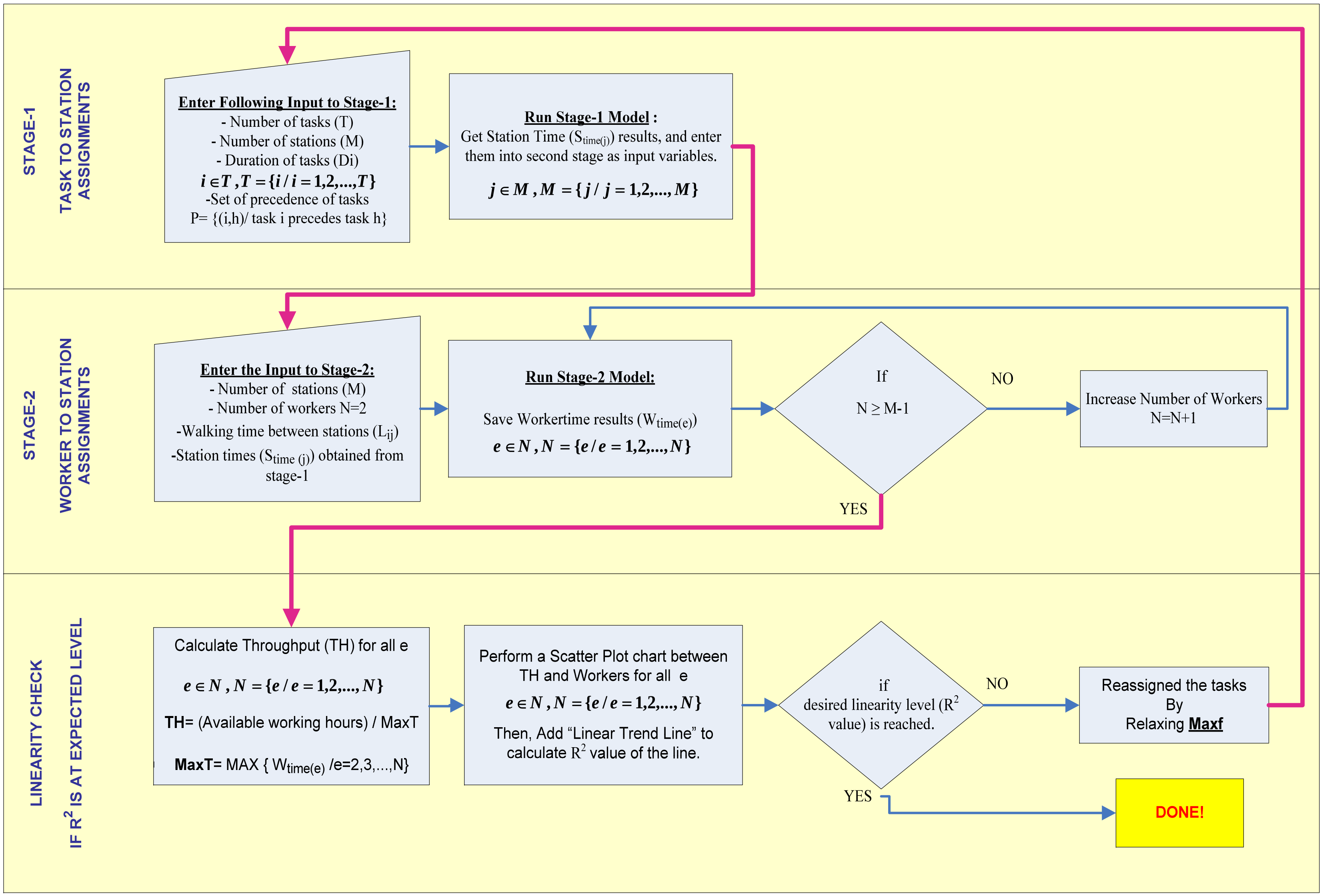


Diagram 2 Mathematical Model Process Flow

4.3.1 Stage-1 Model Description

Stage-1 is developed to distribute tasks to stations without considering (including) worker assignments. Initial objective function is to balance station times. However, designing a system with well-balanced station times may not always help to achieve proportional throughput per worker in the end. Hence, initial objective function is modified at next iterations by relaxing the balancing constraints. Relaxation occurs through reallocation of tasks to stations. Required inputs for Stage-1 are (1) number of stations, (2) number of tasks, (3) precedence relations, and (4) deterministic task processing times.

4.3.1.1 Stage-1 Notation

$i \equiv$ *indices for tasks* ($i = 1, 2, \ldots, n \ldots, T$)

$j \equiv$ *indices for workstations* ($j = 1, 2, 3 \ldots, n \ldots, M$)

Task is the work element needed to complete the final product.

Workstation is the physical working area on which tasks are assigned according to precedence diagram and cycle time requirements.

4.3.1.2 Stage-1 Input Parameters

$T \equiv$ Number of tasks

$M \equiv$ Number of workstations

$D_i \equiv$ Standard time to process the task i (times are assumed to be deterministic)

$P(i, h) \equiv$ A set of precedence of tasks, P={(i, h) / task i must be completed before task h}

4.3.1.3 Stage-1 Decision Variables

$$X_{ij} = \begin{cases} 1, & \text{if task } i \text{ is assigned to station } j \\ 0, & \text{otherwise} \end{cases}$$

$Maxf$ ≡ Maximum of the time difference between station times

$Stime_j$ ≡ Sum of all the tasks' processing time assigned to station j

4.3.1.4 Stage-1 Objective Function

Objective function is to minimize the maximum of the time difference between stations. Reducing the largest of the gap drives the station times closer. However, maxf is gradually increased (relaxed) in every iteration if good linearity is not accomplished in the end.

$$\text{Minimize } (Maxf)$$

4.3.1.5 Stage-1 Constraints

1. Each task can be assigned to *only one* workstation, and consequently tasks cannot be shared between stations.

$$\sum_{j=1}^{S} Xij = 1 \qquad \forall i = 1...T$$

2. All station has to be assigned. No dummy station is allowed. If the number of tasks is greater than the number of stations, more tasks can be assigned to same station.

$$\sum_{i=1}^{T} Xij \geq 1 \qquad \forall j = 1...M$$

3. Every task has to be assigned based on the precedence relations. P = {(h,i) / task h must be completed before task i} . If task h is assigned to station j, task i can *only* be assigned to station k, $k \in M \quad M = \{k / k = j, j+1, j+2, ...M\}$.

$$Xhj \le \sum_{k=j}^{M} Xik \qquad \forall (h,i) \in P$$

$$\forall j = 1...M$$

4. Following constraints are used to compare the times between every station. Variable *Maxf* takes the largest value of the gaps. Parts A and B give the total processing time at station j and g respectively. *Maxf* is aimed to minimize through the objective function.

$$\underbrace{\sum_{i=1}^{T} Di * Xij}_{Part\ A} - \underbrace{\sum_{i=1}^{T} Di * Xig}_{Part\ B} \le Maxf \qquad \forall j = 1...(M-1)$$

$$\forall g = (j+1)...M$$

$$\sum_{i=1}^{T} Di * Xig - \sum_{i=1}^{T} Di * Xij \le Maxf \qquad \forall j = 1...(M-1)$$

$$\forall g = (j+1)...M$$

Numerical Example:

Figure 4 explains the logic of *Maxf*. The number in parenthesis is the station time. Maximum station time is 9 minutes, and minimum station time is 5vminutes. Thus, the maximum of the difference between station times is 4. *Maxf* = 9-5 = 4

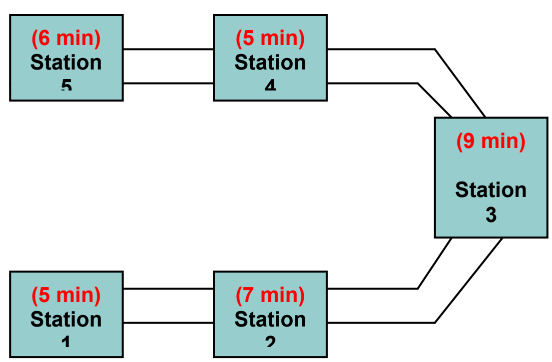


Figure 4 Illustration for Maxf

5. This is a label constraint that shows the total processing time at each station.

$$Stime_j = \sum_{i=1}^{T} Di * Xij \qquad \forall j = 1 \ldots \mathrm{M}$$

4.3.2 Stage-2 Model Description

Stage-1 is developed to establish the workload at each given station complying with precedence requirements. Stage-2, on the other hand, is used to create the walk-paths by allocating workers to these stations with the aim of balancing worker times. Worker time is sum of the task processing and walking times. This phase of model considers two major constraints when developing the walk-paths; (1) crossover and (2) balanced workload. Workers cannot be assigned to the stations that require crossing other's walk-path, because it causes disruption in the flow. Moreover, the objective of balanced worker time is tried to achieve when developing the walk-paths.

Stage-2 is repeated for all different number of workers in order to generate the linearity metric. This research reiterated Stage-2 for all e, $e \in N$ $N = \{e / e = 1,2,3,...M-1\}$. The output for M worker is same as the output from Stage-1, because the number of worker is equal to the number of stations. Consequently, each worker can only be assigned to one station.

4.3.2.1 Stage-2 Notation

e ≡ indices for workers (i = 1, 2, …, n …, N)

j ≡ indices for workstations (j = 1, 2, 3…, n …, M)

Workers perform the tasks assigned to them. All workers are cross-trained and are capable of operating each job.

4.3.2.2 Stage-2 Input Parameters

$N \equiv$ Number of workers employed.

$M \equiv$ Number of workstations

$Stime_j \equiv$ Total processing time at station j

$L_{ij} \equiv$ Walking time between station i and j. In this thesis, it is sufficient to input the distances for all i=1…M-1 and j= (i+1)…M

$LS_{\max} \equiv$ This input is used to prevent crossovers.

$$LS_{\max} = \begin{cases} M/2 \quad , & \text{if } M \text{ is even number} \\ \lfloor M/2 \rfloor, & \text{if } M \text{ is odd number} \end{cases}$$

$HS_{\max} \equiv$ This input is used to prevent crossovers.

$$HS_{\max} = \begin{cases} (M/2)+1 \ , & \text{if } M \text{ is even number} \\ \lceil M/2 \rceil + 1 \ , & \text{if } M \text{ is odd number} \end{cases}$$

4.3.2.3 Stage-2 Decision Variables

1. This variable assigns stations to workers.

$$Y_{je} = \begin{cases} 1, & \text{if station } j \text{ is assigned to worker } e \\ 0, & \text{otherwise} \end{cases}$$

2. This variable is used to determine the walk-path for each worker. If worker *e* is assigned both stations *j* and *h*, then $W_{jhe} = 1$. It means worker *e*'s walk-path includes these two stations.

$$W_{jhe} = \begin{cases} 1, & \text{if station } j \text{ and station } h \text{ is assigned to same worker } e \\ 0, & \text{otherwise} \end{cases}$$

3. Variable IM_{jhe} is referred as *Immediate station.* It is used to determine the sequence of stations that worker *e* moves in his walk-path. If $IM_{jhe} = 1$, then worker *e* moves to station *h* once he finishes processing at station *j*.

$$IM_{jhe} = \begin{cases} 1, & \text{if station } j \text{ and } h \text{ is assigned to same worker } e, \\ & \text{and if worker } e \text{ moves to station } h \text{ immediately after station } j \\ 0, & \text{otherwise} \end{cases}$$

4. Variable FS_{je} denotes *First Station* in the walk-path**.** If $FS_{je} = 1$, then *j* is the very first station in the sequence that worker *e* will begin to work.

$$FS_{je} = \begin{cases} 1, & \text{if station } j \text{ is the } \textit{first} \text{ station in the walkpath worker } e \text{ assigned} \\ 0, & \text{otherwise} \end{cases}$$

5. Variable LS_{je} implies *Last Station* in the walk-path. If $LS_{je} = 1$, then j is the last station in the sequence that worker e has to follow. Once the job is finished processing at station j, the worker goes back to the very first station to start the next cycle.

$$LS_{he} = \begin{cases} 1, & \text{if station } h \text{ is the last station in the walkpath worker } e \text{ assigned} \\ 0, & \text{otherwise} \end{cases}$$

6. Variable B_{jhe} is created to determine the end of the cycle. If $B_{jhe} = 1$, then h is the last and j is the first station in the sequence.

$$B_{jhe} = \begin{cases} 1, & \text{if station } j \text{ and } h \text{ is assigned to same worker } e, \\ & \text{and If worker } e \text{ goes back from last station } h \text{ to first station } i \\ 0, & \text{otherwise} \end{cases}$$

7. $MaxT$ ≡ Maximum of the walk-path time

$$MaxT = \max\,(Wtime_1, Wtime_2, Wtime_3, ..., Wtime_{\text{Nmax}})$$

8. $Wtime_e$ ≡ Sum of all the tasks' processing time assigned to worker *e,* also referred as walk-path time of worker e

4.3.2.4 Stage-2 Objective Function

The goal of Stage 2 is to balance worker times. $MaxT$ controls the gap between any two workers. The gap is tried to keep at a minimum by minimizing $MaxT$ in the objective function.

$$\text{Minimize } (MaxT)$$

4.3.2.5 Stage-2 Constraints

1. Every station can only be assigned to *one worker.*

$$\sum_{e=1}^{N} Y_{je} = 1 \qquad \forall j = 1\ldots M$$

2. One worker can be assigned to many stations if the number of workers is less than the number of stations. The following constraints determine the stations in every worker's walk-path. All stations that are assigned to same worker have to be allocated to that worker's walk-path.

$$1 + Wjge \geq Yge + Yje \qquad \begin{array}{l} \forall j = 1\ldots M-1 \\ \forall g = (j+1)\ldots M \\ \forall e = 1\ldots N \end{array} \qquad \ldots\ldots\ldots\ldots\ldots (1)$$

$$Wjge \leq Yge \qquad \begin{array}{l} \forall j = 1\ldots M-1 \\ \forall g = (j+1)\ldots M \\ \forall e = 1\ldots N \end{array} \qquad \ldots\ldots\ldots\ldots\ldots (2)$$

$$Wjge \leq Yje \qquad \begin{array}{l} \forall j = 1\ldots M-1 \\ \forall g = (j+1)\ldots M \\ \forall e = 1\ldots N \end{array} \qquad \ldots\ldots\ldots\ldots\ldots (3)$$

$$Wjge = 0 \qquad \begin{array}{l} \forall j = 1\ldots M \\ \forall g = 1\ldots j \\ \forall e = 1\ldots N \end{array} \qquad \ldots\ldots\ldots\ldots\ldots (4)$$

Numerical Example:

Stations 1, 2, and 7 are assigned to worker A. Thus, all of these stations are placed in worker A's walk-path as illustrated in Figure 4.

Stations 3 and 5 are assigned to worker B. Hence, both stations are located in worker B's walk-path as shown in Figure 3.

Consequently, the following variables will be assigned to the values as listed.

$W_{12A}=1$, $W_{17A}=1$, $W_{27A}=1$, $W_{35B}=1$

$W_{13A}=0$, $W_{73A}=0$, $W_{23A}=0$, $W_{15A}=0$, $W_{75A}=0$, $W_{25A}=0 W_{13B}=0$, $W_{73B}=0$, $W_{23B}=0$, $W_{15B}=0$, $W_{75B}=0$, $W_{25B}=0$

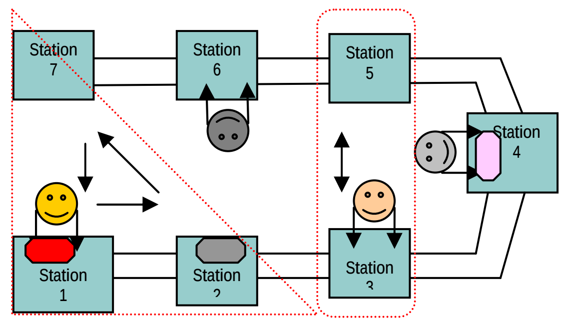


Figure 4 Walk-path assignments

3. The following constraints prevent the crossovers. Two types of crossovers are not allowed in this research; (i) diagonal and (ii) horizontal crossovers. Constraint 1 prevents crossovers diagonally; constraints 2 and 3 prevent crossovers horizontally.

$$3 - Y_{je} - Y_{ge} \geq Y_{hf} + Y_{rf} \qquad \forall j = 1 \ldots M-1 \qquad \forall g = (j+1) \ldots M$$
$$\forall h = (j+1) \ldots (g-1) \qquad \forall r = (g+1) \ldots M \qquad \ldots. (1)$$
$$\forall e = 1 \ldots N \qquad \forall f = 1 \ldots N,\ e <> f$$

$$(g-j-1)*(Yje+Yge-1) \le \sum_{k=(j+1)}^{(g-1)} Ykf \qquad \forall j = 1...(LS\max - 2)$$
$$\forall g = (j+2)...LS\max \qquad ..... (2)$$
$$\forall e = 1...N \quad \forall f = 1...N \;, \; e <> f$$

$$(g-j-1)*(Yje+Yge-1) \le \sum_{k=(j+1)}^{(g-1)} Ykf \qquad \forall j = HS\max ...(M-2)$$
$$\forall g = (j+2)...M \qquad .... (3)$$
$$\forall e = 1...N \quad \forall f = 1...N \;, \; e <> f$$

Numerical Example:

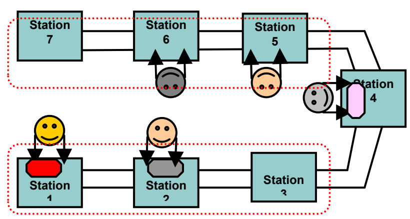


Figure 5. Horizontal Crossovers

Worker A is assigned to Stations 1 and 3. Worker B is assigned to Station 2. Worker A crossover worker B's walk-path horizontally

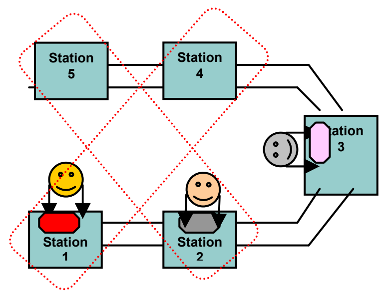


Figure 6. Diagonal Crossovers

Worker A is responsible for Stations 1 and 4. Worker B is responsible for Stations 2 and 6. Workers A and B diagonally cross each others' walk-paths.

4. The following constraints are used to determine the sequence of workstations in each walk-path. Constraints (1) determine the *next immediate* station in the sequence worker *e* has to move if multiple stations are assigned to worker *e.* Constraints (2) are used to decide the *last station* in the sequence to complete one cycle. Constraints (3) are created to determine the *first station* in the sequence. Return time from last station to first station is calculated through constraints (4).

$$IMjge \geq Wjge - \sum_{h=(j+1)}^{(g-1)} Wjhe \qquad \forall j = 1...(M-1)$$

$$\forall g = (j+1)...\mathrm{M} \qquad .... (1a)$$

$$\forall e = 1...N$$

$$Wjge \geq IMjge \qquad \forall j = 1...M-1$$

$$\forall g = (j+1)...M \qquad .... (1b)$$

$$\forall e = 1...N$$

$$\sum_{h=(j+1)}^{(g-1)} Wjhe \leq M * (1 - IMjge) \qquad \forall j = 1...(M-1)$$

.

$$\forall g = (j+1)...\mathrm{M} \qquad ...... (1c)$$

$$\forall e = 1...N$$

$$\sum_{g=1}^{M} (g * LSge) \geq (j * Yje) \qquad \forall j = 1...M$$

$$...... (2a)$$

$$\forall e = 1...N$$

$$\sum_{j=1}^{M} LSje = 1 \qquad \forall e = 1...N \qquad ...... (2b)$$

$$LSje \leq Yje \qquad \forall e = 1...N$$

$$\forall j = 1...M \qquad ...... (2c)$$

$$\sum_{g=1}^{M} (M-g) * FSge \geq (M-j) * Yje \qquad \forall j = 1...M$$

$$...... (3a)$$

$$\forall e = 1...N$$

$$\sum_{j=1}^{M} FSje = 1 \qquad \forall e = 1...N \qquad ...... (3b)$$

$$FSje \leq Yje \qquad \forall e = 1...N \quad \forall j = 1...M \qquad \text{...... (3c)}$$

$$Bjge + 1 \geq FSge + LSje \qquad \forall j = 2...M \quad \forall g = 1...(j-1) \quad \forall e = 1...N \qquad \text{...... (4a)}$$

$$Bjge \leq FSge \qquad \forall e = 1...N \quad \forall j = 2...M \quad \forall g = 1...(j-1) \qquad \text{...... (4b)}$$

$$Bjge \leq LSge \qquad \forall e = 1...N \quad \forall j = 2...M \quad \forall g = 1...(j-1) \qquad \text{....... (4c)}$$

Numerical Example:

Worker A is responsible for Stations 1, 2 and 7. The process starts at Station 1, then moves to Station 2, and finishes at Station 7. Same process flow iterates at every cycle.

First Station= Station 1 (FS_{1A}=1)

Immediate Station = Station 2 (IM_{12A} =1),

Last Station= Station 7 (LS_{7A} =1), *Back function* is B_{71A}=1.

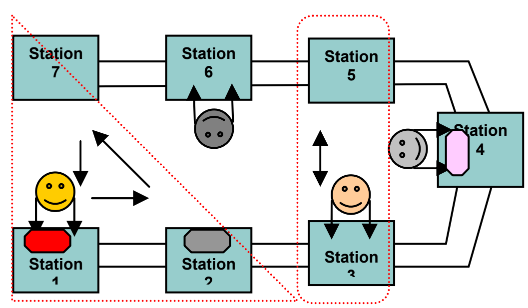


Figure 7. Sequence of Stations in a Walk-path

5. This constraint is developed to balance worker times by driving the maximum of the worker time to a minimum. The first sum (#1) calculates all the task processing times, the second sum (#2) calculates the walking time from station to station, and the third sum (#3) calculates return time after completing one cycle. The sum of all three gives the overall walk-path time for each worker.

$$\underbrace{\sum_{j=1}^{M} Tj * Yje}_{(1)} + \underbrace{\sum_{j=1}^{M-1} \sum_{g=(j+1)}^{M} Djg * IMjge}_{(2)} + \underbrace{\sum_{j=2}^{M} \sum_{g=1}^{(j-1)} Djg * Bjge}_{(3)} \leq \max T \qquad \forall e = 1 \ldots N$$

Numerical Example:

Figure 8 illustrates the breakdowns for walk-path time. Worker A is responsible for Stations 1, 2, and 7. Processing times at these stations are 5, 3, and 2 minutes respectively. Walking times are 1, 2, and 1 minutes from Station 1 to 2, 2 to 7 and 7 to 1 respectively. Hence, overall walk-path time for worker A is (5+3+2)+(1+2)+(1)=14 minutes.

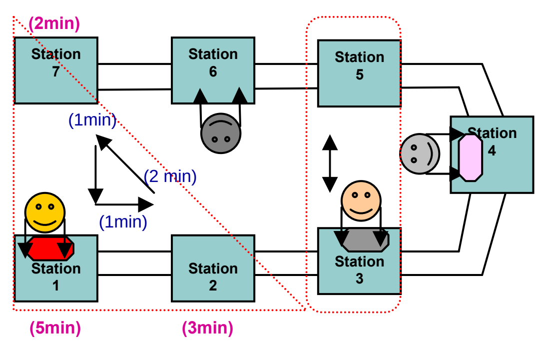


Figure 8. Walk-path Time

6. This equation is referred as a label constraint. It shows total walk-path time for each worker. The walk-path time comprise "task processing time at each station" and "walking times between stations".

$$\text{Wtime}\,(\text{e}) \;=\; \sum_{j=1}^{M} Tj * Yje \;+\; \sum_{j=1}^{M-1} \; \sum_{g=(j+1)}^{M} Djg \;\; * IMjge \;+\; \sum_{j=2}^{M} \; \sum_{g=1}^{(j-1)} Djg \;\; * Bjge \qquad \forall e = 1 \ldots N$$

4.3.3 Linearity Metric - R^2

The main objective of the thesis is to determine the satisfactory cell design for different numbers of workers in order to achieve proportional throughput per worker. Linearity metric (R^2) is employed as a measurement of the proportionality. A scatter plot is charted between throughput and the number of workers in order to be able to compute R^2. Then, a linear trendline is tried to fit on the points by setting the intercept at "0". Trendline that passes from origin shows the percent of proportionality between two variables i.e. throughput and the number of workers. It is necessary to set the intercept at "0" since the goal is to determine the proportionality. When the intercept is included, the result only explains if the points are linear.

In this study, extreme points i.e. N=1 and N=M are eliminated in the linearity metric calculation because designing a production line for only 1 worker is not practical, and having M workers will not bring flexibility to production. It is experienced that computing R^2 for the most favorable number of workers helps to reach satisfactory results earlier than including all the workers ranging from 2 to (M-1).

Numerical Example:

The production manager would like to design a u-shape production cell for product X. It requires 21 tasks to manufacture product X. However, the company challenges with very unsteady environment. Demand rate and the available number of workers per day fluctuate frequently. Hence, they would like to find a design that will minimize the loss in throughput per person. Because of the space constraint, a maximum of 6 stations can be allowed.

In this example the model is set up for 6 stations. After running Stage-1, the following output is achieved.

Station Times (sec)
Stime[1] = 26.00
Stime[2] = 13.00
Stime[3] = 15.00
Stime[4] = 29.00
Stime[5] = 32.00
Stime[6] = 15.00

Output from Stage-1 is inputted to Stage-2. The next step is to perform the Stage-2 for each number of workers ranging from 2 to 6. The results of 5 runs are listed. When performing the Stage-2 for a different number of workers, the station times remain same. Only the number of workers is altered.

Walk-Path Time in Each Configuration (sec)				
N=2	**N=3**	**N=4**	**N=5**	**N=6**
Wtime[1] = 75.00	Wtime[1] = 45.00	Wtime[1] = 41.00	Wtime[1] = 26.00	Wtime[1] = 26.00
Wtime[2] = 63.00	Wtime[2] = 47.00	Wtime[2] = 36.00	Wtime[2] = 15.00	Wtime[2] = 13.00
	Wtime[3] = 46.00	Wtime[3] = 32.00	Wtime[3] = 29.00	Wtime[3] = 15.00
		Wtime[4] = 29.00	Wtime[4] = 32.00	Wtime[4] = 29.00
			Wtime[5] = 32.00	Wtime[5] = 32.00
				Wtime[6] = 15.00
***maxT*=75**	***maxT*=47**	***maxT*=41**	***maxT*=32**	***maxT*=32**

The third step is to calculate the throughput (TH) in each configuration. TH is calculated by dividing "available working time" to "*MaxT*". It is assumed that the available working time per shift is 422 minutes. Throughput results are as follows:

N	***MaxT***	**TH (unit / shift)** ***TH= (422*60) / MaxT***
2	75	338
3	47	539
4	41	618
5	32	791
6	32	791

The last step is to calculate *Linearity Metric-R^2* to decide if the results are acceptable. It is advised to use the most favorable number of workers in R^2 computation. Two different charts are presented to explain the reason. Chart 1 includes all the workers $e, e \in N \; N = \{e / e = 2,3,4,5,6\}$, while chart 2 only includes the workers *e,* $e \in N \; N = \{e / e = 2,3,4,5\}$. R^2 values are 0.83 and 0.96 respectively. The larger the R^2, the better the results are. The production manager will not find the first result i.e. R^2= 0.86 satisfactory since threshold R^2 value for this product is set at 90 percent. As the current iteration is resulted with lower R^2 value than threshold, the manager will continue to iterate Stages 1 and 2 until a 90 percent or greater result is achieved. On the other hand, current iteration is able to achieve an R^2 value greater than threshold if the computation is performed for most favorable number of workers i.e. N= [2, 5] (see Chart 2). Performing the computation for the most desirable number of workers decreases the iteration steps.

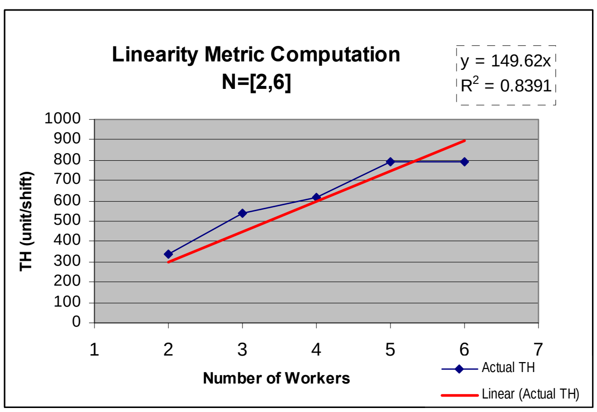


Chart 1 Linearity Metric Computation, N= [2,6]

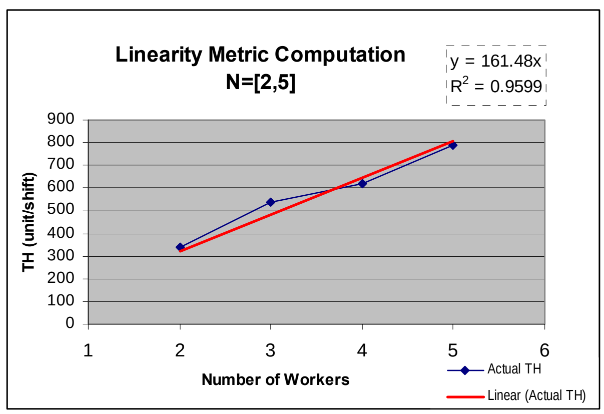


Chart 2 Linearity Metric Computation, N= [2,5]

4.4 Computational Time and Iteration Steps Reduction Methods

The size of the problem and the computational time were major challenges at modeling phase. Initial goal was to develop one stage model that would be capable of performing both task-to-station and worker-to-station assignments for various number of workers simultaneously with the objective of achieving proportional throughput. This approach was only successful for the small problems. In order to be able to manipulate medium and larger size problems, the mathematical model is separated into two stages.

Staging method helped to improve the computational time issues significantly for small and medium size problems, but new techniques were implemented to decrease the solving time for larger size problems and the iteration steps. The cutting plane method is developed to reduce solving time, and the percentage method is created to decrease the number of iterations.

4.4.1 Cutting Plane Method

Stage II takes a considerably longer time for larger size problems ($M>13$ stations). For instance, a problem with 19 stations may require a week or two weeks to come up with a feasible solution. One way of improving computational time is to decrease the number of nodes in the search tree. Adding some of the predetermined worker-to-stations assignments into the model as new constraints helps to minimize the number of nodes in investigation. The critical factors are crossover and the maximum cycle time in the development of the assignments. Fixing some assignments manually do not affect the solution that the program will reach, because pre-assignments are developed considering the same constraints as the

mathematical model does. The following steps explain the process of determining the assignments manually.

Steps to create cutting planes:

Step 1: Run Stage-1 for the specified number of stations. In this example, the model is run for 13 stations.

Step 2: Document *Station Time* results from Stage-1

Example:

Stime[1] = 7.00	Stime[8] = 12.00
Stime[2] = 14.00	Stime[9] = 12.00
Stime[3] = 7.00	Stime[10] =13.00
Stime[4] = 9.00	Stime[11] = 10.00
Stime[5] = 9.00	Stime[12] = 9.00
Stime[6] = 8.00	Stime[13] = 12.00
Stime[7] = 3.00	

Step 3: Draw a u-shape layout and put the station times at each box. Every box defines one station. Calculate total processing time by side.

Example:

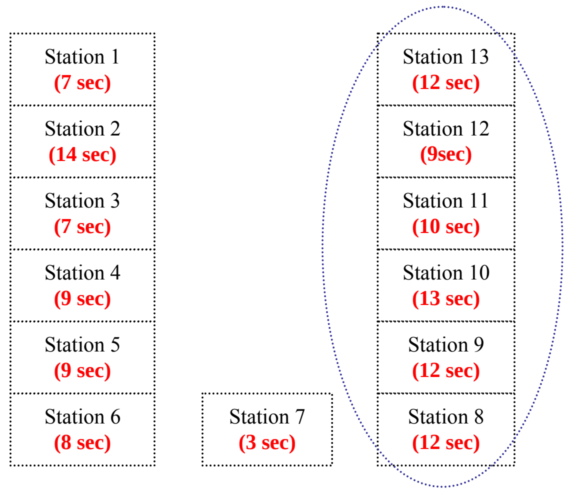
Station 1 **(7 sec)**		Station 13 **(12 sec)**
Station 2 **(14 sec)**		Station 12 **(9 sec)**
Station 3 **(7 sec)**		Station 11 **(10 sec)**
Station 4 **(9 sec)**		Station 10 **(13 sec)**
Station 5 **(9 sec)**		Station 9 **(12 sec)**
Station 6 **(8 sec)**	Station 7 **(3 sec)**	Station 8 **(12 sec)**
Total Processing Time by side: **54 sec**		**68 sec**

Step 4: Run Stage-2 for N=2 workers. Save objective value.

Objective Value: 73.00
Wtime[1] = 65.00
Wtime[2] = 73.00

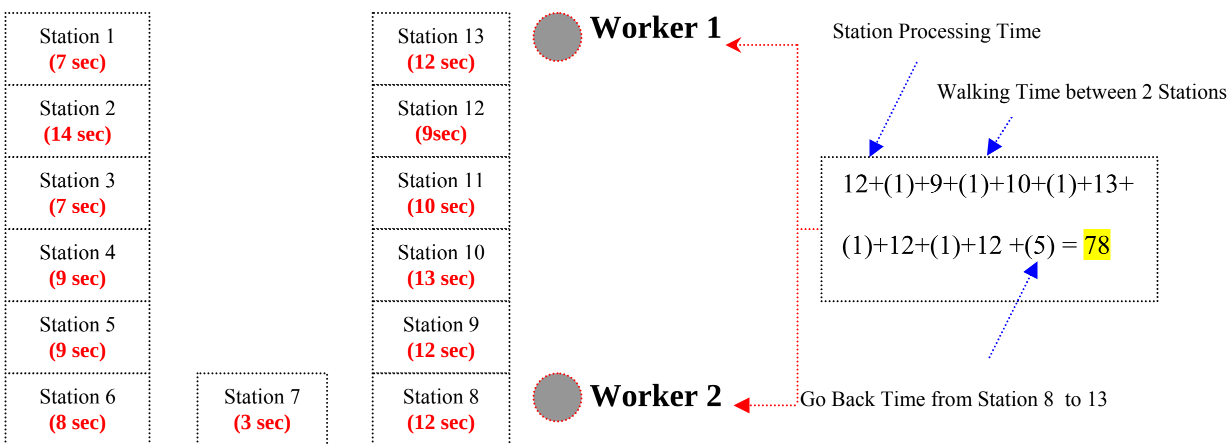
Step 5: Objective value decreases when the number of workers increases as it is the function of the cell cycle time. Cycle time drops when more workers are employed. Hence, the objective values of the iterations for N >2 will be less than or equal to 73sec. In the next iteration i.e. N=3, a worker will be assigned to the stations until the walk-path time reaches to 73sec. In addition, no crossovers are allowed in the assignment process.

One of the possible stations is tried to be fixed in the walk-path for as many workers as it is possible. A basic rule for fixing one station is to begin from the side that has the largest sum of the processing times. In this example, it is the right-hand side that adds up 68 sec.



Total Processing Time by side: **54 sec** **68 sec**

Step 6: Begin to assign workers to stations from the selected side. In this example worker 1 is assigned to Station 13, while worker 2 is assigned to Station 8.



As sum of the station times and walking times between Station 8 and 13 is greater than 73sec (i.e. upper bound for walk-path time), different workers are able to assign to these stations.

Workers cannot cross each other, thus the only way of assigning Stations 8 and 13 to the same worker is to assign all of the stations between 8 and 13 to the same worker as

long as it complies with the upper bound objective function constraint from the previous iteration.

Step 7: Enter manually assigned constraints into the model, and increase the number of workers from 2 to 3 and run Stage 2. Y[j,e] is the decision variable for assigning station *j* to worker *e*.

Example: *1st constraint:* Y[13,1]=1;

2nt constraint: Y[8,2]=1;

3rd constraint: $maxT \leq 73$;

The model will establish the assignments for 3 workers. Since N < M, one worker will be responsible for multiple stations. With the cutting plane method, one of the stations in the walk-path is fixed manually for some workers before running the mathematical model. Other stations in the walk-path will be determined by the program. Cutting plane method does not affect the optimal assignments.

Step 8: Once the model is completed the run for N=3, the same process is repeated until N=M-1. Table 4 illustrates the worker-to-stations assignments that are fixed manually for different objective values (i.e. *maxT*).

Table 4 Stations that can be fixed manually to different workers for M=13 stations

N	MaxT	Manually Fixed Stations with Different Workers
2	73	Y[1,1]
3	48	Y[8,1], Y[13,2]
4	37	Y[8,1], Y[13,2]
5	32	Y[8,1], Y[13,2], Y[10,3]
6	26	Y[8,1], Y[13,2], Y[10,3], Y[2,4]
7	24	Y[8,1], Y[13,2], Y[10,3], Y[2,4]
8	21	Y[8,1], Y[13,2], Y[10,3], Y[2,4], Y[9,5]
9	20	Y[8,1], Y[13,2], Y[10,3], Y[2,4], Y[9,5]
10	19	Y[8,1], Y[13,2], Y[10,3], Y[2,4], Y[9,5]
11	18	Y[8,1], Y[13,2], Y[10,3], Y[2,4], Y[9,5]
12	14	Y[8,1], Y[13,2], Y[10,3], Y[2,4], Y[9,5], Y[12,6]

As illustrated in Table 4, more stations can be fixed manually when both the number of workers increases and the objective function drops. Fixing many stations in advance improve the solving time dramatically.

4.4.2 Percentage Method

It is experienced that the percentage of workload at every station has significant impact on the result. Identifying the optimal proportions of work at stations can help to reduce the number of iterations in order to reach an acceptable solution.

Current mathematical model may require multiple iterations for a desirable solution. Multiple iterations can be a major problem for larger size problems since each step for large problems takes considerably long time.

Percentage method is developed to improve the computational time by decreasing the number of iterations. The motivation in this method is to iterate one selected problem until a satisfactory result is accomplished. Then, calculate the percentage of assignments at every station. These percentages will be established as standards for different problems that require *same number of stations*. When a different problem is run, established standards will be incorporated to Stage-1. This time, instead of balancing the station times, program will try assigning the tasks to stations up to the percentage levels specified in the model for every station.

After running multiple data, it is proved in the analysis section that majority of the experiments accomplished satisfactory results through Percentage Method (refer analysis section for details).

Percentage Method Steps:

Step 1: Select a small or medium size problem to establish the proportions at every station. A small size problem is desired since it requires less time to compute, and creates

an opportunity for multiple iterations. More iteration may lead better solutions. This thesis mainly used data sets with 21 to 35 tasks when running the experiments.

Step 2: Determine the number of stations. To test the efficiency of Percentage Method, this research used 10 samples for each configuration from 5 to 16 stations and 5 samples for the configurations with 17,18 and 19 stations.

Step 3: Run Stage 1. Document the station times.

Step 4: Proceed to Stage 2. Enter the essential inputs i.e. number of workers, station times, and walking times.

Step 5: Compute R^2 for current iteration. Determine if R^2 is satisfactory.

Step 6: If R^2 is not acceptable, go back to Stage 1. Relax the balancing constraint. Then, repeat Steps 4 and 5 until a satisfactory result is achieved.

Step 7: If an acceptable solution is reached, stop. Go back to Stage 1 output (i.e. *Station Times*) from last iteration. Calculate the percent of workload each station assigned.

Step 8: Document the percentages and save the results for different problems.

Step 9: When a new problem is tested, modify Stage 1. Comment the balancing constraints (i.e. Constraints 4 at Stage 1). Insert the percentages determined at Step 8.

Step 10: Run Stage 1 with the modified constraints. Document *Station Times.*

Step 11: Proceed to Stage 2. Enter the essential inputs. No modifications are made at Stage 2.

Step 12: Compute R^2, and determine if it is acceptable. If acceptable, process is done. Otherwise, go back to Step 3 and repeat the loop until Step 6. If a satisfactory result is not reached after 1st iteration, it implies that percentages computed at Step 8 are not good fit for this problem. To achieve a desired solution for the specific problem, original model

(model without percentages) should be used, and Stage 1 and 2 should be repeated as many times as it requires.

Numerical Example:

Stage I Output	**Station Times (min)**	**% of overall workload**	**R^2 Value**
Stime[1] = 8.00	8	0.17	0.96
Stime[2] = 5.00	5	0.11	
Stime[3] = 8.00	8	0.17	
Stime[4] = 8.00	8	0.17	
Stime[5] = 3.00	3	0.07	
Stime[6] = 5.00	5	0.11	
Stime[7] = 5.00	5	0.11	
Stime[8] = 4.00	4	0.09	
Total	**46**	**1.00**	

Stage-1 Modification:

Three constraints as follows will not be used in the percentage method.

$$\sum_{i=1}^{T} Di * Xij - \sum_{i=1}^{T} Di * Xig \leq Maxf \qquad \forall j = 1...(M-1) \qquad .... (1)$$

$$\forall g = (j+1)...M$$

$$\sum_{i=1}^{T} Di * Xig - \sum_{i=1}^{T} Di * Xij \leq Maxf \qquad \forall j = 1...(M-1) \qquad .... (2)$$

$$\forall g = (j+1)...M$$

$Maxf$ < K (K is the bound for relaxing) .... (3)

The following constraints are added to the model. Percentage values are subject to change according to configuration or trial results.

Station 1:

$$Stime[1] - ((\sum_{i=1}^{T} Di) * 0.17) \leq Maxf \qquad ((\sum_{i=1}^{T} Di) * 0.17) - Stime[1] \leq Maxf$$

Station 2:

$$Stime[2] - ((\sum_{i=1}^{T} Di) * 0.11) \leq Maxf \qquad ((\sum_{i=1}^{T} Di) * 0.11) - Stime[2] \leq Maxf$$

Station 3:

$$Stime[3] - ((\sum_{i=1}^{T} Di) * 0.17) \leq Maxf \qquad ((\sum_{i=1}^{T} Di) * 0.17) - Stime[3] \leq Maxf$$

Station 4:

$$Stime[4] - ((\sum_{i=1}^{T} Di) * 0.17) \leq Maxf \qquad ((\sum_{i=1}^{T} Di) * 0.17) - Stime[4] \leq Maxf$$

Station 5:

$$Stime[5] - ((\sum_{i=1}^{T} Di) * 0.07) \leq Maxf \qquad ((\sum_{i=1}^{T} Di) * 0.07) - Stime[5] \leq Maxf$$

Station 6:

$Stime[6] - ((\sum_{i=1}^{T} Di\ ) * 0.11) \leq\ Maxf \qquad ((\sum_{i=1}^{T} Di\ ) * 0.11) - Stime[6] \leq\ Maxf$

Station 7:

$Stime[7] - ((\sum_{i=1}^{T} Di\ ) * 0.11) \leq\ Maxf \qquad ((\sum_{i=1}^{T} Di\ ) * 0.11) - Stime[7] \leq\ Maxf$

Station 8:

$Stime[8] - ((\sum_{i=1}^{T} Di\ ) * 0.09) \leq\ Maxf \qquad ((\sum_{i=1}^{T} Di\ ) * 0.09) - Stime[8] \leq\ Maxf$

Diagram 3 explains the steps used in developing the percentages for each configuration.

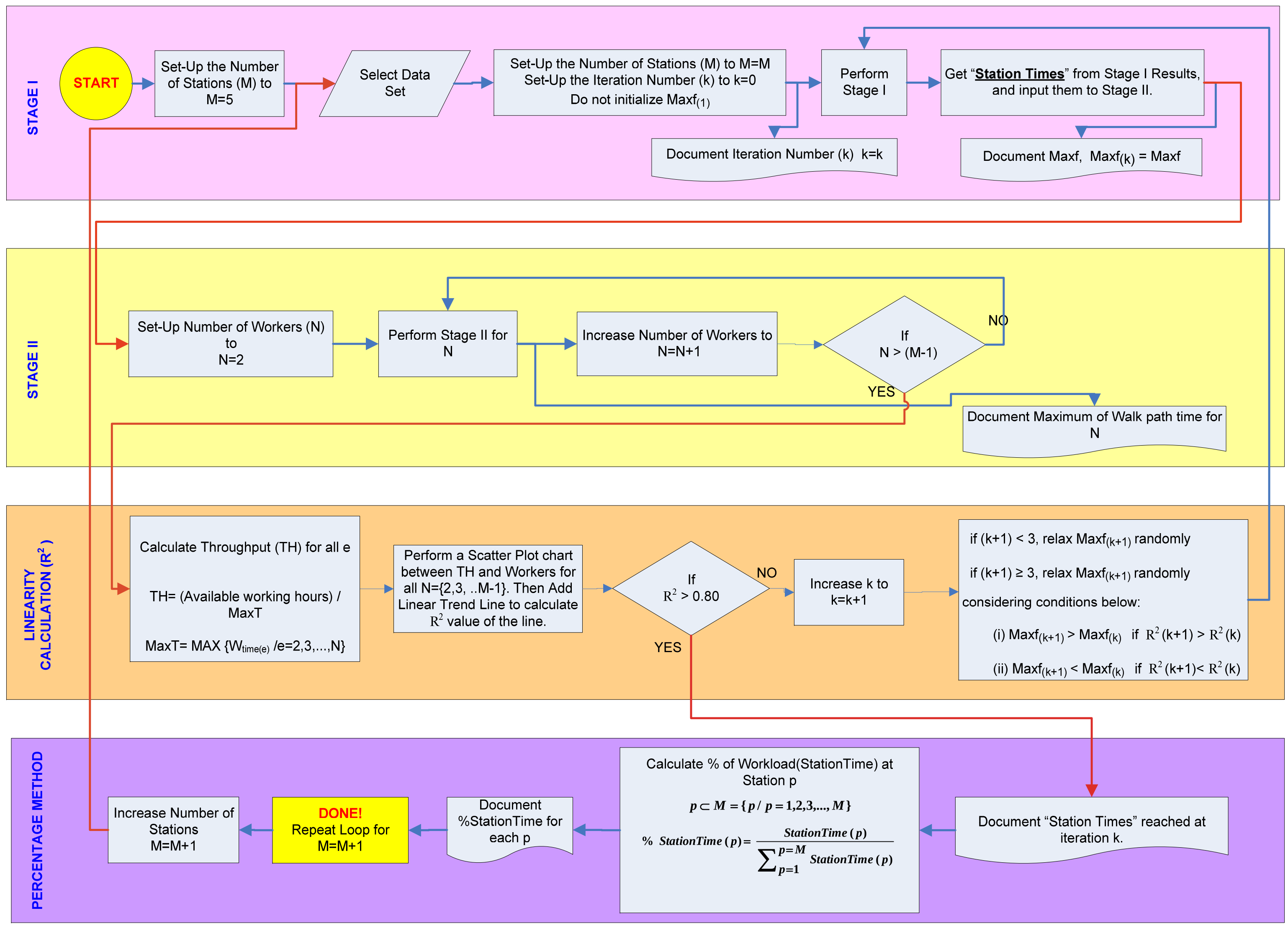


Diagram 3. Percentage Method Process Flow

5 Data Collection and Analysis

Model efficiency is tested by running multiple different data sets for each number of stations ranging from 5 to 19. Following areas are analyzed in this paper.

1. Solving Time
2. Linearity Metric-R^2 results
3. Worker utilization
4. Throughput results

5.1 Data Collection

Both Delphi real world data and data from previous research are used to test the efficiency of the model. Delphi data is shown in Appendix C. Data from the literature is obtained from the line balancing webpage (http://www.wiwi.uni-jena.de/Entscheidung/alb/index.htm) that Scholl and Klein prepared. Those are the collection of data sets used in the literature for both simple and u-line balancing problems. This research used 10 out of 25 different data sets presented in the website (see Appendix C). Each data set consists of task processing time and precedence diagrams.

Analysis is conducted for $\forall m \in M$, M= [5, 19] for $\forall n \in N$, N= [2, M-1]. Extreme points (i.e. N=1 and N=M) are eliminated in the calculation of R^2. Sample size is set to 10 for small and medium size problems i.e. M= [5, 16], and larger size problems i.e. M= [17, 19] are tested with 5 samples. Total of 135 configuration with 1275 runs {*10*(N-2)* for all N= [5, 16] + *10*(N-2)* for all N= [17, 19] + *1*(19-5+1)* for Stage-1 runs} are completed.

All experiments are conducted on Intel Pentium 4CPU, 3.20GHz, and 1 GB of

RAM computers. Table 5 summarizes data collection. Row defines the number of stations, and column indicates the sample number with a description of the data set used in each configuration. Parenthesis in each name show the number of tasks associated with the data set.

Table 5. Summary of Data Sets Used in the Analysis

<table>
<tr><th></th><th colspan="10">Sample #</th></tr>
<tr><th>Station#</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>10</th></tr>
<tr><td>5</td><td rowspan="12">Mitchell (21)</td><td rowspan="12">Roszieg (25)</td><td rowspan="12">Sawyer (30)</td><td rowspan="12">Gunther (35)</td><td rowspan="12">Kilbrid (45)</td><td rowspan="12">Hahn (53)</td><td rowspan="12">Warnecke (58)</td><td rowspan="12">Tonge (70)</td><td rowspan="12">Wee-Mag (75)</td><td rowspan="12">Delphi (79)</td></tr>
<tr><td>6</td></tr>
<tr><td>7</td></tr>
<tr><td>8</td></tr>
<tr><td>9</td></tr>
<tr><td>10</td></tr>
<tr><td>11</td></tr>
<tr><td>12</td></tr>
<tr><td>13</td></tr>
<tr><td>14</td></tr>
<tr><td>15</td></tr>
<tr><td>16</td></tr>
<tr><td>17</td><td rowspan="3">Sawyer (30)</td><td rowspan="3">Gunther (35)</td><td rowspan="3">Warnecke (58)</td><td rowspan="3">Tonge (70)</td><td rowspan="3">Delphi (79)</td><td rowspan="3" colspan="5"></td></tr>
<tr><td>18</td></tr>
<tr><td>19</td></tr>
</table>

As a starting point, selected data sets are used to calculate the best percentage of overall workload in each configuration. Then, the entire experiment is conducted with relative percentages in order to prove that this method can produce good results with different data sets.

5.1.1 Solving Time

The model developed in this thesis is solved with OPL Studio using CPLEX program. A branch and cut model is used in searching for an optimal solution. Varied data may take a different length of run times to reach the optimal solutions depending on the search tree.

Stage-1 can be stopped anytime when a best solution is reached if Stage-1 is not modified with the percentage method. Otherwise, it is advised to wait for an optimal solution. In this research, all data set are run with percentage method. While 50% of the data set reaches to an optimal solution, the rest is stopped at best solution. Terminating the time at best solution varies; however, we tried to end it as close as the objective value reaches zero or best solution remains same for quite long time. On the other hand, Stage-1 solving time is not as significant as Stage-2 since any tasks-to-station configuration may yield a good solution. Hence, Stage-1 is called as an iterative step.

Stage-2 solving time results are illustrated in Table 6. The table represents the average computational time to reach the optimal solution in minutes for each number of workers with a different number of stations using the cutting plane and percentage method together. The value in each cell in the table is the average of 10 or 5 samples depending on the configuration. Column specifies the station number, and row indicates the number of

workers. It wasn't possible to collect run times for N= [16, 18] in 19 stations configuration, because the computers used in this experiment did not have enough memory to process those data sets. However, the optimal solution for those configurations is manually calculated by modifying the result of the previous level. The last two rows in the table denote the overall time needed to calculate R^2 value for each number of stations.

Table 6. Stage II Solving Time Results Using Cutting Plane & Percentage Method (min)

	STAGE-2 SOLVING TIME (MIN)														
	Number of Stations														
Number of Workers	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**
2	<1	<1	<1	<1	0.01	0.01	0.02	0.02	0.04	0.05	0.09	0.10	0.16	0.20	0.30
3	<1	<1	0.01	0.01	0.02	0.05	0.07	0.13	0.26	0.37	0.76	1.02	2.48	2.55	4.70
4	<1	<1	0.01	0.04	0.05	0.15	0.36	0.44	0.73	1.00	2.49	2.45	14.93	9.50	18.60
5		<1	0.02	0.07	0.10	0.64	0.92	1.06	0.49	1.81	7.21	5.54	16.81	21.64	129.01
6			<1	0.07	0.31	0.36	0.62	0.24	0.73	3.26	18.02	16.54	142.63	260.94	221.79
7				0.15	0.03	0.38	0.22	0.29	0.68	1.82	41.33	49.28	347.92	302.98	344.28
8					0.03	0.12	0.43	0.24	0.44	0.39	14.36	17.30	69.50	694.92	1244.32
9						0.17	0.06	0.45	0.10	0.86	11.35	0.37	29.38	291.86	256.33
10							0.04	0.09	0.11	0.67	3.64	0.71	23.28	24.29	722.35
11								0.05	0.16	1.13	1.11	0.76	17.17	21.79	98.62
12									0.07	1.26	1.51	0.28	1.12	51.40	6.71
13										0.25	1.73	0.21	0.41	4.07	1.81
14											2.76	0.26	1.07	1.04	6.03
15												0.56	0.61	2.38	9.81
16													0.98	8.96	17.34
17														12.53	N/A
18															N/A
SUM (min)	<1	<1	<1	0.3	0.6	1.9	2.7	3.0	3.8	12.9	106.4	95.4	668.5	1711.0	3082.0
SUM (hr)	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.1	0.1	0.2	1.8	1.6	11.1	28.5	51.4

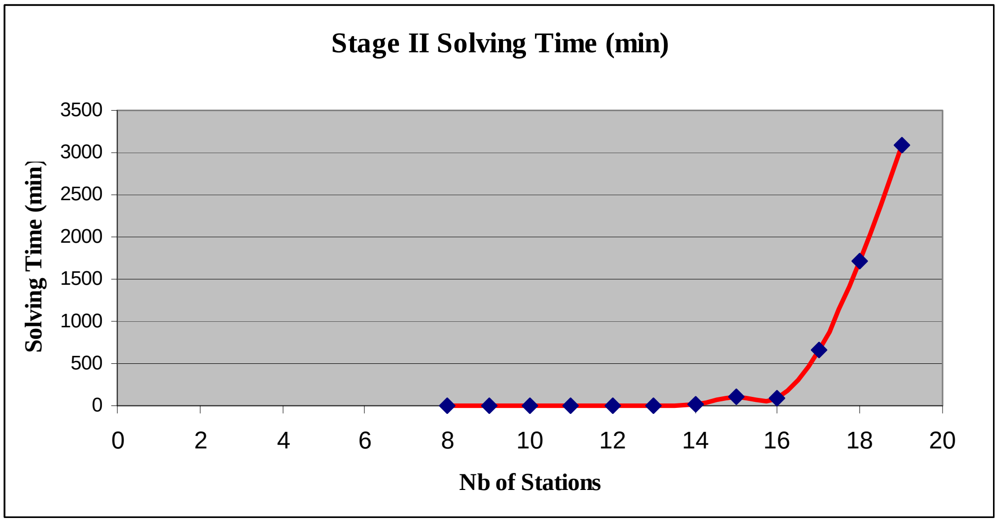


Chart 3. Stage 2 Solving Time Plot with Exponential Growth

As per Table 6 and Chart 3, it can be concluded that solving time increases exponentially when the number of stations increase. The number of stations is the major factor in computational time. According to experiment results, it proves that the mathematical model is capable of solving problems in seconds if the station number is less than 9; in less than 5 minutes if the station number is between 10 and 13. It can take longer when the number of stations is greater than 13.

Table 7. Contribution of Solving Time Reduction Methods

Station #	Cutting Plane (-) Percentage Method (-) (min)	Cutting Plane (+) Percentage Method (+) (min)	Improvement
5	0.005	0.0001	50
6	0.03	0.0002	126
7	0.09	0.0008	115
8	0.83	0.01	83
9	19.96	0.01	1996
10	224	0.03	7467
11	77.5	0.05	1550
12	1200	0.05	24000
M>13	Unable to run	0.06	

Table 7 demonstrates the significant contribution of time reduction techniques (cutting plane and percentage methods) to the original model. It was unattainable to perform the middle and larger size problems with the original model. Hence, the comparison between original model and the time reduction techniques is performed for small size problems. This signifies that medium and larger size problems can be handled by employing these techniques.

The second column shows the average solving time needed to reach optimality using the original model. The third column illustrates the same results using the cutting plane and percentage method together. The fourth column shows how much improvement is achieved with the new techniques. For example, an optimal solution can be reached 50 times faster with time reduction methods than with the original model. It is concluded from Table 7 that time reduction techniques are more successful for medium and larger size problems

In addition, solving time with the original model is not only the function of workers. The value of *Maxf*, the variable that determines the work load balance between stations at Stage-1, has contribution on solving time as well. Solving time decreases when *maxf* increases within the same data set. Table 8 exemplifies that computational time takes 224 minutes for Station 10 while it only takes 77.5 minutes for Station 11. *Maxf* for Station 10 is relatively small. Table 8 demonstrates the differences. The percentage method determines the best *maxf* that can lead to achieve a high R^2 value and low solving time.

Table 8 Effect of *Maxf* on Stage-2 Solving Time in Original Model

Iteration #	Data Name	Maxf	Station #	Total Solving Time (min)
28.1	Mitchell	2	10	372
28.2	Mitchell	8	10	76
29.1	Mitchell	3	11	161
29.2	Mitchell	8	11	70
29.3	Mitchell	15	11	2

Contribution of Cutting Plane to Solving Time:

Cutting plane has a significant contribution in reduction of computational time. Number of workers was a significant factor until the implementation of cutting plane into the model. Chart 4 demonstrates that solving time changes with number of workers before and after the cutting plane method is implemented.

Chart 4 is plotted using a data set for 12 stations. The X axis represents the number of workers while the Y axis shows the solving time in seconds. The graph on the left panel demonstrates how solving time increases with the number of workers before using the

cutting plane technique. The chart on the right side explains that solving time increases in the beginning and decreases substantially after a certain point when the cutting plane method is used. Moreover, solving time is 100 times lower with this method.

This type of behavior is seen with any data and any configuration. The main reason for that is more stations and workers can be fixed manually when N increases. Fixing more stations helps to decrease the number of nodes in the search tree.

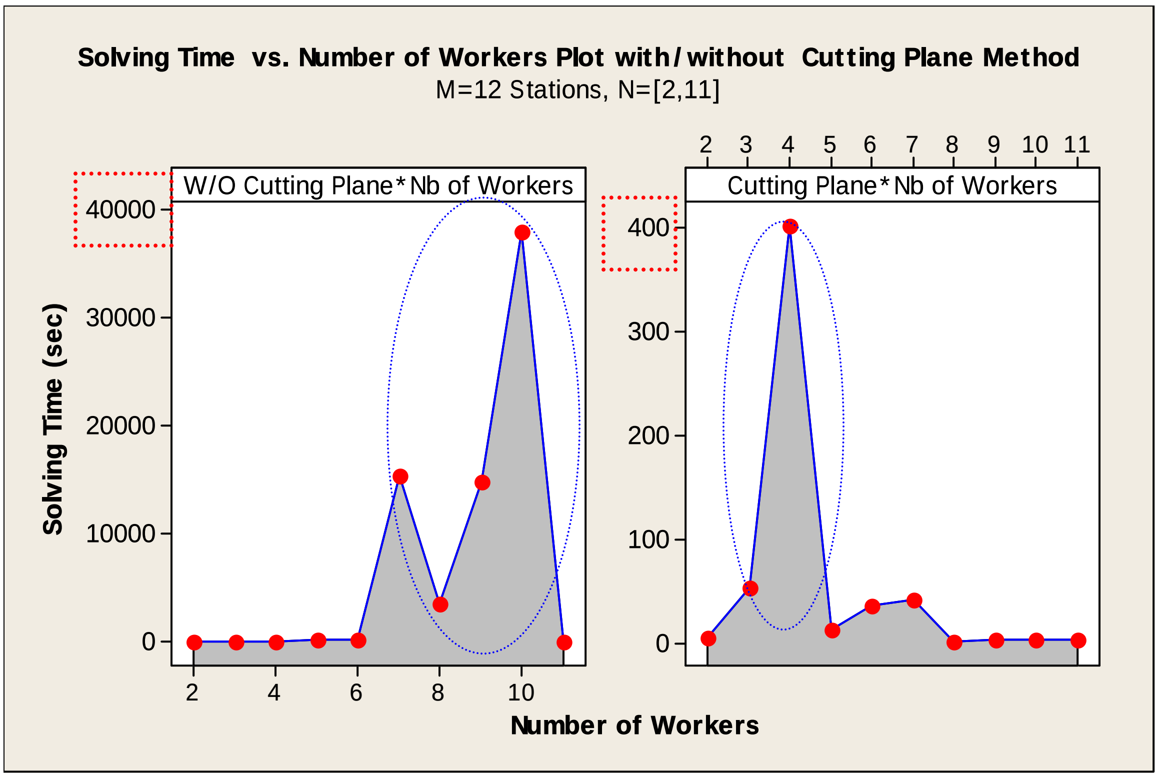


Chart 4 Stage II Solving Time vs. Number of Workers Plot with and without Cutting Plane method

Contribution of Percentage Method in Solving Time:

The main contribution of the percentage method is to decrease the number of iterations. 81% of the experiments result with an R^2 value greater than 0.90 after the *first* run, 13% of them achieved an R^2 value between 0.84 and 0.90 after the *first* run, and only 7% showed poor results. Due to time limitations, no further iterations were conducted for the poorer results.

Cutting plane has the most significant impact on time reduction. However, the percentage method showed some contribution to decrease the solving time. Table 9 shows how fast the optimal solution can be reached if the percentage method is used. Cutting plane is used for both, but the percentage method is altered.

Table 9. Contribution of Percentage Method in Stage II Solving Time

Station #	**Percentage Method (-) Cutting Plane (+) (min)**	**Percentage Method (+) Cutting Plane (+) (min)**	**Improvement**
N≤11	no significant difference	no significant difference	N/A
12	9	3	3
13	22	4	6
14	31	13	2
15	325	106	3
16	729	95	8
17	4765	668	7
18	3782	1711	2
19	7122	3092	2

5.1.2 Linearity Metric-R^2

Linearity metric is the measurement of proportionality. The higher the R^2 value is, the higher the possibility of achieving the proportional throughput per person in each configuration. The R^2 metric takes a value between 0 and 1. $R^2 = 1$ denotes the perfect linearity in which the throughput per person is equal for any number of workers, Linearity metric is calculated for all the configurations between 5 to 19 stations. Table 10 presents the results.

Table 10. Linearity Metric –R^2 Results by Configuration

	$R^2 \geq 0.90$			$0.90 > R^2 \geq 0.80$			$R^2 < 0.80$			
Station #	Average	Number of Occurrence	% of Occurrence	Average	Number of Occurrence	% of Occurrence	Average	Number of Occurrence	% of Occurrence	Total Number of Samples
5	0.95	10	100%							10
6	0.98	10	100%							10
7	0.95	10	100%							10
8	0.96	10	100%							10
9	0.94	2	20%	0.87	8	80%				10
10	0.96	10	100%							10
11	0.96	10	100%							10
12	0.95	10	100%							10
13	0.94	9	90%	0.80	1	10%				10
14	0.92	7	70%	0.85	2	20%	0.76	1	10%	10
15	0.92	9	90%				0.65	1	10%	10
16	0.94	6	60%	0.85	2	20%	0.50	2	20%	10
17	0.92	2	20%	0.85	2	20%	0.52	1	10%	5
18	0.94	3	30%	0.86	1	10%	0.71	1	10%	5
19	0.98	1	10%	0.82	1	10%	0.64	3	30%	5
TOTAL	0.94	108	81%	0.84	17	13%	0.64	9	7%	135

Three levels of R^2 value are analyzed in Table 10: **(1)** $R^2 \geq 0.90$, **(2)** $0.90 > R^2 \geq 0.80$, **(3)** $R^2 < 0.80$. The *average column* shows the mean of all the results per R^2 level. Results are also categorized by stations. The *number of occurrence column* shows the number of samples achieved the specified R^2 range. The *percent of occurrence column* calculates the percentage of occurrence out of the total number of samples. *Total line* represents the average of all data without considering the number of stations.

After the first iteration 81% of the experiments achieved 94% linearity; 13% of experiments accomplish 84% linearity. 7% of the experiments showed poor results. As illustrated in Table 10, poor results occur for larger size problems. It is because small number of iterations was run to calculate the right percentages due to high computational time requirements. If the result is not satisfactory, problem is advised to run without percentage method as much iteration as required.

5.1.3 Worker Utilization

Achieving high linearity is desired; however, balanced workload between workers is also critical in a production cell design. Over or under utilized workers impact the efficiency of the production. It is difficult to design a well-balanced line for various numbers of workers. Hence, it requires a trade off between dynamic environment and stable environment. It is less complicated to balance the line for a stable environment in which the number of workers is fixed.

Utilization is computed for all workers in each configuration. The following example shows the formula used in this research to calculate the utilization.

Numerical Example: M=6 and N=5

Stage-2 Output	**Walk-path Time (sec)**	**U_k (Wtime [k] / Ideal Walk-path Time)**
Wtime[1] = 21.0000	21	0.80
Wtime[2] = 30.0000	30	1.14
Wtime[3] = 30.0000	30	1.14
Wtime[4] = 29.0000	29	1.10
Wtime[5] = 22.0000	22	0.83
Ideal Walk-path Time :	**26.4**	

Maximum utilization difference = 1.14-0.80 = 0.34

Ideal walk-path time is accomplished when all the walk-path time is perfectly balanced. In this example, the ideal walk-path time is 26.4 seconds that is the average of all the walk-path times. Due to task precedence constraints and balancing the production cell for a variable number of workers, it is difficult to achieve ideal time.

Table 11 represents the maximum utilization difference between U_k and U_r in a production cell with M stations. U_k is defined as k^{th} worker's utilization with M number of stations.

$$U_k \in U,\ U = \{U_k / k = 2,3,\ldots,M\} \quad M = \{5,6,\ldots,19\}$$
$$U_{k(max)} = Max\{U_k / k = 2,3,\ldots,M\}$$
$$U_{r(min)} = Min\{U_k / k = 2,3,\ldots,M\}$$

$U_{k(max)}$ and $U_{r(min)}$ denote the most over-utilized and under-utilized workers respectively. The difference of both gives the maximum utilization difference in the system. If the difference is small, it is considered to be a well-balanced system.

Table 11. Maximum Utilization Difference by Number of Workers and Stations

	Number of Stations														
Nb. Of Workers	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
2	0.11	0.03	0.03	0.06	0.06	0.04	0.04	0.01	0.00	0.00	0.00	0.02	0.04	0.02	0.00
3	0.09	0.24	0.12	0.10	0.02	0.02	0.11	0.05	0.04	0.00	0.03	0.01	0.03	0.02	0.03
4	0.16	0.19	0.27	0.12	0.14	0.12	0.07	0.07	0.07	0.05	0.07	0.04	0.04	0.07	0.03
5	0.96	0.14	0.19	0.23	0.22	0.15	0.06	0.03	0.05	0.07	0.04	0.03	0.12	0.06	0.04
6		1.17	0.33	0.25	0.16	0.22	0.08	0.11	0.14	0.20	0.04	0.08	0.12	0.11	0.07
7			0.85	0.43	0.21	0.33	0.22	0.30	0.26	0.04	0.05	0.06	0.12	0.09	0.18
8				0.82	0.56	0.14	0.20	0.34	0.16	0.17	0.15	0.14	0.32	0.21	0.08
9					1.08	0.30	0.25	0.21	0.25	0.25	0.24	0.21	0.23	0.20	0.09
10						0.60	0.33	0.34	0.26	0.36	0.25	0.20	0.27	0.28	0.15
11							0.86	0.49	0.46	0.24	0.33	0.16	0.25	0.37	0.33
12								0.96	0.75	0.40	0.27	0.53	0.44	0.43	0.34
13									1.19	0.48	0.44	0.63	0.56	0.29	0.57
14										0.89	0.31	0.83	0.66	0.36	0.71
15											0.78	0.91	0.73	0.49	0.69
16												1.60	0.90	0.49	0.77
17													1.28	1.12	n/a
18														1.44	n/a
19															1.49

In order to be able to read the Table 11 clearly, all the results greater than 0.25 are highlighted. As per Table 11, it is easier to achieve a balanced workload if the number of workers is lower than the number of stations. When worker number gets closer to station number, idle time between workers increases. It is because the flexibility decreases and the worker is assigned to fewer stations. Chart 4 illustrates the work balance between workers.

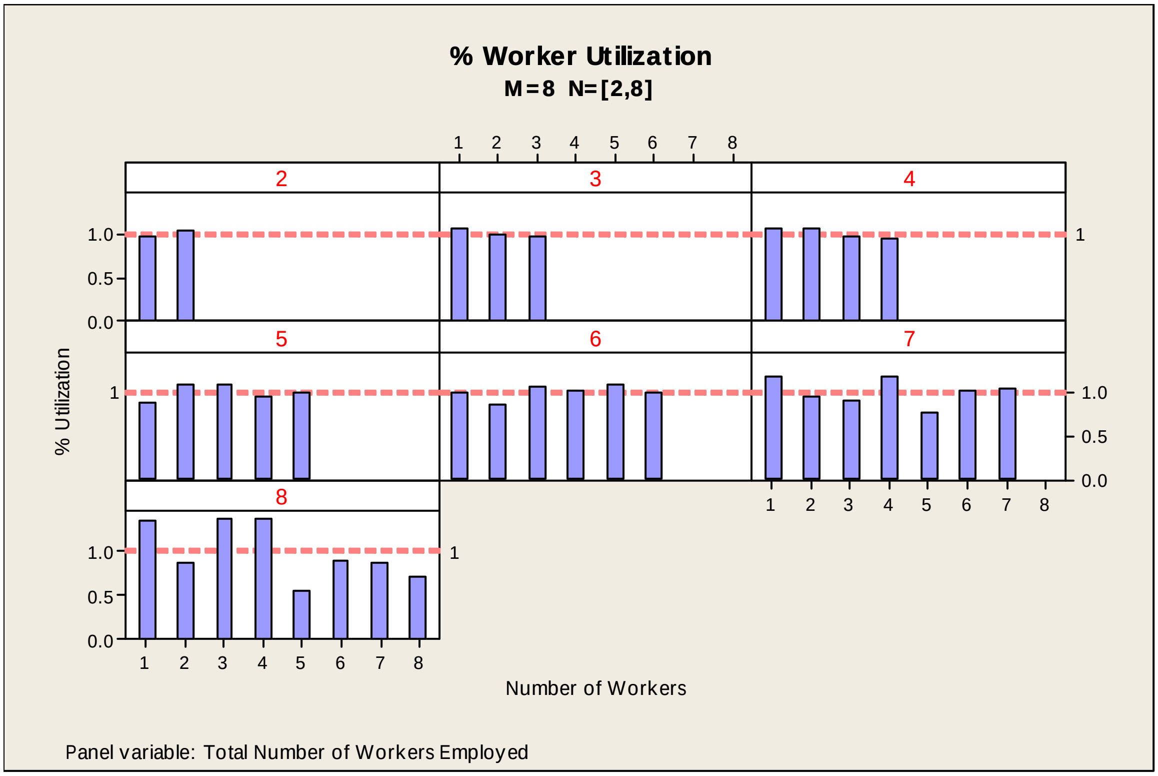


Chart 4. % Worker Utilization for M=8 Stations

5.1.4 Throughput

Throughput is another important metric in a cell design. There is a trade off between flexibility and throughput. The two goals can be achieved at the same time depending on the conditions. A numerical example is given to show the relationship.

Table 12 shows the gap between actual throughput and ideal throughput. Ideal TH is theoretical throughput where precedence constraints and walking times are not included. Actual TH is calculated using *maxT* for each configuration. Data in each cell presents the average gap by workers and stations. Gap increases when the number of workers gets closer to the number of stations.

	GAP between ACTUAL TH and IDEAL TH																
	Number of Stations																
Nb. of Workers	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	MIN GAP	MAX GAP
2	7%	6%	5%	8%	8%	9%	8%	9%	7%	9%	9%	9%	7%	6%	7%	5%	9%
3	9%	14%	10%	12%	6%	9%	10%	12%	10%	11%	10%	11%	8%	9%	8%	6%	14%
4	16%	13%	16%	12%	10%	15%	14%	12%	12%	9%	12%	13%	10%	10%	10%	9%	16%
5		13%	23%	21%	16%	17%	16%	12%	11%	13%	14%	16%	11%	10%	11%	10%	23%
6			18%	19%	19%	24%	13%	20%	19%	17%	14%	16%	15%	11%	11%	11%	24%
7				24%	21%	29%	21%	24%	22%	12%	17%	18%	14%	14%	14%	12%	29%
8					31%	23%	24%	26%	24%	18%	20%	18%	19%	13%	17%	13%	31%
9						25%	18%	29%	22%	22%	23%	23%	19%	14%	18%	14%	29%
10							26%	28%	22%	27%	25%	26%	21%	20%	16%	16%	28%
11								28%	26%	31%	28%	23%	25%	25%	20%	20%	31%
12									31%	32%	31%	27%	26%	28%	24%	24%	32%
13										33%	32%	31%	29%	33%	29%	29%	33%
14											33%	36%	34%	34%	33%	33%	36%
15												40%	38%	35%	35%	35%	40%
16													42%	29%	39%	29%	42%
17														33%	41%	33%	41%
18															44%	44%	44%

Table 12. Average Gap between Actual TH and Ideal TH

Numerical example:

Company X has a dynamic environment where the number of workers fluctuates each day. According to the historical data (data #1 and data#2), the probability of having specified number of workers each shift is demonstrated on the table. What cell configuration should be selected in order to achieve maximum throughput for this kind of environment?

	P (number of workers available/ shift)	
Number of Workers	**Data #1**	**Data #2**
2	0%	1%
3	0%	1%
4	0%	1%
5	0%	1%
6	3%	1%
7	1%	3%
8	1%	1%
9	2%	2%
10	0%	**81%**
11	1%	1%
12	15%	2%
13	**65%**	1%
14	10%	2%
15	1%	1%
16	1%	1%
17	0%	0%
	100%	100%

	Number of Workers																	Expected TH	
Number of Stations	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	R^2 Values	Data #1	Data #2
5	97	143	175														0.96	N/A	138
6	101	131	183	208													0.95	N/A	156
7	101	141	173	201	258												0.96	258	175
8	98	138	182	213	251	278											0.97	258	215
9	97	150	190	220	253	291	291										0.90	269	230
10	95	141	173	213	243	256	325	347									0.97	288	241
11	96	141	188	224	269	288	294	352	352								0.91	300	340
12	98	137	172	234	241	272	298	325	402	415							0.96	297	383
13	99	139	183	232	253	281	321	372	444	460	460						0.98	414	423
14	96	141	181	232	256	294	357	390	390	390	390	390					0.80	384	376
15	93	141	180	222	267	294	352	362	384	415	429	437	487				0.93	432	374
16	96	141	181	222	258	298	338	367	384	384	384	384	384	384			0.63	379	370
17	95	140	177	232	256	317	338	390	390	390	390	390	390	390	390		0.52	385	377
18	94	141	185	226	272	291	352	396	408	408	452	452	469	506	633	633	0.94	447	398
Max. of TH	101	150	190	234	272	317	357	396	444	460	460	452	487	506	633	633			
Ideal TH	105	157	210	262	315	367	419	472	524	577	629	681	734	786	839	891			
% Gap (Max-Ideal)/Ideal	4%	4%	10%	11%	14%	14%	15%	16%	15%	20%	27%	34%	34%	36%	25%	29%			

Table 13. Expected Throughput Based on Worker's Availability per Shift

Table 13 is used to calculate the expected throughput based on the expected number of workers each shift. One data set is repeated for all different configurations i.e. M=[5,18]. Throughput is computed using the following formula:

$$TH = \frac{\text{Available Working Time}}{\text{MaxT}}$$

The R^2 value is listed for each configuration. *Max of TH* is the maximum achievable throughput for specified number of workers. *Ideal TH* is the theoretical throughput that can maximally be achieved. *% Gap* represents the percentage difference of “Max of TH” from “Ideal TH”. Column *Expected TH* demonstrates the expected throughput for the expected number of workers per shift. The following formula is used to calculate the expected throughput.

$$E(THj) = \sum_{e=1}^{M} P(\text{Availability}(e)/\text{shift}) * TH(e) / \sum_{e=1}^{M} P(\text{Availability}(e)/\text{shift}) * e \qquad \forall j = 5,6,\ldots 18$$

Two different data sets are used to illustrate the difference. If data set #1 is selected, the manager should design the production cell with 18 stations since maximum TH can be only achieved under these conditions. According to data set #2, 13 stations need to be selected. It can be concluded from this example that the configuration with high linearity may not always offer the highest throughput. Hence, manager will trade off between linearity and throughput in this example.

6 Conclusion

One of the major challenges in U-shaped cell design is to determine the optimal number of stations for a variable number of workers with trial and error methods. It is very iterative and time consuming. Moreover, no research has been done for a dynamic environment so far.

The proposed mixed integer linear model has proved to be very efficient in determining satisfactory cell design for a variable number of workers with high throughput and low idle time. Moreover, 93% of the experiment resulted with high linearity after the first iteration. The mathematical model is capable of solving larger size problems up to 19 stations in a considerable time. Medium and small size problems can be solved in less than 30minutes. Computers used in this research didn't have the highest capacity. Although, model run out of memory for workers greater than 16 in a 19 station configuration, it will not be a problem with higher capacity computers. Solving time is important but it is not the most critical objective since line balancing is not performed daily as a scheduling problem. Unlike the other line balancing algorithms, this model shows expected throughput and linearity for every possible number of workers and stations. Whenever demand or production rate changes, the manager will refer to the table that shows the entire throughput by the number of workers and stations to determine optimal walk-paths for the expected throughput.

Another advantage of the proposed model is that rebalancing can be performed by simply altering the number of workers if planned cycle time is changed. Previous line

balancing techniques require redesign of the entire production cell whenever rebalancing was needed. Hence, they are more time consuming and costly.

Lastly, proposed algorithm includes walking times and crossover issues while determining the walk-path assignments. The majority of the previous research did not consider walking times and crossover problems.

7 Recommendation for Future Research

This thesis is an initial work for u-line balancing in a dynamic environment. However, the model is developed for a single model production with deterministic processing times. In the practical world, a mixed-model production with stochastic task times is more common. This model can be extended for mixed production and stochastic task times.

This research developed an algorithm for flexible work environments. No study was conducted to compare the trade-off between flexibility and throughput. Future research can investigate the profit and losses if our algorithm is used.

Machine waiting time is not considered in this paper. Future research can modify this model including waiting times.

8 References

1. Erel, E., Sabuncuoglu, I. and Aksu, B.A. (2001). Balancing of U-type Assembly Systems Using Simulated Annealing, *International Journal of Production Research,* v 39, n 13, p 3003-3015
2. Gokcen, H. and Agpak, K. (2006). A Goal Programming Approach to Simple U-Line Balancing Problem, *European Journal of Operational Research,* v 171, n 2, p 577-585.
3. Urban, T. (1998). Optimal Balancing of U-Shaped Assembly Lines, *Management Science,* v 44, n 5, p 738-741.
4. Martinez, U. and Duff, W. (2004). Heuristic Approaches to Solve the U-Shaped Line Balancing Problem Augmented by Genetic Algorithms, *IEEE Systems and Information Engineering Design Symposium,* p 287-293.
5. Miltenberg, J. and Wijngaard, J. (1994). The U-Line Line Balancing Problem, *Management Science,* v 40, n 10, p 1378-1388
6. Miltenberg, J. (2001). One-Piece Flow Manufacturing on U-shaped Production Lines: A Tutorial, *IIE Transactions,* v 33, n 4, p 303-321.
7. Miltenberg, J. (2001). U-shaped Production Lines: A Review of Theory and Practice, *International Journal of Production Economics,* v 70, n 3, p 201-204.
8. Monden, Y. (1993). *Toyota Production System (2nd Edition)*, Industrial Engineering and Management Press, Institute of Industrial Engineers, Norcross, GA, p 161.
9. Nakade, K. and Ohno, K. (1997). Analysis and Optimization of a U-Shaped Production Line, *Journal of the Operations Research Society of Japan,* v 40, n 1, p 90-104.

10. Nakade, K. and Ohno, K. (1997). Stochastic Analysis of a U-Shaped Production Line with Multiple Workers, *Computers & Industrial Engineering*, v 33, n 3-4, p 809-812.

11. Nakade, K. and Ohno, K. (1999). An Optimal Worker Allocation Problem for a U-Shaped Production Line, *International Journal of Production Economics,* v 60-61, Apr, p 353-358.

12. Nakade, K. and Ohno, K. (2003). Separate and Carousel Type Allocations of Workers in a U-Shaped Production Line, *European Journal of Operational Research,* v 145, n 2, p 403-424.

13. Scholl, A and Klein, R. (1999). ULINO: Optimally Balancing U-Shaped JIT Assembly Lines, *International Journal of Production Research,* v 37, n 4, p 721-736.

14. Stockton, D.J., Ardon-Finch, J. and Khalil, R. (2005). Walk Cycle Design For Flexible Manpower Lines Using Genetic Algorithms, *International Journal of Computer Integrated Manufacturing,* v 18, n 1, p 15-26

15. Voss, C.A. (1987). *Just In Time Manufacture*, IFS (Publications) Ltd., UK, Springer-Verlang.

9 APPENDIES

Appendix A: ILOG Program Codes- Stage 1

```
int+  T =...;                          // Input variable-maximum number of tasks
int+  M =...;                          // Input variable-number of stations
float+ D[task]=...;                    // Input variable-duration of task (deterministic time)

range task 1..T;                       // Indices for task – i
range station 1..M;                    // Indices for station – j

var int+ X [task, station] in 0..1;    // Decision variable-task to station assignment variable
var float+ Stime[station];             // Decision variable-station time
var float+ maxf;                       // Decision variable-objective function variable

struct precedence {                    // Variable-precedence diagram-P(i,h)
   int h;
   int i;
};

{precedence} P =...;                   // Input variable- precedence relationships
```

```
Minimize                               // Objective function
maxf
```

```
Subject to

{

 // 1)  Line balancing relaxation constraint - constant number (i.e. 20)  is subject to change

      // maxf>=20;            // used when relaxation is needed.
      // maxf<=20;

 // 2) Each task can be assigned to only one station

        forall ( i in task)
        sum ( j in station) X [ i, j]  = 1;

 // 3) Every station has to be assigned

        forall ( j in station)
        sum ( i in task ) X [ i, j] >= 1;
```

```
// 4) Precedence constraints

   forall (<h,i> in P, j in [1..M])
   X [h, j] <= sum (k in [j..M])  X[i,k];

// 5) Minimize the maximum of the station times

   forall (j in [1..M-1], g in [j+1..M])
  (sum ( i in task) D [i]* (X [i,j]))- (sum ( i in task) D [i]* (X [i,g])) <=  maxf ;

   forall (j in [1..M-1], g in [j+1..M])
   (sum ( i in task) D [i]* ( X i,g]))- (sum ( i in task) D[i]* (X [i,j])) <= maxf ;

// 6) Station Time

   forall ( j in station)
   Stime[j]= sum (i in task) D[i] * X [i,j] ;

// 7) These constraints are only used in Percentage Method- constant numbers (ex. 0.09) are subject
to  change & number of these constraints are determined by number of stations (ex. 3 stations)

  // Stime [1]-0.40* (sum (i in task) D[i]) <= maxf ;
  // 0.40*(sum ( i in task) D[i] ) - Stime[1] <= maxf ;

  // Stime [2]-0.30* (sum (i in task) D[i]) <= maxf ;
  // 0.30 * (sum ( i in task) D[i] ) - Stime[1] <= maxf ;

  // Stime [3]-0.30 * (sum (i in task) D[i]) <= maxf ;
  // 0.30 *(sum ( i in task) D[i] ) - Stime[3] <= maxf ;

} ;

  display ( i in task, j in station: X [i,j] > 0) X [i,j] ;
  display ( j in station: Stime [j] > 0) Stime[j] ;
```

Appendix B: ILOG Program Codes- Stage 2

```
int+ M  =...;                                    // Input variable-number of stations
int+ N  =...;                                    // Input variable- number of workers
int+ Lsmax =...;                                 // Input variable-crossover constraint input
int+ Hsmax =...;                                 // Input variable-crossover constraint input
float+ L [station,station] =...;                 // Input variable-walking time between stations
float+ Stime [station]  =...;                    // Input variable- station times (obtained from Stage 1)

range station 1..M;                              // Indices for stations
range worker 1..N;                               // Indices for workers

var int+  Y [station, worker] in 0..1;           // Decision variable- station to worker assignment
var int+ W [station,station,worker] in 0..1;     // Decision variable- walkpath assignment
var int+ FS [station, worker] in 0..1;           // Decision variable-first station
var int+ IM [station,station,worker] in 0..1;    // Decision variable- immediate station
var int+ LS [station,worker] in 0..1;            // Decision variable- last station
var int+ B [station, station, worker] in 0..1;   // Decision variable- back function
var float+ Wtime [worker];                       // Decision variable- walk-path time
var float+ MaxT;                                 // Decision variable- objective function
```

```
Minimize                                         // Objective Function
MaxT
```

```
Subject to

{

// 1) Walk-path Time Upper bound- obtained from previous run for N=N-1

   Maxt <= 57;                (constant is subject to change)
   Maxt >= 73;

// 2) Each station has to be assigned and can be assigned to only one worker

   forall (j in station)
   sum (e in worker) Y [j,e] = 1 ;

 // 3) Walk-path assignments

  // 3.a) Connect route for the stations assigned to same worker

   forall (j in 1..M-1, g in [j+1..M], e in worker )
   1+ W[j,g,e] >= Y [g,e]+Y [j,e];

   forall (j in 1..M-1, g in [j+1..M], e in worker )
   W [j,g,e] <= Y [g,e];
```

```
forall (j in 1..M-1, g in [j+1..M],e in worker)
W [j,g,e] <= Y [j,e];

forall (j in 1..M, g in 1..j, e in worker)
W [j,g,e] = 0;
```

// 4) Prevent crossovers –diagonal and horizontal crossovers

```
forall (e in worker, f in worker:f<>e, j in [1..M-1], g in [j+1..M], h in [j+1..g-1], r in [g+1..M])
3-Y [j,e] – Y [g,e]>=Y [h,f]+Y [r,f];

forall (e in worker, j in [1..LSmax-2], g in [j+2..LSmax])
(g-j-1)*(Y[ j,e]+Y [g,e]-1)<=sum (k in [j+1..g-1]) Y [k,e];

forall (e in worker, j in [HSmax..M-2], g in [j+2..M])
(g-j-1)*(Y [j,e]+Y [g,e]-1)<=sum (k in [j+1..g-1]) Y [k,e];
```

// 5) Calculate the sequence of stations

// 5a) Determine immediate next station

```
forall ( e in worker, j in [1..M-1], g in [j+1..M])
IM [j,g,e]>=W [j,g,e]-sum(h in [j+1..g-1]) W [j,h,e];

forall (e in worker, j in [1..M-1], g in [j+1..M])
W [j,g,e]>=im[j,g,e];

forall (e in worker, j in [1..M-1], g in [j+1..M])
sum (h in [j+1..g-1]) W [j,h,e]<=M * (1-IM [j,g,e]);
```

// 5b) Determine last station

```
forall (e in worker, j in station)
sum (g in station) g * LS [g,e] >= j * Y [j,e];

forall ( e in worker)
sum (j in station) LS [j,e] = 1;

forall (e in worker, j in station)
LS [j,e] <= Y [j,e];
```

// 5c) Determine First station

```
forall (e in worker, j in station)
sum (g in station) (M-g) * FS[g,e] >= (M-j) * Y [j,e];

forall ( e in worker)
sum (j in station) FS [j,e] = 1;
```

```
        forall (e in worker, j in station)
        FS [j,e] <= Y [j,e];

       5d) Determine Back Function

        forall (e in worker, j in [2..M], g in [1..j-1])
        B [j,g,e]+1 >= LS [j,e]+FS [g,e];

        forall (e in worker, j in [2..M],g in [1..j-1])
        B [j,g,e] <= LS[j,e];

        forall (e in worker, j in [2..M], g in [1..j-1])
        B [j,g,e] <= FS[g,e];

    // 6) Minimize the maximum operator time

        forall ( e in worker)
        sum (j in station) Stime [j] * Y[j,e] +  sum (j in [1..M-1], g in [j+1..M]) L [j,g] * IM[j,g,e]+
        sum(j in [2..M], g in [1..j-1]) B [j,g,e] * L [g,j] <= MaxT;

    // 7) Printing constraint

        forall ( e in worker)
        Wtime[e]=sum ( j in station) Y[j,e] * Stime[j] + sum (j in [1..M-1], g in [j+1..M]) L [j,g] * IM [j,g,e]+
        sum (j in [2..M], g in [1..j-1]) B [j,g,e] * L [g,j] ;

    // 8) Cutting Plane Constraints- used only when cutting plane method is employed

        // Y [1,1] = 1;        // these constraints are subject to change by problem type & settings
        // Y [4,2] = 1;

};

    display (j in station, e in worker: LS [j,e]>0) LS [j,e];
    display (j in station, e in worker: FS [j,e]>0) FS [j,e];
    display (j in station, g in station, e in worker: W [j,g,e] > 0) W [j,g,e];
    display (j in station, e in worker :Y [j,e]>0) Y [j,e];
    display ( e in worker :Wtime[e]>0) Wtime[e];
```

Appendix C: Stage 1 and Stage 2 Data

All "M" is subject to change for different configurations.

MITCHELL, N=21
D =[12,4,9,5,9,4,8,7,5,10,3,6,5,3,5,3,13,5,4,3,7];

P = {<1,2>,<1,3>,<2,21>,<3,4>,<4,5>,<4,21>, <5,6>,<5,7>,<6,8>,<7,8>,<7,14>,<8,9>,<9,10>,<9,11>,
<9,12>,<9,13>,<10,15>,<11,15>,<12,15>,<13,17>,<13,18>,<14,19>,<15,16>,<15,18>,<16,17>,<17,20>,<18,19>};

T =21;
M = 8;

ROSZIEG, N=25
D =[4,3,9,5,9,4,8,7,5,1,3,1,5,3,5,3,13,5,2,3,7,5,3,8,4];

P ={<1,3>,<2,3>,<3,4>,<4,5>,<4,8>,<5,6>,<6,7>,<6,10>,<7,11>,<7,12>,<8,9>,<8,11>,<9,13>,<9,10>,
<11,13>,<12,15>,<13,14>,<14,16>,<14,19>,<14,20>,<15,17>,<15,22>,<16,18>,<17,18>,<17,23>,<18,25>,
<19,22>,<20,21>,<20,25>,<21,22>,<21,24>,<23,25>};

T =25;
M = 8;

SAWYER30, N=30
D =[8,7,19,10,2,6,14,10,1,4,14,15,5,12,9,10,2,10,18,16,21,14,16,7,17,9,25,7,14,2];

P ={<1,4>,<1,5>,<2,11>,<2,12>,<3,16>,<3,17>,<4,7>,<5,6>,<6,7>,<7,8>,<8,9>,<9,26>,<10,24>,<12,13>,
<13,14>,<14,15>,<14,20>,<15,22>,<16,20>,<17,18>,<18,19>,<20,21>,<20,24>,<21,22>,<22,23>,<23,27>,
<24,25>,<25,26>,<26,27>,<27,28>,<27,29>,<29,30>};

T =30;
M = 8;

GUNTHER, N=35
D =[29,3,5,22,6,14,2,5,22,30,23,30,23,2,19,29,2,2,19,29,6,10,16,23,5,5,5,40,2,5,5,1,40,2,2];

P ={<1,2>,<1,5>,<1,7>,<1,10>,<1,12>,<2,3>,<3,4>,<4,11>,<5,6>,<6,7>,<6,8>,<7,14>,<7,18>,<8,9>,<9,13>,
<10,14>,<11,28>,<11,33>,<12,18>,<13,28>,<13,33>,<14,15>,<15,16>,<16,21>,<17,20>,<18,19>,<19,20>,
<20,21>,<21,22>,<21,25>,<21,30>,<21,32>,<22,23>,<23,24>,<24,27>,<25,26>,<26,27>,<27,28>,<27,33>,
<27,34>,<28,29>,<30,31>,<31,32>,<32,33>,<33,35>};

T = 35;
M = 17;

KILBRID, N=45
D = [9,9,10,10,17,17,13,13,20,20,10,11,6,22,11,19,12,3,7,4,55,14,27,29,26,6,5,24,4,5,7,4,15,3,7,9,4,7,5,4,21,12,6,5,5];

P={<1,3>,<1,7>,<2,4>,<2,8>,<3,5>,<4,6>,<5,9>,<6,10>,<7,9>,<7,14>,<8,10>,<8,14>,<9,41>,<10,41>,<11,13>,
<12,13>,<12,37>,<13,14>,<13,15>,<14,17>,<14,25>,<14,29>,<14,30>,<14,31>,<14,32>,<15,16>,<15,18>,<15,23>,
<15,24>,<16,19>,<17,26>,<17,27>,<18,19>,<19,20>,<19,33>,<20,21>,<21,22>,<22,28>,<23,33>,<24,33>,<25,26>,
<26,38>,<27,28>,<27,33>,<28,38>,<29,41>,<30,41>,<31,41>,<32,41>,<33,34>,<33,35>,<33,36>,<34,38>,<35,40>,

<36,38>,<37,43>,<38,40>,<39,41>,<40,41>,<41,42>,<42,44>,<42,45>};

T = 45;
M = 8;

HAHN, N=53
D =[100,971,142,142,142,103,96,99,1207,160,180,82,60,112,420,1556,236,259,125,601,80,80,70,89,89,105,
330,132,69,99,70,70,158,191,70,53,50,125,353,70,128,65,1775,91,113,487,138,80,80,65,40,742,1085];

P={<1,2>,<1,3>,<1,4>,<1,5>,<1,6>,<1,7>,<2,36>,<3,36>,<4,9>,<5,9>,<6,9>,<7,9>,<8,9>,<9,10>,<10,11>,
<11,12>,<12,13>,<12,14>,<12,15>,<13,16>,<13,17>,<13,18>,<14,22>,<15,16>,<15,17>,<16,29>,<16,30>,
<16,31>,<17,19>,<18,42>,<19,20>,<19,21>,<19,22>,<20,23>,<21,24>,<22,25>,<23,36>,<24,36>,<25,26>,
<26,27>,<27,28>,<28,29>,<28,30>,<28,31>,<28,32>,<28,33>,<28,34>,<29,35>,<30,35>,<31,35>,<32,35>,
<33,35>,<34,35>,<35,36>,<36,37>,<37,38>,<37,29>,<37,40>,<38,41>,<39,41>,<40,41>,<41,42>,<42,43>,
<42,44>,<42,45>,<42,46>,<42,47>,<43,48>,<44,49>,<45,51>,<45,52>,<46,50>,<47,51>,<47,52>,<48,51>,
<48,52>,<49,51>,<49,52>,<50,51>,<50,52>,<51,53>,<52,53>};

T =53;
M = 8;

WARNECKE, N=58
D =[10,53,41,36,35,17,34,23,14,52,33,34,12,52,12,33,44,7,15,13,29,37,43,23,24,9,16,12,26,22,51,47,34,23,
12,52,12,33,44,7,15,13,29,37,43,23,24,9,16,12,26,12,52,12,33,44,7,15];

P={<1,9>,<2,25>,<3,21>,<3,22>,<4,10>,<5,36>,<6,23>,<6,24>,<7,43>,<8,44>,<9,11>,<11,13>,<11,14>,
<11,15>,<12,17>,<13,17>,<14,17>,<15,17>,<16,18>,<17,20>,<18,19>,<19,23>,<20,21>,<20,25>,<21,26>,
<21,44>,<22,27>,<23,31>,<24,31>,<25,38>,<26,29>,<27,30>,<28,32>,<28,33>,<29,31>,<30,34>,<31,36>,
<32,45>,<33,35>,<34,36>,<35,43>,<36,37>,<36,38>,<37,39>,<37,40>,<37,41>,<38,52>,<39,43>,<40,43>,
<41,42>,<42,43>,<43,45>,<44,45>,<45,46>,<45,48>,<45,49>,<46,47>,<46,50>,<47,51>,<48,55>,<49,58>,
<50,58>,<51,52>,<52,53>,<53,54>,<54,55>,<54,56>,<55,57>,<56,57>,<57,58>};

T =58;
M =8;

TONGE, N=70
D=[17,66,54,52,6,88,21,128,68,70,85,21,134,135,94,90,50,143,19,54,50,40,73,12,152,42,45,74,26,11,
31,50,102,46,35,40,2,1,3,13,16,25,21,43,30,83,89,56,59,43,11,26,44,121,38,68,22,7,16,32,25,27,156,28,15,26
,18,72,23,27];

P={<1,2>,<1,41>,<1,69>,<1,70>,<2,3>,<3,4>,<3,68>,<4,6>,<4,7>,<5,6>,<5,24>,<5,30>,<6,8>,<7,8>,<8,12>,
<9,10>,<10,11>,<11,12>,<12,13>,<12,14>,<13,23>,<14,23>,<15,16>,<16,17>,<16,18>,<17,19>,<18,19>,
<19,20>,<19,22>,<19,57>,<20,21>,<21,23>,<22,23>,<23,25>,<23,31>,<23,33>,<24,25>,<25,26>,<25,26>,
<25,28>,<25,29>,<26,35>,<27,35>,<28,35>,<29,35>,<30,31>,<31,32>,<32,35>,<33,34>,<34,35>,<35,36>,
<35,44>,<35,48>,<35,51>,<35,53>,<35,56>,<35,60>,<35,61>,<35,62>,<36,37>,<37,38>,<38,39>,<39,40>,
<40,42>,<41,42>,<42,43>,<43,50>,<44,45>,<45,46>,<46,47>,<47,50>,<48,49>,<49,50>,<51,52>,<52,54>,
<53,54>,<54,55>,<57,58>,<58,59>,<59,60>,<61,65>,<62,63>,<63,64>,<64,65>,<64,66>,<64,67>};

T = 70;
M =17;

WEE-MAG, N=75
D =[23,24,25,26,23,22,6,22,23,21,22,15,5,23,4,26,21,5,24,25,26,26,24,27,20,23,25,13,3,11,21,22,21,22,25,8,22,

24,22,21,6,26,22,6,21,25,11,22,21,25,22,22,23,22,22,25,23,21,22,22,22,22,21,27,23,2,26,25,24,22,24,4,22,10,25];

P={<1,2>,<1,3>,<1,4>,<1,5>,<1,6>,<1,7>,<2,15>,<3,13>,<3,24>,<4,8>,<4,14>,<4,16>,<5,12>,<5,15>,<6,9>,
<6,10>,<6,11>,<6,13>,<9,20>,<9,24>,<10,18>,<12,19>,<13,22>,<15,17>,<15,20>,<15,23>,<16,21>,<16,26>,
<17,30>,<18,26>,<18,30>,<20,27>,<21,33>,<24,25>,<25,28>,<25,30>,<25,33>,<25,34>,<26,31>,<26,32>,
<26,41>,<27,29>,<27,35>,<27,36>,<31,37>,<31,39>,<32,44>,<32,45>,<33,41>,<35,38>,<35,42>,<36,40>,
<36,43>,<39,51>,<40,46>,<42,47>,<43,48>,<43,50>,<46,51>,<46,48>,<47,49>,<47,50>,<47,52>,<47,53>,
<49,59>,<49,61>,<49,62>,<50,54>,<50,55>,<50,60>,<50,62>,<52,56>,<52,57>,<53,58>,<55,63>,<58,65>,
<59,64>,<59,66>,<62,67>,<66,68>,<68,69>,<68,70>,<68,71>,<68,72>,<68,73>,<68,74>,<68,75>};

T =75;
M =8;

DELPHI DATA-CELL 842, N=79
D =[5,4,5,4,4,4,4,2,2,2,2,2,5,18,0,3,5,3,3,3,3,3,3,3,3,3,3,3,5,4,3,4,3,36,15,0,5,8,4,6,3,3,7,5,3,4,3,3,3,
3,3,3,3,5,3,3,2,0,3,3,3,3,3,6,2,3,3,3,3,4,3,3,5,3,3,7,0,2,9,19];

P ={<1,2>,<1,3>,<1,4>,<1,5>,<1,6>,<1,7>,<1,8>,<1,9>,<10,11>,<10,12>,<12,13>,<11,13>,<13,14>,<14,15>,
<1,16>,<4,16>,<9,16>,<5,16>,<6,16>,<7,16>,<8,16>,<15,16>,<16,17>,<17,18>,<17,19>,<17,20>,<17,21>,
<17,22>,<17,23>,<17,24>,<17,25>,<17,26>,<17,27>,<27,28>,<18,28>,<19,28>,<20,28>,<21,28>,<22,28>,
<23,28>,<24,28>,<25,28>,<26,28>,<28,29>,<29,30>,<30,31>,<31,32>,<32,33>,<33,34>,<34,35>,<35,36>,
<36,37>,<37,38>,<39,40>,<39,41>,<39,42>,<42,43>,<40,44>,<41,44>,<42,44>,<37,45>,<44,45>,<43,45>,
<45,46>,<46,47>,<46,48>,<46,49>,<46,50>,<46,51>,<51,52>,<47,52>,<48,52>,<49,52>,<50,52>,<52,53>,
<53,54>,<53,55>,<55,56>,<54,56>,<56,57>,<57,58>,<57,59>,<57,60>,<57,61>,<57,62>,<58,63>,<59,63>,
<60,63>,<61,63>,<62,63>,<58,64>,<59,64>,<60,64>,<61,64>,<62,64>,<64,65>,<64,66>,<64,67>,<64,68>,
<38,69>,<58,69>,<59,69>,<60,69>,<61,69>,<62,69>,<69,70>,<69,71>,<58,72>,<59,72>,<60,72>,<61,72>,
<62,72>,<72,73>,<72,74>,<74,75>,<73,75>,<70,75>,<71,75>,<65,75>,<66,75>,<67,75>,<68,75>,<63,75>,
<75,76>,<76,77>,<77,78>,<78,79>};

T =79;
M =19;

Walking Times:

5 STATIONS

```
L= [        [0,1,2,2,1]
            [0,0,1,1,1]
            [0,0,0,1,2]
            [0,0,0,0,1]
            [0,0,0,0,0]
        ];
```

6 STATIONS

```
L=  [   [0,1,2,3,2,1]
        [0,0,1,2,1,2]
        [0,0,0,1,2,3]
        [0,0,0,0,1,2]
        [0,0,0,0,0,1]
        [0,0,0,0,0,0]
    ];
```

7 STATIONS

```
L=    [    [0,1,2,3,3,2,1]
           [0,0,1,2,2,1,1]
           [0,0,0,1,1,1,2]
           [0,0,0,0,1,2,3]
           [0,0,0,0,0,1,2]
           [0,0,0,0,0,0,1]
           [0,0,0,0,0,0,0]
  ];
```

8 STATIONS

```
L =    [   [0,1,2,3,4,3,2,1]
           [0,0,1,2,3,2,1,2]
           [0,0,0,1,2,1,2,3]
           [0,0,0,0,1,2,3,4]
           [0,0,0,0,0,1,2,3]
           [0,0,0,0,0,0,1,2]
           [0,0,0,0,0,0,0,1]
           [0,0,0,0,0,0,0,0]
         ];
```

9 STATIONS

```
L =    [   [0,1,2,3,4,4,3,2,1]
            [0,0,1,2,3,3,2,1,2]
           [0,0,0,1,2,2,1,2,3]
           [0,0,0,0,1,1,2,3,4]
           [0,0,0,0,0,1,2,3,4]
           [0,0,0,0,0,0,1,2,3]
           [0,0,0,0,0,0,0,1,2]
           [0,0,0,0,0,0,0,0,1]
           [0,0,0,0,0,0,0,0,0]
           ];
```

10 STATION

```
L    = [    [0,1,2,3,4,5,4,3,2,1]
            [0,0,1,2,3,4,3,2,1,2]
            [0,0,0,1,2,3,2,1,2,3]
            [0,0,0,0,1,2,1,2,3,4]
           [0,0,0,0,0,1,2,3,4,5]
            [0,0,0,0,0,0,1,2,3,4]
           [0,0,0,0,0,0,0,1,2,3]
            [0,0,0,0,0,0,0,0,1,2]
            [0,0,0,0,0,0,0,0,0,1]
```

```
        [0,0,0,0,0,0,0,0,0,0]
    ];
```

11 STATION

```
L    = [    [0,1,2,3,4,5,5,4,3,2,1]
            [0,0,1,2,3,4,4,3,2,1,2]
            [0,0,0,1,2,3,3,2,1,2,3]
            [0,0,0,0,1,2,2,1,2,3,4]
            [0,0,0,0,0,1,1,2,3,4,5]
            [0,0,0,0,0,0,1,2,3,4,5]
            [0,0,0,0,0,0,0,1,2,3,4]
            [0,0,0,0,0,0,0,0,1,2,3]
            [0,0,0,0,0,0,0,0,0,1,2]
           [0,0,0,0,0,0,0,0,0,0,1]
            [0,0,0,0,0,0,0,0,0,0,0]
];
```

12 STATION

```
L  = [     [0,1,2,3,4,5,6,5,4,3,2,1]
           [0,0,1,2,3,4,5,4,3,2,1,2]
           [0,0,0,1,2,3,4,3,2,1,2,3]
           [0,0,0,0,1,2,3,2,1,2,3,4]
           [0,0,0,0,0,1,2,1,2,3,4,5]
           [0,0,0,0,0,0,1,2,3,4,5,6]
           [0,0,0,0,0,0,0,1,2,3,4,5]
           [0,0,0,0,0,0,0,0,1,2,3,4]
           [0,0,0,0,0,0,0,0,0,1,2,3]
           [0,0,0,0,0,0,0,0,0,0,1,2]
          [0,0,0,0,0,0,0,0,0,0,0,1]
          [0,0,0,0,0,0,0,0,0,0,0,0]
];
```

13 STATION

```
L    = [   [0,1,2,3,4,5,6,6,5,4,3,2,1]
           [0,0,1,2,3,4,5,5,4,3,2,1,2]
           [0,0,0,1,2,3,4,4,3,2,1,2,3]
           [0,0,0,0,1,2,3,3,2,1,2,3,4]
           [0,0,0,0,0,1,2,2,1,2,3,4,5]
           [0,0,0,0,0,0,1,1,2,3,4,5,6]
           [0,0,0,0,0,0,0,1,2,3,4,5,6]
           [0,0,0,0,0,0,0,0,1,2,3,4,5]
           [0,0,0,0,0,0,0,0,0,1,2,3,4]
           [0,0,0,0,0,0,0,0,0,0,1,2,3]
           [0,0,0,0,0,0,0,0,0,0,0,1,2]
           [0,0,0,0,0,0,0,0,0,0,0,0,1]
           [0,0,0,0,0,0,0,0,0,0,0,0,1]
```

```
];
```

14 STATION

```
L   = [   [0,1,2,3,4,5,6,7,6,5,4,3,2,1]
          [0,0,1,2,3,4,5,6,5,4,3,2,1,2]
          [0,0,0,1,2,3,4,5,4,3,2,1,2,3]
          [0,0,0,0,1,2,3,4,3,2,1,2,3,4]
          [0,0,0,0,0,1,2,3,2,1,2,3,4,5]
          [0,0,0,0,0,0,1,2,1,2,3,4,5,6]
          [0,0,0,0,0,0,0,1,2,3,4,5,6,7]
          [0,0,0,0,0,0,0,0,1,2,3,4,5,6]
          [0,0,0,0,0,0,0,0,0,1,2,3,4,5]
          [0,0,0,0,0,0,0,0,0,0,1,2,3,4]
          [0,0,0,0,0,0,0,0,0,0,0,1,2,3]
          [0,0,0,0,0,0,0,0,0,0,0,0,1,2]
          [0,0,0,0,0,0,0,0,0,0,0,0,0,1]
          [0,0,0,0,0,0,0,0,0,0,0,0,0,0]
];
```

15 STATION

```
L   = [    [0,1,2,3,4,5,6,7,7,6,5,4,3,2,1]
           [0,0,1,2,3,4,5,6,6,5,4,3,2,1,2]
           [0,0,0,1,2,3,4,5,5,4,3,2,1,2,3]
           [0,0,0,0,1,2,3,4,4,3,2,1,2,3,4]
           [0,0,0,0,0,1,2,3,3,2,1,2,3,4,5]
           [0,0,0,0,0,0,1,2,2,1,2,3,4,5,6]
           [0,0,0,0,0,0,0,1,1,2,3,4,5,6,7]
           [0,0,0,0,0,0,0,0,1,2,3,4,5,6,7]
           [0,0,0,0,0,0,0,0,0,1,2,3,4,5,6]
           [0,0,0,0,0,0,0,0,0,0,1,2,3,4,5]
           [0,0,0,0,0,0,0,0,0,0,0,1,2,3,4]
           [0,0,0,0,0,0,0,0,0,0,0,0,1,2,3]
           [0,0,0,0,0,0,0,0,0,0,0,0,0,1,2]
           [0,0,0,0,0,0,0,0,0,0,0,0,0,0,1]
           [0,0,0,0,0,0,0,0,0,0,0,0,0,0,1]
];
```

16 STATION

```
L   = [   [0,1,2,3,4,5,6,7,8,7,6,5,4,3,2,1]
          [0,0,1,2,3,4,5,6,7,6,5,4,3,2,1,2]
          [0,0,0,1,2,3,4,5,6,5,4,3,2,1,2,3]
          [0,0,0,0,1,2,3,4,5,4,3,2,1,2,3,4]
          [0,0,0,0,0,1,2,3,4,3,2,1,2,3,4,5]
          [0,0,0,0,0,0,1,2,3,2,1,2,3,4,5,6]
          [0,0,0,0,0,0,0,1,2,1,2,3,4,5,6,7]
          [0,0,0,0,0,0,0,0,1,2,3,4,5,6,7,8]
          [0,0,0,0,0,0,0,0,0,1,2,3,4,5,6,7]
```

```
            [0,0,0,0,0,0,0,0,0,0,1,2,3,4,5,6]
            [0,0,0,0,0,0,0,0,0,0,0,1,2,3,4,5]
            [0,0,0,0,0,0,0,0,0,0,0,0,1,2,3,4]
            [0,0,0,0,0,0,0,0,0,0,0,0,0,1,2,3]
            [0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,2]
            [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1]
            [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]
];
```

17 STATION

```
L   = [     [0,1,2,3,4,5,6,7,8,8,7,6,5,4,3,2,1]
            [0,0,1,2,3,4,5,6,7,7,6,5,4,3,2,1,2]
            [0,0,0,1,2,3,4,5,6,6,5,4,3,2,1,2,3]
            [0,0,0,0,1,2,3,4,5,5,4,3,2,1,2,3,4]
            [0,0,0,0,0,1,2,3,4,4,3,2,1,2,3,4,5]
            [0,0,0,0,0,0,1,2,3,3,2,1,2,3,4,5,6]
            [0,0,0,0,0,0,0,1,2,2,1,2,3,4,5,6,7]
            [0,0,0,0,0,0,0,0,1,1,2,3,4,5,6,7,8]
            [0,0,0,0,0,0,0,0,0,1,2,3,4,5,6,7,8]
            [0,0,0,0,0,0,0,0,0,0,1,2,3,4,5,6,7]
            [0,0,0,0,0,0,0,0,0,0,0,1,2,3,4,5,6]
            [0,0,0,0,0,0,0,0,0,0,0,0,1,2,3,4,5]
            [0,0,0,0,0,0,0,0,0,0,0,0,0,1,2,3,4]
            [0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,2,3]
            [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,2]
            [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1]
            [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]
];
```

18 STATION

```
L   = [     [0,1,2,3,4,5,6,7,8,9,8,7,6,5,4,3,2,1]
            [0,0,1,2,3,4,5,6,7,8,7,6,5,4,3,2,1,2]
            [0,0,0,1,2,3,4,5,6,7,6,5,4,3,2,1,2,3]
            [0,0,0,0,1,2,3,4,5,6,5,4,3,2,1,2,3,4]
            [0,0,0,0,0,1,2,3,4,5,4,3,2,1,2,3,4,5]
            [0,0,0,0,0,0,1,2,3,4,3,2,1,2,3,4,5,6]
            [0,0,0,0,0,0,0,1,2,3,2,1,2,3,4,5,6,7]
            [0,0,0,0,0,0,0,0,1,2,1,2,3,4,5,6,7,8]
            [0,0,0,0,0,0,0,0,0,1,2,3,4,5,6,7,8,9]
            [0,0,0,0,0,0,0,0,0,0,1,2,3,4,5,6,7,8]
            [0,0,0,0,0,0,0,0,0,0,0,1,2,3,4,5,6,7]
            [0,0,0,0,0,0,0,0,0,0,0,0,1,2,3,4,5,6]
            [0,0,0,0,0,0,0,0,0,0,0,0,0,1,2,3,4,5]
            [0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,2,3,4]
            [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,2,3]
            [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,2]
            [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1]
            [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]
        ];
```

19 STATION

```
L   = [   [0,1,2,3,4,5,6,7,8,9,9,8,7,6,5,4,3,2,1]
          [0,0,1,2,3,4,5,6,7,8,8,7,6,5,4,3,2,1,2]
          [0,0,0,1,2,3,4,5,6,7,7,6,5,4,3,2,1,2,3]
          [0,0,0,0,1,2,3,4,5,6,6,5,4,3,2,1,2,3,4]
          [0,0,0,0,0,1,2,3,4,5,5,4,3,2,1,2,3,4,5]
          [0,0,0,0,0,0,1,2,3,4,4,3,2,1,2,3,4,5,6]
          [0,0,0,0,0,0,0,1,2,3,3,2,1,2,3,4,5,6,7]
          [0,0,0,0,0,0,0,0,1,2,2,1,2,3,4,5,6,7,8]
          [0,0,0,0,0,0,0,0,0,1,1,2,3,4,5,6,7,8,9]
          [0,0,0,0,0,0,0,0,0,0,1,2,3,4,5,6,7,8,9]
          [0,0,0,0,0,0,0,0,0,0,0,1,2,3,4,5,6,7,8]
          [0,0,0,0,0,0,0,0,0,0,0,0,1,2,3,4,5,6,7]
          [0,0,0,0,0,0,0,0,0,0,0,0,0,1,2,3,4,5,6]
          [0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,2,3,4,5]
          [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,2,3,4]
          [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,2,3]
          [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,2]
          [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1]
          [0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0]
    ];
```

Appendix D: Summary of Data Results

Configuration: 5 Stations

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output/ Shift	R^2	Part/ Person
Mitchell	21	original	5	0.02	1.3	1.3	1.3	2	0.02	71	71	357	0.98	178.5
								3	0.09	50	50	506		168.7
								4	0.14	39	39	649		162.3
Roszieg	25	original	5	0.05	0.5	0.5	0.5	2	0.03	68	68	372	0.96	186
								3	0.08	47	47	539		179.7
								4	0.11	38	38	666		166.5
Sawyer	30	1*2	5	0.88	0.36	0.36	0.36	2	0.06	175	175	145	0.97	72.5
								3	0.08	121	121	209		69.7
								4	0.08	97	97	261		65.3
Gunther	35	1*2	5	1.7	0.13	0.13	0.13	2	0.05	260	260	97	0.96	48.5
								3	0.08	177	177	143		47.7
								4	0.09	145	145	175		43.8
Kilbrid	45	1*2	5	0.61	0.52	0.52	0.52	2	0.05	296	296	86	0.94	43
								3	0.08	200	200	127		42.3
								4	0.09	166	166	153		38.3
Hahn	53	1*4	5	25.31	4.2	4.2	4.2	2	0.03	7449	7449	3	0.96	1.5
								3	0.08	4928	4928	5		1.7
								4	0.09	4209	4209	6		1.5
Warnecke	58	1*4	5	71.84	0.48	0.48	0.48	2	0.05	828	828	31	0.94	15.5
								3	0.08	558	558	45		15
								4	0.09	464	464	55		13.8
Tonge	70	1*4	5	957.36	1	0.6	n/a	2	0.06	1868	1868	14	0.91	7
								3	0.08	1245	1245	20		6.7
								4	0.06	1052	1052	24		6
Wee-Mag	75	1*4	5	1.31	0.3	0.3	0.3	2	0.05	802	802	32	0.92	16
								3	0.08	540	540	47		15.7
								4	0.08	450	450	56		14
Cell 842	79	1*2	5	2.34	0.48	0.48	0.48	2	0.06	187	187	135	0.96	67.5
								3	0.08	129	129	196		65.3
								4	0.08	104	104	243		60.8

Configuration: 6 Stations

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output / Shift	R^2	Part/ Person
Mitchell	21	original	6	0.8	1.5	1.5	1.5	2	0.13	71	71	357	0.99	178.5
								3	0.17	53	53	478		159.3
								4	0.25	39	39	649		162.3
								5	0.11	30	30	844		168.8
Roszieg	25	original	6	2.22	2.25	2.25	2.25	2	0.11	69	69	367	0.96	183.5
								3	0.17	50	50	506		168.7
								4	0.3	40	40	633		158.3
								5	0.22	28	28	904		180.8
Sawyer	30	1*2	6	0.73	1.12	1.12	1.12	2	0.14	171	171	148	0.99	74
								3	0.22	128	128	198		66
								4	0.27	93	93	272		68
								5	0.3	75	75	338		67.6
Gunther	35	1*2	6	0.88	12.8	12.8	12.8	2	0.09	251	251	101	0.95	50.5
								3	0.16	194	194	131		43.7
								4	0.33	138	138	183		45.8
								5	0.25	122	122	208		41.6
Kilbrid	45	1*2	6	2.36	1.08	1.08	1.08	2	0.11	287	287	88	0.99	44
								3	0.19	212	212	119		39.7
								4	0.19	157	157	161		40.3
								5	0.09	126	126	201		40.2
Hahn	53	1*4	6	61.14	53.25	53.25	53.25	2	0.11	7106	7106	4	0.94	2
								3	0.16	5271	5271	5		1.7
								4	0.19	3887	3887	7		1.8
								5	0.31	3211	3211	8		1.6
Warnecke	58	1*4	6	35.11	3.04	3.04	3.04	2	0.05	799	799	32	0.99	16
								3	0.28	591	591	43		14.3
								4	0.22	438	438	58		14.5
								5	0.19	353	353	72		14.4
Tonge	70	1*4	6	338.44	6.3	6.3	6.3	2	0.06	1788	1788	14	0.99	7
								3	0.2	1330	1330	19		6.3
								4	0.24	979	979	26		6.5
								5	0.24	802	802	32		6.4
WeeMag	75	1*4	6	7.27	2.96	2.96	2.96	2	0.11	772	772	33	0.99	16.5
								3	0.19	572	572	44		14.7
								4	0.23	422	422	60		15
								5	0.39	343	343	74		14.8
Cell 842	79	1*2	6	3.34	0.9	0.9	0.9	2	0.09	183	183	138	0.99	69
								3	0.19	136	136	186		62
								4	0.19	100	100	253		63.3
								5	0.11	79	79	321		64.2

Configuration: 7 Stations

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output/ Shift	R^2	Part/ Person
Mitchell	21	original	7	0.23	0.6	0.6	0.6	2	0.2	70	70	362	0.96	181
								3	0.3	49	49	517		172.3
								4	0.89	40	40	633		158.3
								5	1.64	35	35	723		144.6
								6	0.09	26	26	974		162.3
Roszieg	25	original	7	22.72	3	3	3	2	0.05	66	66	384	0.92	192
								3	0.3	47	47	539		179.7
								4	0.61	37	37	684		171
								5	0.86	32	32	791		158.2
								6	0.11	28	28	904		150.7
Sawyer	30	original	7	1.09	0.92	0.92	0.92	2	0.23	168	168	151	0.96	75.5
								3	0.34	120	120	211		70.3
								4	0.61	97	97	261		65.3
								5	1.25	84	84	301		60.2
								6	0.08	65	65	390		65
Gunther	35	1*2	7	94.88	1.89	1.89	1.89	2	0.23	251	251	101	0.96	50.5
								3	0.38	179	179	141		47
								4	0.95	146	146	173		43.3
								5	1.23	126	126	201		40.2
								6	0.11	98	98	258		43
Kilbrid	45	1*2	7	265.58	2.16	1.16	n/a	2	0.22	288	288	88	0.96	44
								3	0.38	204	204	124		41.3
								4	1.13	165	165	153		38.3
								5	1.34	145	145	175		35
								6	0.11	111	111	228		38
Hahn	53	1*2	7	125.41	36	36	36	2	0.73	7092	7092	4	0.90	2
								3	0.3	4962	4962	5		1.7
								4	0.89	4112	4112	6		1.5
								5	1.22	3534	3534	7		1.4
								6	0.08	2843	2843	9		1.5
Warnecke	58	1*4	7	17.17	2.84	2.84	2.84	2	0.22	800	800	32	0.96	16
								3	0.3	569	569	44		14.7
								4	0.74	460	460	55		13.8
								5	1.06	400	400	63		12.6
								6	0.13	312	312	81		13.5
Tonge	70	1*4	7	583.31	9.3	9.3	9.3	2	0.16	1794	1794	14	0.95	7
								3	0.31	1267	1267	20		6.7
								4	0.75	1033	1033	25		6.3
								5	1.14	899	899	28		5.6
								6	0.11	707	707	36		6
WeeMag	75	1*4	7	4.86	2.2	2.2	2.2	2	0.22	774	774	33	0.96	16.5
								3	0.33	552	552	46		15.3
								4	0.81	447	447	57		14.3
								5	1.11	387	387	65		13
								6	0.09	302	302	84		14
Cell842	79	1*2	7	55.7	5.26	5.26	5.26	2	0.22	189	189	134	0.96	67
								3	0.31	132	132	192		64
								4	0.44	98	98	258		64.5
								5	1.63	91	91	278		55.6
								6	0.44	71	71	357		59.5

Configuration: 8 Stations

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output/ Shift	R^2	Part/ Person
Mitchell	21	original	8	6.08	1.3	1.3	1.3	2	0.25	74	74	342	0.95	171
								3	0.64	50	50	506		168.7
								4	2.52	39	39	649		162.3
								5	3.52	35	35	723		144.6
								6	6.51	27	27	938		156.3
								7	11.72	26	26	974		139.1
Roszieg	25	original	8	2.44	0.75	0.75	0.75	2	0.27	69	69	367	0.96	183.5
								3	0.81	50	50	506		168.7
								4	2.02	36	36	703		175.8
								5	5	33	33	767		153.4
								6	5.25	28	28	904		150.7
								7	10.2	24	24	1055		150.7
Sawyer	30	original	8	3.19	1.32	1.32	1.32	2	0.27	173	173	146	0.96	73
								3	0.8	122	122	208		69.3
								4	2.03	92	92	275		68.8
								5	2.97	80	80	317		63.4
								6	5.11	67	67	378		63
								7	9.5	61	61	415		59.3
Gunther	35	1*2	8	75.94	0.87	0.87	0.87	2	0.34	259	259	98	0.97	49
								3	0.77	184	184	138		46
								4	2	139	139	182		45.5
								5	3.67	119	119	213		42.6
								6	4.36	101	101	251		41.8
								7	9.92	91	91	278		39.7
Kilbrid	45	1*2	8	20.25	0.68	0.68	0.68	2	0.3	295	295	86	0.97	43
								3	0.72	209	209	121		40.3
								4	2.13	159	159	159		39.8
								5	3.31	136	136	186		37.2
								6	3.91	114	114	222		37
								7	9.28	103	103	246		35.1

Configuration: 8 Stations-Continued

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output/ Shift	R^2	Part/ Person
Hahn	53	1*4	8	13.47	231.15	231.15	231.15	2	0.28	7456	7456	3	0.93	1.5
								3	0.78	5151	5151	5		1.7
								4	2.14	3824	3824	7		1.8
								5	5.06	3422	3422	7		1.4
								6	1.22	2604	2604	10		1.7
								7	1.59	2604	2604	10		1.4
Warnecke	58	1*4	8	466.81	0.72	0.72	0.72	2	0.31	825	825	31	0.97	15.5
								3	0.81	575	575	44		14.7
								4	2.86	441	441	57		14.3
								5	3.11	380	380	67		13.4
								6	3.7	317	317	80		13.3
								7	10.69	287	287	88		12.6
Tonge	70	1*4	8	604	1.7	0.6	n/a	2	0.34	1845	1845	14	0.97	7
								3	0.64	1261	1261	20		6.7
								4	2.41	990	990	26		6.5
								5	4.89	850	850	30		6
								6	3.11	711	711	36		6
								7	9.55	641	641	40		5.7
WeeMag	75	1*4	8	72.34	0.83	0.83	0.83	2	0.31	799	799	32	0.97	16
								3	0.92	557	557	45		15
								4	2.08	428	428	59		14.8
								5	4.69	367	367	69		13.8
								6	3.27	308	308	82		13.7
								7	9.19	278	278	91		13
Cell842	79	1*2	8	680.73	1.06	1.06	1.06	2	0.27	187	187	135	0.97	67.5
								3	0.91	136	136	186		62
								4	1.83	100	100	253		63.3
								5	3.45	87	87	291		58.2
								6	3.66	73	73	347		57.8
								7	8.95	66	66	384		54.9

Configuration: 9 Stations

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output/ Shift	R^2	Part/ Person
Mitchell	21	original	9	0.91	0.6	0.6	0.6	2	0.56	71	71	357	0.95	178.5
								3	1.31	49	49	517		172.3
								4	3.45	38	38	666		166.5
								5	9.09	32	32	791		158.2
								6	17.39	28	28	904		150.7
								7	1.44	23	23	1101		157.3
								8	1.06	23	23	1101		137.6
Roszieg	25	original	9	19.42	1.75	1.75	1.75	2	0.42	69	69	367	0.93	183.5
								3	1.5	46	46	550		183.3
								4	3.31	37	37	684		171
								5	7.84	32	32	791		158.2
								6	17.98	27	27	938		156.3
								7	1.69	23	23	1101		157.3
								8	0.94	23	23	1101		137.6
Sawyer	30	original	9	72.95	0.84	0.84	0.84	2	0.47	171	171	148	0.89	74
								3	1.73	113	113	224		74.7
								4	3.03	88	88	288		72
								5	5.36	76	76	333		66.6
								6	17.61	66	66	384		64
								7	2.25	58	58	437		62.4
								8	2.45	58	58	437		54.6
Gunther	35	1*2	9	41.76	0.81	0.81	0.81	2	0.78	260	260	97	0.90	48.5
								3	1.42	169	169	150		50
								4	3.59	133	133	190		47.5
								5	6.78	115	115	220		44
								6	22.44	100	100	253		42.2
								7	2.01	87	87	291		41.6
								8	1.26	87	87	291		36.4
Kilbrid	45	1*2	9	21.3	0.64	0.64	0.64	2	0.58	296	296	86	0.88	43
								3	1.19	192	192	132		44
								4	1.97	151	151	168		42
								5	6.28	130	130	195		39
								6	18.14	113	113	224		37.3
								7	2	100	100	253		36.1
								8	1.67	100	100	253		31.6

Configuration: 9 Stations-Continued

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output/ Shift	R^2	Part/ Person
Hahn	53	1*4	9	254.45	25.7	25.7	25.7	2	0.56	7329	7329	3	0.88	1.5
								3	1.55	4769	4769	5		1.7
								4	2.42	3701	3701	7		1.8
								5	5.59	3204	3204	8		1.6
								6	15.01	2675	2675	9		1.5
								7	2.44	2552	2552	10		1.4
								8	1.86	2552	2552	10		1.3
Warnecke	58	1*4	9	2101	2.36	0.64	n/a	2	0.61	821	821	31	0.86	15.5
								3	1.27	532	532	48		16
								4	2.67	417	417	61		15.3
								5	4.61	362	362	70		14
								6	21.98	310	310	82		13.7
								7	2.11	281	281	90		12.9
								8	1.02	281	281	90		11.3
Tonge	70	1*4	9	6585	1.7	0.6	n/a	2	0.58	1840	1840	14	0.86	7
								3	1.39	1201	1201	21		7
								4	3.67	925	925	27		6.8
								5	4.95	815	815	31		6.2
								6	18.83	682	682	37		6.2
								7	1.64	633	633	40		5.7
								8	2.17	633	633	40		5
WeeMag	75	1*4	9	5181.55	1.06	0.82	n/a	2	0.61	795	795	32	0.88	16
								3	1.19	517	517	49		16.3
								4	2.86	403	403	63		15.8
								5	4.34	352	352	72		14.4
								6	20.81	301	301	84		14
								7	1.94	270	270	94		13.4
								8	2.14	270	270	94		11.8
Cell 842	79	1*2	9	730	4.32	0.72	4.32	2	0.84	191	191	133	0.85	66.5
								3	1.48	127	127	199		66.3
								4	4.16	101	101	251		62.8
								5	8.13	78	78	325		65
								6	17.66	70	70	362		60.3
								7	2.73	66	66	384		54.9
								8	2.11	66	66	384		48

Configuration: 10 Stations

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output/ Shift	R^2	Part/ Person
Mitchell	21	original	10	1.44	0.6	0.6	0.6	2	0.66	75	75	338	0.96	169
								3	3.55	49	49	517		172.3
								4	11.28	41	41	618		154.5
								5	39.69	33	33	767		153.4
								6	99.64	30	30	844		140.7
								7	197.97	28	28	904		129.1
								8	28.55	22	22	1151		143.9
								9	50.2	20	20	1266		140.7
Roszieg	25	original	10	1643	4.25	2.25	n/a	2	0.83	70	70	362	0.92	181
								3	2.13	49	49	517		172.3
								4	7.7	37	37	684		171
								5	30.49	31	31	817		163.4
								6	10.75	30	30	844		140.7
								7	3.89	25	25	1013		144.7
								8	4.58	23	23	1101		137.6
								9	10.61	21	21	1206		134
Sawyer	30	original	10	7.89	1.12	1.12	1.12	2	0.92	176	176	144	0.96	72
								3	2.94	117	117	216		72
								4	9.19	94	94	269		67.3
								5	39.44	79	79	321		64.2
								6	2.91	69	69	367		61.2
								7	4.55	66	66	384		54.9
								8	4.23	51	51	496		62
								9	5.81	47	47	539		59.9
Gunther	35	1*2	10	6658	4.19	4.19	4.19	2	0.81	266	266	95	0.97	47.5
								3	2.63	180	180	141		47
								4	9.51	146	146	173		43.3
								5	48.47	119	119	213		42.6
								6	2.67	104	104	243		40.5
								7	3.28	99	99	256		36.6
								8	6.22	78	78	325		40.6
								9	6.97	73	73	347		38.6
Kilbrid	45	1*2	10	6893	0.76	0.64	n/a	2	0.77	303	303	84	0.96	42
								3	3.06	199	199	127		42.3
								4	7.72	163	163	155		38.8
								5	56.86	134	134	189		37.8
								6	2.95	120	120	211		35.2
								7	4.33	114	114	222		31.7
								8	5.59	87	87	291		36.4
								9	4.95	81	81	313		34.8

Configuration: 10 Stations-Continued

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output/ Shift	R^2	Part/ Person
Hahn	53	1*4	10	64.16	151.45	151.45	151.45	2	0.78	7142	7142	4	0.96	2
								3	2.63	4858	4858	5		1.7
								4	6.66	3749	3749	7		1.8
								5	30.55	3256	3256	8		1.6
								6	0.89	2848	2848	9		1.5
								7	0.91	2548	2548	10		1.4
								8	1.09	2029	2029	12		1.5
								9	0.69	1976	1976	13		1.4
Warnecke	58	1*4	10	1628	2.64	0.64	n/a	2	0.78	835	835	30	0.96	15
								3	2.41	553	553	46		15.3
								4	11.44	451	451	56		14
								5	5.75	371	371	68		13.6
								6	3.03	331	331	76		12.7
								7	3.3	314	314	81		11.6
								8	3.58	242	242	105		13.1
								9	5.92	230	230	110		12.2
Tonge	70	1*4	10	128481	1.1	0.3	n/a	2	1.03	1856	1856	14	0.97	7
				330839	1.1	0.3		3	3.13	1216	1216	21		7
								4	7.03	998	998	25		6.3
								5	36.59	824	824	31		6.2
								6	1.58	726	726	35		5.8
								7	3.44	664	664	38		5.4
								8	4.47	534	534	47		5.9
								9	6.77	498	498	51		5.7
WeeMag	75	1*4	10	8495	3.09	0.87	n/a	2	0.76	807	807	31	0.96	15.5
								3	2.61	536	536	47		15.7
								4	7.64	439	439	58		14.5
								5	53.8	359	359	71		14.2
								6	87.59	323	323	78		13
								7	3.77	301	301	84		12
								8	4.59	234	234	108		13.5
								9	4.78	220	220	115		12.8
Cell842	79	1*2	10	2119.33	8.32	8.32	8.32	2	0.72	191	191	133	0.93	66.5
								3	2.17	126	126	201		67
								4	9.31	110	110	230		57.5
								5	40.31	84	84	301		60.2
								6	2.84	78	78	325		54.2
								7	4.36	75	75	338		48.3
								8	7.52	64	64	396		49.5
								9	5.7	51	51	496		55.1

Configuration: 11 Stations

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output/ Shift	R^2	Part/ Person
Mitchell	21	original	11	20.92	1.6	1.6	1.6	2	1.03	71	71	357	0.98	178.5
								3	4	50	50	506		168.7
								4	23.16	40	40	633		158.3
								5	75.64	32	32	791		158.2
								6	224.81	27	27	938		156.3
								7	12.14	25	25	1013		144.7
								8	14.34	22	22	1151		143.9
								9	2.73	18	18	1407		156.3
								10	3.17	18	18	1407		140.7
Roszieg	25	original	11	359.53	1.75	1.75	1.75	2	0.78	68	68	372	0.97	186
								3	5	47	47	539		179.7
								4	22.58	37	37	684		171
								5	56.09	32	32	791		158.2
								6	13.63	26	26	974		162.3
								7	39.48	23	23	1101		157.3
								8	9.25	21	21	1206		150.8
								9	2.86	18	18	1407		156.3
								10	2.95	18	18	1407		140.7
Sawyer	30	original	11	2883	2.2	1.8	n/a	2	1.03	171	171	148	0.96	74
								3	4.44	119	119	213		71
								4	18	94	94	269		67.3
								5	47.94	76	76	333		66.6
								6	3.34	60	60	422		70.3
								7	12.42	60	60	422		60.3
								8	12.25	51	51	496		62
								9	2.52	44	44	575		63.9
								10	2.27	44	44	575		57.5
Gunther	35	1*2	11	6.81	17.85	17.85	17.85	2	1.2	263	263	96	0.91	48
								3	3.44	180	180	141		47
								4	18.91	135	135	188		47
								5	86.97	113	113	224		44.8
								6	3.49	94	94	269		44.8
								7	3.92	88	88	288		41.1
								8	19.95	86	86	294		36.8
								9	1.2	72	72	352		39.1
								10	1.28	72	72	352		35.2
Kilbrid	45	1*2	11	4834	0.64	0.6	n/a	2	1	299	299	85	0.97	42.5
								3	4.89	203	203	125		41.7
								4	17.22	167	167	152		38
								5	59.45	130	130	195		39
								6	3.34	103	103	246		41
								7	1.7	98	98	258		36.9
								8	11.95	93	93	272		34
								9	3.59	72	72	352		39.1
								10	3.05	72	72	352		35.2

Configuration: 11 Stations-Continued

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output/ Shift	R^2	Part/ Person
Hahn	53	1*4	11	473.16	383.25	383.25	383.25	2	0.88	7095	7095	4	0.95	2
								3	3.83	4949	4949	5		1.7
								4	9.89	3724	3724	7		1.8
								5	42.92	3064	3064	8		1.6
								6	8.27	2789	2789	9		1.5
								7	5.94	2438	2438	10		1.4
								8	5.95	2020	2020	13		1.6
								9	13.05	1953	1953	13		1.4
								10	2.56	1855	1855	14		1.4
Warnecke	58	1*4	11	2325	6.6	0.6	n/a	2	0.97	838	838	30	0.96	15
								3	5.09	553	553	46		15.3
								4	23.13	464	464	55		13.8
								5	50.63	368	368	69		13.8
								6	102.58	286	286	89		14.8
								7	2.02	276	276	92		13.1
								8	2.69	262	262	97		12.1
								9	0.98	201	201	126		14
								10	1.26	201	201	126		12.6
Tonge	70	1*4	11	863.66	2.6	0.5	n/a	2	1.25	1835	1835	14	0.96	7
								3	3.71	1246	1246	20		6.7
								4	29.33	1010	1010	25		6.3
								5	14.63	791	791	32		6.4
								6	7.38	640	640	40		6.7
								7	27.42	606	606	42		6
								8	158.2	571	571	44		5.5
								9	2.06	458	458	55		6.1
								10	1.56	458	458	55		5.5
WeeMag	75	1*4	11	19.5	2.05	0.87	n/a	2	0.97	805	805	31	0.97	15.5
								3	4.39	545	545	46		15.3
								4	34.14	445	445	57		14.3
							`	5	48.92	348	348	73		14.6
								6	2.63	278	278	91		15.2
								7	13.14	265	265	96		13.7
								8	12.81	246	246	103		12.9
								9	2.41	193	193	131		14.6
								10	2.89	193	193	131		13.1
Cell 842	79	1*2	11	446.83	1.7	1.7	1.7	2	1.13	192	192	132	0.98	66
								3	5.02	131	131	193		64.3
								4	17.28	106	106	239		59.8
								5	67.66	85	85	298		59.6
								6	5.06	66	66	384		64
								7	10.91	65	65	390		55.7
								8	12.53	57	57	444		55.5
								9	2.84	45	45	563		62.6
								10	2.98	45	45	563		56.3

Configuration: 12 Stations

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output/ Shift	R^2	Part/ Person
Mitchell	21	original	12	7.81	0.6	0.6	0.6	2	1.45	74	74	342	0.95	171
								3	8.86	54	54	469		156.3
								4	42.06	41	41	618		154.5
								5	377.56	32	32	791		158.2
								6	16.73	29	29	873		145.5
								7	4.98	26	26	974		139.1
								8	2.89	24	24	1055		131.9
								9	2.16	22	22	1151		127.9
								10	2.74	20	20	1266		126.6
								11	3.05	18	18	1407		127.9
Roszieg	25	original	12	5959	4.75	2.75	n/a	2	1.17	74	74	342	0.97	171
								3	7.81	49	49	517		172.3
								4	35.13	37	37	684		171
								5	7.09	31	31	817		163.4
								6	7.31	27	27	938		156.3
								7	22.14	27	27	938		134
								8	1.34	22	22	1151		143.9
								9	1.59	18	18	1407		156.3
								10	1.84	18	18	1407		140.7
								11	2.13	16	16	1583		143.9
Sawyer	30	original	12	55.22	1.08	1.08	1.08	2	1.56	171	171	148	0.96	74
								3	7.81	124	124	204		68
								4	9.61	93	93	272		68
								5	32.41	73	73	347		69.4
								6	11.8	67	67	378		63
								7	2.55	62	62	408		58.3
								8	3.05	55	55	460		57.5
								9	5.03	50	50	506		56.2
								10	4.61	44	44	575		57.5
								11	6.8	42	42	603		54.8
Lutz	32	1*4	12	96.55	96.4	96.4	96.4	2	1.17	7236	7236	3	0.94	1.5
								3	6.31	5096	5096	5		1.7
								4	33.51	3786	3786	7		1.8
								5	5.01	3046	3046	8		1.6
								6	15.53	2740	2740	9		1.5
								7	1.83	2454	2454	10		1.4
								8	1.86	2348	2348	11		1.4
								9	3.45	2230	2230	11		1.2
								10	3.61	1906	1906	13		1.3
								11	2.22	1628	1628	16		1.5

Configuration: 12 Stations-Continued

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output/ Shift	R^2	Part/ Person
Gunther	35	1*2	12	4067.55	2.7	1.2	n/a	2	1.5	259	259	98	0.96	49
								3	6.42	185	185	137		45.7
								4	10.83	147	147	172		43
								5	25.31	108	108	234		46.8
								6	3.23	105	105	241		40.2
								7	5.2	93	93	272		38.9
								8	9.56	85	85	298		37.3
								9	25.91	78	78	325		36.1
								10	31.97	63	63	402		40.2
								11	2.25	61	61	415		37.7
Kilbrid	45	1*2	12	2560	1.84	0.84	n/a	2	1.16	295	295	86	0.96	43
								3	9.09	214	214	118		39.3
								4	13.36	161	161	157		39.3
								5	24.3	124	124	204		40.8
								6	31.16	119	119	213		35.5
								7	45.36	103	103	246		35.1
								8	36.38	94	94	269		33.6
								9	2.17	89	89	284		31.6
								10	2.39	77	77	329		32.9
								11	3.2	70	70	362		32.9
Hahn	53	1*4	12	156.72	433.2	433.2	433.2	2	1.25	7177	7177	4	0.93	2
								3	7.61	5039	5039	5		1.7
								4	33.66	3740	3740	7		1.8
								5	109.95	3002	3002	8		1.6
								6	1.89	2711	2711	9		1.5
								7	2.2	2363	2363	11		1.6
								8	4.75	2133	2133	12		1.5
								9	10.41	2021	2021	13		1.4
								10	1.86	1775	1775	14		1.4
								11	2.27	1775	1775	14		1.3
Warnecke	58	1*4	12	1295.84	3.68	1.68	n/a	2	1.45	809	809	31	0.96	15.5
								3	7.06	587	587	43		14.3
								4	43.7	441	441	57		14.3
								5	7.23	346	346	73		14.6
								6	27.95	317	317	80		13.3
								7	66.09	294	294	86		12.3
								8	1.52	258	258	98		12.3
								9	1.8	244	244	104		11.6
								10	2.44	209	209	121		12.1
								11	2.88	196	196	129		11.7

Configuration: 12 Stations-Continued

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output/ Shift	R^2	Part/ Person
Tonge	70	1*4	12	16413	5.1	3.2	n/a	2	1.41	1804	1804	14	0.95	7
								3	10.11	1290	1290	20		6.7
								4	33.92	954	954	27		6.8
								5	4.66	755	755	34		6.8
								6	17.61	696	696	36		6
								7	2.01	643	643	39		5.6
								8	1.56	573	573	44		5.5
								9	1.91	537	537	47		5.2
								10	2.53	468	468	54		5.4
								11	2.89	424	424	60		5.5
Dept842	79	1*2	12	8467	8.32	5.86	5.86	2	1.34	189	189	134	0.94	67
								3	7.56	132	132	192		64
								4	10.22	99	99	256		64
								5	41.7	81	81	313		62.6
								6	13.7	71	71	357		59.5
								7	22.11	64	64	396		56.6
								8	80.03	61	61	415		51.9
								9	213.69	57	57	444		49.3
								10	2.74	52	52	487		48.7
								11	3.05	43	43	589		53.5

Configuration: 13 Stations

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output/ Shift	R^2	Part/ Person
Mitchell	21	original	13	3.47	0.6	0.6	0.6	2	2.58	72	72	352	0.98	176
								3	14.92	51	51	496		165.3
								4	26.94	40	40	633		158.3
								5	60.02	30	30	844		168.8
								6	50.86	28	28	904		150.7
								7	4.67	25	25	1013		144.7
								8	2.58	23	23	1101		137.6
								9	4.31	20	20	1266		140.7
								10	4.86	18	18	1407		140.7
								11	3.28	15	15	1688		153.5
								12	3.83	15	15	1688		140.7
Roszieg	25	original	13	6546	4.25	1.75	n/a	2	2.39	73	73	347	0.95	173.5
								3	15.48	48	48	528		176
								4	22.55	37	37	684		171
								5	23	32	32	791		158.2
								6	55.06	26	26	974		162.3
								7	18	24	24	1055		150.7
								8	4.11	21	21	1206		150.8
								9	12.3	20	20	1266		140.7
								10	25.61	19	19	1333		133.3
								11	63.44	18	18	1407		127.9
								12	3.89	14	14	1809		150.8
Sawyer	30	original	13	5319	1.88	1.88	1.88	2	0.33	171	171	148	0.97	74
								3	15.31	118	118	215		71.7
								4	71.83	92	92	275		68.8
								5	21.31	72	72	352		70.4
								6	82.72	68	68	372		62
								7	180.08	59	59	429		61.3
								8	135.05	54	54	469		58.6
								9	15.16	46	46	550		61.1
								10	14.89	39	39	649		64.9
								11	3.47	37	37	684		62.2
								12	4.16	37	37	684		57
Gunther	35	1*2	13	1267	4.51	4.51	4.51	2	2.09	257	257	99	0.98	49.5
								3	17.08	182	182	139		46.3
								4	19.2	138	138	183		45.8
								5	23.86	109	109	232		46.4
								6	65.33	100	100	253		42.2
								7	7.2	90	90	281		40.1
								8	2.81	79	79	321		40.1
								9	3.66	68	68	372		41.3
								10	4.69	57	57	444		44.4
								11	3.16	55	55	460		41.8
								12	3.83	55	55	460		38.3

Configuration: 13 stations-continued

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output/ Shift	R^2	Part/ Person
Kilbrid	45	1*2	13	10636	0.68	0.56	n/a	2	2.3	292	292	87	0.96	43.5
								3	17.2	204	204	124		41.3
								4	25.69	156	156	162		40.5
								5	71.98	120	120	211		42.2
								6	10.38	112	112	226		37.7
								7	24.5	98	98	258		36.9
								8	38.53	92	92	275		34.4
								9	4.24	79	79	321		35.7
								10	4.97	71	71	357		35.7
								11	3.17	66	66	384		34.9
								12	3.86	66	66	384		32
Hahn	53	1*4	13	375.95	371.5	371.5	371.5	2	2.41	7155	7155	4	0.80	2
								3	12.39	4838	4838	5		1.7
								4	83	3795	3795	7		1.8
								5	3.73	2990	2990	8		1.6
								6	4.94	2686	2686	9		1.5
								7	8.7	2413	2413	10		1.4
								8	11.64	2161	2161	12		1.5
								9	2.09	2012	2012	13		1.4
								10	2.69	2012	2012	13		1.3
								11	3.47	2012	2012	13		1.2
								12	3.88	2012	2012	13		1.1
Warnecke	58	1*4	13	348	6.12	0.56	N/A	2	2.77	797	797	32	0.92	16
								3	12.83	545	545	46		15.3
								4	22.64	420	420	60		15
								5	39.25	333	333	76		15.2
								6	2.81	299	299	85		14.2
								7	3.34	262	262	97		13.9
								8	5.58	249	249	102		12.8
								9	5.91	207	207	122		13.6
								10	2.81	191	191	133		13.3
								11	3.34	191	191	133		12.1
								12	4.16	191	191	133		11.1
Tonge	70	1*4	13	4000	9.9	0.5	N/A	2	2.59	1790	1790	14	0.92	7
								3	17.11	1236	1236	20		6.7
								4	64.69	934	934	27		6.8
								5	16.22	753	753	34		6.8
								6	80.66	679	679	37		6.2
								7	101.86	589	589	43		6.1
								8	6.97	551	551	46		5.8
								9	7.75	463	463	55		6.1
								10	2.75	431	431	59		5.9
								11	3.39	431	431	59		5.4
								12	4.13	431	431	59		4.9

Configuration: 13 Stations-continued

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output/ Shift	R^2	Part/ Person
WeeMag	75	1*4	13	5425.92	3.06	0.88	n/a	2	1.98	787	787	32	0.96	16
								3	19.03	548	548	46		15.3
								4	16.06	418	418	61		15.3
								5	17.03	325	325	78		15.6
								6	75.05	305	305	83		13.8
								7	23.98	257	257	99		14.1
								8	49.56	248	248	102		12.8
								9	2.52	211	211	120		13.3
								10	2.73	190	190	133		13.3
								11	3.14	178	178	142		12.9
								12	3.75	178	178	142		11.8
Cell842	79	1*2	13	2779	4.86	4.86	4.86	2	2.06	186	186	136	0.93	68
								3	13.81	130	130	195		65
								4	88.02	103	103	246		61.5
								5	16.17	79	79	321		64.2
								6	7.99	72	72	352		58.7
								7	35.52	70	70	362		51.7
								8	6.48	56	56	452		56.5
								9	3.5	51	51	496		55.1
								10	2.97	43	43	589		58.9
								11	3.17	43	43	589		53.5
								12	3.81	43	43	589		49.1

Configuration: 14 Stations

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output / Shift	R^2	Part/ Person
Mitchell	21	original	14	3.53	0.5	0.5	0.5	2	3.63	74	74	342	0.91	171
								3	26.88	52	52	487		162.3
								4	38.94	39	39	649		162.3
								5	60.84	31	31	817		163.4
								6	211.83	29	29	873		145.5
								7	211.44	22	22	1151		164.4
								8	6.81	21	21	1206		150.8
								9	17.17	20	20	1266		140.7
								10	141.92	20	20	1266		126.6
								11	209.89	18	18	1407		127.9
								12	6.09	17	17	1489		124.1
								13	7.44	16	16	1583		121.8
Roszieg	25	original	14	6498.1	4.25	1.75	n/a	2	3.55	71	71	357	0.92	178.5
								3	18.5	49	49	517		172.3
								4	117.33	37	37	684		171
								5	36.88	30	30	844		168.8
								6	103.26	27	27	938		156.3
								7	72.05	23	23	1101		157.3
								8	91.27	21	21	1206		150.8
								9	382.81	19	19	1333		148.1
								10	5.22	18	18	1407		140.7
								11	5.78	17	17	1489		135.4
								12	5.69	16	16	1583		131.9
								13	6.58	16	16	1583		121.8
Sawyer	30	original	14	1608	1.8	1.8	1.8	2	3.22	171	171	148	0.90	74
								3	22.28	121	121	209		69.7
								4	37.86	89	89	284		71
								5	51.84	73	73	347		69.4
								6	174.17	64	64	396		66
								7	43.51	50	50	506		72.3
								8	2.81	49	49	517		64.6
								9	8.44	47	47	539		59.9
								10	7.72	44	44	575		57.5
								11	29.63	44	44	575		52.3
								12	55.84	40	40	633		52.8
								13	84.06	36	36	703		54.1

Configuration: 14 Station-continued

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output / Shift	R^2	Part/ Person
Lutz	32	1*4	14	691.31	124.7	124.7	124.7	2	2.09	7190	7190	4	0.96	2
								3	16.25	4912	4912	5		1.7
								4	45.06	3866	3866	7		1.8
								5	142.59	3016	3016	8		1.6
								6	128.72	2534	2534	10		1.7
								7	60.13	2220	2220	11		1.6
								8	82.33	2052	2052	12		1.5
								9	5.69	1846	1846	14		1.6
								10	9.33	1810	1810	14		1.4
								11	15.02	1642	1642	15		1.4
								12	29.78	1480	1480	17		1.4
								13	6.84	1432	1432	18		1.4
Gunther	35	1*2	14	17.83	17.85	17.85	17.85	2	3.25	265	265	96	0.80	48
								3	22.48	180	180	141		47
								4	42.7	140	140	181		45.3
								5	143.67	109	109	232		46.4
								6	181.31	99	99	256		42.7
								7	432.33	86	86	294		42
								8	7.27	71	71	357		44.6
								9	3.92	65	65	390		43.3
								10	3.91	65	65	390		39
								11	4.78	65	65	390		35.5
								12	5.75	65	65	390		32.5
								13	6.73	65	65	390		30
Kilbrid	45	1*2	14	7017.9	5.32	2.32	n/a	2	3.5	294	294	86	0.93	43
								3	21.55	207	207	122		40.7
								4	115.48	150	150	169		42.3
								5	181.03	126	126	201		40.2
								6	331.92	109	109	232		38.7
								7	102.56	88	88	288		41.1
								8	3.78	84	84	301		37.6
								9	5.22	78	78	325		36.1
								10	13.11	78	78	325		32.5
								11	24.09	73	73	347		31.5
								12	50.91	67	67	378		31.5
								13	6.63	59	59	429		33

Configuration: 14 Station-continued

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output / Shift	R^2	Part/ Person
Hahn	53	1*4	14	390.38	383.25	383.25		2	3.2	7075	7075	4	0.76	2
								3	24.38	4784	4784	5		1.7
								4	96.52	3681	3681	7		1.8
								5	35.36	3047	3047	8		1.6
								6	127.48	2666	2666	9		1.5
								7	4.98	2200	2200	12		1.7
								8	8.44	1986	1986	13		1.6
								9	15.19	1916	1916	13		1.4
								10	5.47	1775	1775	14		1.4
								11	4.83	1775	1775	14		1.3
								12	5.8	1775	1775	14		1.2
								13	6.81	1775	1775	14		1.1
Warnecke	58	1*2	14	6565.7	4.64	1.64	n/a	2	3.13	794	794	32	0.93	16
								3	21.8	549	549	46		15.3
								4	36.51	411	411	62		15.5
								5	194.47	337	337	75		15
								6	163.39	293	293	86		14.3
								7	34.22	235	235	108		15.4
								8	6.75	226	226	112		14
								9	20.97	218	218	116		12.9
								10	62.72	199	199	127		12.7
								11	4.63	192	192	132		12
								12	5.94	180	180	141		11.8
								13	7.16	159	159	159		12.2
Wee-Mag	75	1*4	14	27843	2.07	1.1	n/a	2	3.19	787	787	32	0.94	16
								3	24.86	548	548	46		15.3
								4	29.73	408	408	62		15.5
								5	180.05	335	335	76		15.2
								6	295.88	291	291	87		14.5
								7	52.11	236	236	107		15.3
								8	4.44	221	221	115		14.4
								9	14.86	214	214	118		13.1
								10	36.77	198	198	128		12.8
								11	158.97	192	192	132		12
								12	5.86	176	176	144		12
								13	7.38	159	159	159		12.2
dept 842	79	1*2	14	2805	4.86	4.86	4.86	2	3.33	189	189	134	0.92	67
								3	25.75	134	134	189		63
								4	40.75	96	96	264		66
								5	61.94	82	82	309		61.8
								6	238.95	71	71	357		59.5
								7	76.69	57	57	444		63.4
								8	18.66	56	56	452		56.5
								9	42.25	54	54	469		52.1
								10	114.27	48	48	528		52.8
								11	220.28	46	46	550		50
								12	582.64	42	42	603		50.3
								13	11.66	41	41	618		47.5

Configuration: 15 Stations

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output / Shift	R^2	Part/ Person
Mitchell	21	original	15	121.05	2.9	2.9	2.9	2	4.8	74	74	342	0.97	171
								3	50.36	50	50	506		168.7
								4	107.11	39	39	649		162.3
								5	184.28	31	31	817		163.4
								6	266.69	26	26	974		162.3
								7	121.76	23	23	1101		157.3
								8	128.3	23	23	1101		137.6
								9	415.03	22	22	1151		127.9
								10	445.08	18	18	1407		140.7
								11	225.72	17	17	1489		135.4
								12	191.13	15	15	1688		140.7
								13	249.22	14	14	1809		139.2
								14	10.64	13	13	1948		139.1
Roszieg	25	original	15	119.56	1.75	1.75	1.75	2	6.33	73	73	347	0.95	173.5
								3	49.17	49	49	517		172.3
								4	394.5	38	38	666		166.5
								5	123.31	32	32	791		158.2
								6	420.59	27	27	938		156.3
								7	2196.45	24	24	1055		150.7
								8	726.91	21	21	1206		150.8
								9	183.25	19	19	1333		148.1
								10	652.56	18	18	1407		140.7
								11	232.91	18	18	1407		127.9
								12	320.31	16	16	1583		131.9
								13	456	15	15	1688		129.8
								14	1477.27	14	14	1809		129.2
Sawyer	30	original	15	84.44	2.84	2.84	2.84	2	5.05	173	173	146	0.97	73
								3	39.9	117	117	216		72
								4	374.56	91	91	278		69.5
								5	649.01	74	74	342		68.4
								6	622.3	64	64	396		66
								7	1451	54	54	469		67
								8	543.7	50	50	506		63.3
								9	33.53	46	46	550		61.1
								10	25.03	44	44	575		57.5
								11	58.58	38	38	666		60.5
								12	27.13	37	37	684		57
								13	31.83	34	34	745		57.3
								14	11.05	32	32	791		56.5

Configuration: 15 Station-continued

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output / Shift	R^2	Part/ Person
Gunther	35	1*2	15	7561	2.64	0.98	n/a	2	5.52	272	272	93	0.93	46.5
								3	44.09	179	179	141		47
								4	146.88	141	141	180		45
								5	392.44	114	114	222		44.4
								6	667	95	95	267		44.5
								7	5070.34	86	86	294		42
								8	783.94	72	72	352		44
								9	295.47	70	70	362		40.2
								10	430.66	66	66	384		38.4
								11	10.91	61	61	415		37.7
								12	16.73	59	59	429		35.8
								13	43.99	58	58	437		33.6
								14	81.77	52	52	487		34.8
Kilbrid	45	1*2	15	884	10.84	0.68	n/a	2	6	297	297	85	0.96	42.5
								3	47.59	204	204	124		41.3
								4	81.69	156	156	162		40.5
								5	757.03	131	131	193		38.6
								6	467.67	103	103	246		41
								7	303.75	99	99	256		36.6
								8	2025	85	85	298		37.3
								9	135.88	77	77	329		36.6
								10	519	72	72	352		35.2
								11	8.5	67	67	378		34.4
								12	8.42	66	66	384		32
								13	39.98	62	62	408		31.4
								14	11.3	55	55	460		32.9
Hahn	53	1*4	15	2206	603.85	129.85	n/a	2	5.34	7136	7136	4	0.65	2
								3	38.64	4764	4764	5		1.7
								4	35.02	3686	3686	7		1.8
								5	64.39	3037	3037	8		1.6
								6	367.52	2596	2596	10		1.7
								7	48.64	2136	2136	12		1.7
								8	22.67	2031	2031	12		1.5
								9	4.19	1867	1867	14		1.6
								10	5.42	1867	1867	14		1.4
								11	6.64	1867	1867	14		1.3
								12	7.86	1867	1867	14		1.2
								13	9.5	1867	1867	14		1.1
								14	11.34	1867	1867	14		1

Configuration: 15 Stations-continued

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output / Shift	R^2	Part/ Person
Warnecke	58	1*2	15	6847	5.4	1.4	n/a	2	5.98	825	825	31	0.93	15.5
								3	46.3	549	549	46		15.3
								4	96.78	423	423	60		15
								5	439.285	343	343	74		14.8
								6	423.23	284	284	89		14.8
								7	6923	256	256	99		14.1
								8	491.98	223	223	114		14.3
								9	7.67	221	221	115		12.8
								10	18.94	196	196	129		12.9
								11	14.19	186	186	136		12.4
								12	41.09	181	181	140		11.7
								13	50.09	170	170	149		11.5
								14	11.83	160	160	158		11.3
Tonge	70	1*4	15	5056	7.5	2.5	n/a	2	5.23	1873	1873	14	0.93	7
								3	49.45	1230	1230	21		7
								4	89.06	970	970	26		6.5
								5	812.27	770	770	33		6.6
								6	6584	650	650	39		6.5
								7	76533	568	568	45		6.4
								8	490.72	504	504	50		6.3
								9	15.77	490	490	52		5.8
								10	34.34	440	440	58		5.8
								11	80.13	428	428	59		5.4
								12	211.2	394	394	64		5.3
								13	9.5	385	385	66		5.1
								14	12.06	355	355	71		5.1
WeeMag	75	1*4	15	46206	4.93	1.91	n/a	2	4.59	824	824	31	0.92	15.5
								3	42.39	551	551	46		15.3
								4	98.34	420	420	60		15
								5	477.55	338	338	75		15
								6	642.19	281	281	90		15
								7	5600.89	256	256	99		14.1
								8	1189	229	229	111		13.9
								9	5619	216	216	117		13
								10	40.63	199	199	127		12.7
								11	17.22	186	186	136		12.4
								12	40.59	182	182	139		11.6
								13	112.65	173	173	146		11.2
								14	19.42	159	159	159		11.4
Dept 842	79	1*2	15	2049	8.32	8.32	8.32	2	4.53	195	195	130	0.96	65
								3	45.14	130	130	195		65
								4	67.39	104	104	243		60.8
								5	427.91	83	83	305		61
								6	350.98	69	69	367		61.2
								7	604.55	64	64	396		56.6
								8	2216	58	58	437		54.6
								9	97.39	48	48	528		58.7
								10	9.92	45	45	563		56.3
								11	13.36	45	45	563		51.2
								12	39.34	42	42	603		50.3
								13	38.19	38	38	666		51.2
								14	11.31	36	36	703		50.2

Configuration: 16 Stations

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	**T**	**L[i,j]**	**M**	**Solving Time (sec)**	**Best Bound**	**Lower Bound**	**Optimal Solution**	**N**	**Solving Time (sec)**	**Best Solution**	**Optimal Solution**	**Output / Shift**	**R^2**	**Part/ Person**
Mitchell	21	original	16	49.78	2.6	2.6	2.6	2	8.08	76	76	333	0.92	166.5
								3	59.45	52	52	487		162.3
								4	132.77	39	39	649		162.3
								5	260.47	34	34	745		149
								6	827.59	28	28	904		150.7
								7	19866	27	27	938		134
								8	2888.45	21	21	1206		150.8
								9	7.88	20	20	1266		140.7
								10	12.14	19	19	1333		133.3
								11	8.74	15	15	1688		153.5
								12	10.49	15	15	1688		140.7
								13	12.26	15	15	1688		129.8
								14	16.14	15	15	1688		120.6
								15	41.92	15	15	1688		112.5
Roszieg	25	original	16	201	4.25	1	n/a	2	6.14	72	72	352	0.83	176
				49891	4.25	2.25		3	62.97	50	50	506		168.7
								4	137.41	39	39	649		162.3
								5	195.03	32	32	791		158.2
								6	685.11	26	26	974		162.3
								7	55.61	23	23	1101		157.3
								8	31.45	22	22	1151		143.9
								9	46.63	19	19	1333		148.1
								10	42.03	17	17	1489		148.9
								11	8.83	16	16	1583		143.9
								12	10.33	16	16	1583		131.9
								13	12.5	16	16	1583		121.8
								14	15.34	16	16	1583		113.1
								15	18.11	16	16	1583		105.5
Sawyer	30	original	16	6819	4.8	1.04	n/a	2	7.28	175	175	145	0.97	72.5
								3	60.75	119	119	213		71
								4	115.66	92	92	275		68.8
								5	230.33	75	75	338		67.6
								6	1411.67	63	63	402		67
								7	1448	59	59	429		61.3
								8	862.56	49	49	517		64.6
								9	9.55	47	47	539		59.9
								10	13.88	44	44	575		57.5
								11	30	39	39	649		59
								12	16.5	37	37	684		57
								13	12.94	32	32	791		60.8
								14	15.31	32	32	791		56.5
								15	17.92	32	32	791		52.7

Configuration: 16 Stations-continued

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output / Shift	R^2	Part/ Person
Gunther	35	1*2	16	127.56	18.47	18.47	18.47	2	4.72	264	264	96	0.63	48
								3	59.45	179	179	141		47
								4	90.47	140	140	181		45.3
								5	190.13	114	114	222		44.4
								6	752.2	98	98	258		43
								7	1016	85	85	298		42.6
								8	1222.53	75	75	338		42.3
								9	53.31	69	69	367		40.8
								10	7.17	66	66	384		38.4
								11	8.66	66	66	384		34.9
								12	10.55	66	66	384		32
								13	12.81	66	66	384		29.5
								14	15.47	66	66	384		27.4
								15	17.78	66	66	384		25.6
Kilbrid	45	1*2	16	3541	1.92	0.92	n/a	2	6.36	306	306	83	0.96	41.5
								3	68.2	204	204	124		41.3
								4	213.81	162	162	156		39
								5	531	132	132	192		38.4
								6	1235	112	97	226		37.7
								7	763.63	97	97	261		37.3
								8	2727.14	82	82	309		38.6
								9	9.53	82	82	309		34.3
								10	23.74	76	76	333		33.3
								11	16.77	63	63	402		36.5
								12	63.3	57	57	444		37
								13	12.8	55	55	460		35.4
								14	17.84	55	55	460		32.9
								15	58.66	55	55	460		30.7
Hahn	53	1*4	16	7337	450.08	450.08	450.08	2	6.38	7107	7107	4	0.37	2
								3	41.28	4765	4765	5		1.7
								4	131	3631	3631	7		1.8
								5	153.73	3141	3141	8		1.6
								6	1088.33	2186	2186	12		2
								7	10.25	1926	1926	13		1.9
								8	13.55	1804	1804	14		1.8
								9	7.55	1775	1775	14		1.6
								10	7.97	1775	1775	14		1.4
								11	9.02	1775	1775	14		1.3
								12	10.86	1775	1775	14		1.2
								13	13.03	1775	1775	14		1.1
								14	15.28	1775	1775	14		1
								15	22.89	1775	1775	14		0.9

Configuration: 16 Stations-continued

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output / Shift	R^2	Part/ Person
Warnecke	58	1*4	16	4544.4	6.12	0.6	n/a	2	5.34	832	832	30	0.93	15
				64475	6.12	0.6	n/a	3	67.67	564	564	45		15
								4	204	447	447	57		14.3
								5	258	365	365	69		13.8
								6	1591	304	304	83		13.8
								7	1169.89	254	254	100		14.3
								8	25.27	228	228	111		13.9
								9	21.44	222	222	114		12.7
								10	151.72	201	201	126		12.6
								11	9.75	173	173	146		13.3
								12	11.61	163	163	155		12.9
								13	12.91	160	160	158		12.2
								14	15.7	160	160	158		11.3
								15	81.14	160	160	158		10.5
Tonge	70	1*4	16	1797	20.5	0.5	n/a	2	5.28	1839	1839	14	0.88	7
				228114	20.5	0.5		3	55.36	1229	1229	21		7
								4	173.19	937	937	27		6.8
								5	1043	768	768	33		6.6
								6	697.81	640	640	40		6.7
								7	4732	573	573	44		6.3
								8	2564	506	506	50		6.3
								9	30.28	469	469	54		6
								10	58.84	429	429	59		5.9
								11	164.75	392	392	65		5.9
								12	11.09	371	371	68		5.7
								13	13.34	371	371	68		5.2
								14	16.2	371	371	68		4.9
								15	32.3	371	371	68		4.5
WeeMag	75	1*4	16	45554	8.09	0.9	n/a	2	5.44	798	798	32	0.92	16
								3	66.05	535	535	47		15.7
								4	139.22	421	421	60		15
								5	245.3	344	344	74		14.8
								6	1273	291	291	87		14.5
								7	94.05	247	247	103		14.7
								8	38.45	226	226	112		14
								9	30.2	206	206	123		13.7
								10	96.23	202	202	125		12.5
								11	189.16	165	165	153		13.9
								12	11.31	157	157	161		13.4
								13	13.11	157	157	161		12.4
								14	15.58	157	157	161		11.5
								15	24.78	157	157	161		10.7
Dept 842	79	1*2	16	3333	4.86	4.86	4.86	2	6.42	193	193	131	0.93	65.5
								3	69.23	137	137	185		61.7
								4	133.89	102	102	248		62
								5	216.23	81	81	313		62.6
								6	364.72	70	70	362		60.3
								7	410.88	61	61	415		59.3
								8	8.02	56	56	452		56.5
								9	7.78	52	52	487		54.1
								10	14.48	49	49	517		51.7
								11	11.61	37	37	684		62.2
								12	10.38	37	37	684		57
								13	13.02	37	37	684		52.6
								14	15	37	37	684		48.9
								15	18.02	37	37	684		45.6

Configuration: 17 Stations

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output / Shift	R^2	Part/ Person
Sawyer	30	original	17	17793	1.8	1.8	1.8	2	9.06	174	174	146	0.93	73
								3	117.18	119	119	213		71
								4	246.05	89	89	284		71
								5	2676	72	72	352		70.4
								6	25430	64	64	396		66
								7	31944	55	55	460		65.7
								8	17588	49	49	517		64.6
								9	1701	48	48	528		58.7
								10	2448	41	41	618		61.8
								11	4499	38	38	666		60.5
								12	54.63	36	36	703		58.6
								13	19.72	32	32	791		60.8
								14	99.7	32	32	791		56.5
								15	48.75	32	32	791		52.7
								16	50.89	32	32	791		49.4
Gunther	35	1*2	17	187.05	17.85	17.85	17.85	2	10.92	266	266	95	0.52	47.5
								3	209.22	181	181	140		46.7
								4	2956	143	143	177		44.3
								5	619.44	109	109	232		46.4
								6	480	99	99	256		42.7
								7	224.73	80	80	317		45.3
								8	109.72	75	75	338		42.3
								9	90.25	65	65	390		43.3
								10	10.11	65	65	390		39
								11	12.61	65	65	390		35.5
								12	29.05	65	65	390		32.5
								13	45.28	65	65	390		30
								14	26.66	65	65	390		27.9
								15	31.52	65	65	390		26
								16	61.03	65	65	390		24.4
Warnecke	58	1*4	17	6875	4.08	0.6	n/a	2	8.24	798	798	32	0.92	16
								3	123.67	543	543	47		15.7
								4	471.13	410	410	62		15.5
								5	421.89	332	332	76		15.2
								6	5081	295	295	86		14.3
								7	15022	243	243	104		14.9
								8	188.8	233	233	109		13.6
								9	562.08	208	208	122		13.6
								10	1372.75	192	192	132		13.2
								11	388.34	179	179	141		12.8
								12	34.2	161	161	157		13.1
								13	22.69	151	151	168		12.9
								14	132.22	151	151	168		12
								15	27.36	151	151	168		11.2
								16	45.01	151	151	168		10.5

Configuration: 17 Stations-continued

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output / Shift	R^2	Part/ Person
Tonge	70	1*4	17	78833	5.4	0.5	n/a	2	11.55	1828	1828	14	0.88	7
								3	160.08	1218	1218	21		7
								4	462.39	938	938	27		6.8
								5	893.93	752	752	34		6.8
								6	7681	654	654	39		6.5
								7	29169	547	547	46		6.6
								8	298	532	532	48		6
								9	781.88	453	453	56		6.2
								10	2558.11	432	432	59		5.9
								11	239.44	394	394	64		5.8
								12	202.22	358	358	71		5.9
								13	18.67	350	350	72		5.5
								14	21.86	350	350	72		5.1
								15	35.84	350	350	72		4.8
								16	91.86	350	350	72		4.5
Dept 842	79	1*2	17	55197	8.32	8.32	8.32	2	7.36	196	196	129	0.82	64.5
								3	132.5	134	134	189		63
								4	343.61	103	103	246		61.5
								5	432	85	85	298		59.6
								6	4116.51	70	70	362		60.3
								7	28017	64	64	396		56.6
								8	2665.92	59	59	429		53.6
								9	5680	49	49	517		57.4
								10	595	44	44	575		57.5
								11	12.73	43	43	589		53.5
								12	15.41	41	41	618		51.5
								13	17.36	41	41	618		47.5
								14	40.09	41	41	618		44.1
								15	40.23	41	41	618		41.2
								16	44.23	41	41	618		38.6

Configuration: 18 Stations

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output / Shift	R^2	Part/ Person
Sawyer	30	original	18	14045	2.08	2.08	2.08	2	10.35	172	172	147	0.86	73.5
								3	150.95	118	118	215		71.7
								4	978.95	88	88	347		86.8
								5	3846.95	73	73	408		81.6
								6	55195.5	62	62	452		75.3
								7	19470	56	56	539		77
								8	83785	47	47	575		71.9
								9	127.22	44	44	618		68.7
								10	528	41	41	649		64.9
								11	33.02	39	39	649		59
								12	156.58	39	39	649		54.1
								13	949.73	38	38	666		51.2
								14	48.61	33	33	767		54.8
								15	137.21	31	31	817		54.5
								16	484	28	28	904		56.5
								17	676	28	28	904		53.2
Gunther	35	1*2	18		1.81	1.81	1.81	2	11.67	270	270	94	0.94	47
								3	174.95	180	180	141		47
								4	567	137	137	185		46.3
								5	723.31	112	112	226		45.2
								6	7019	93	93	272		45.3
								7	50308	87	87	291		41.6
								8	55380	72	72	352		44
								9	19208	64	64	396		44
								10	129.53	62	62	408		40.8
								11	527.16	62	62	408		37.1
								12	1798	56	56	452		37.7
								13	27.47	56	56	452		34.8
								14	55.95	54	54	469		33.5
								15	236.84	50	50	506		33.7
								16	479.39	40	40	633		39.6
								17	784.44	40	40	633		37.2
Warnecke	58	1*2	18	6137.6	4.63	3.04	n/a	2	11.86	802	802	32	0.93	16
								3	162	545	545	46		15.3
								4	412.52	412	412	61		15.3
								5	747.56	334	334	76		15.2
								6	6625	284	284	89		14.8
								7	14287	263	263	96		13.7
								8	44268	209	209	121		15.1
								9	57184	196	196	129		14.3
								10	1142.3	192	192	132		13.2
								11	5913	182	182	139		12.6
								12	13364	175	175	145		12.1
								13	34.97	172	172	147		11.3
								14	66.16	164	164	154		11
								15	112.53	154	154	164		10.9
								16	429.95	123	123	206		12.9
								17	708.04	123	123	206		12.1

Configuration: 18 Stations- continued

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output / Shift	R^2	Part/ Person
Tonge	70	1*4	18	7380	5.4	0.5	n/a	2	14.84	1814	1814	14	0.95	7
				77528	5.4	0.5	n/a	3	146.89	1239	1239	20		6.7
								4	549.11	926	926	27		6.8
								5	1086	762	762	33		6.6
								6	8768	644	644	39		6.5
								7	4311.66	586	586	43		6.1
								8	24818	485	485	52		6.5
								9	10424	435	435	58		6.4
								10	5463	435	435	58		5.8
								11	41.27	424	424	60		5.5
								12	81.2	394	394	64		5.3
								13	157.59	392	392	65		5
								14	67.02	362	362	70		5
								15	136.42	334	334	76		5.1
								16	772.55	280	280	90		5.6
								17	826.53	280	280	90		5.3
Cell 842	79	1*2	18	246357	13.32	3.7	n/a	2	10.5	194	194	131	0.71	35.4
								3	131.3	133	133	190		63.3
								4	342.72	99	99	256		64
								5	86.89	82	82	309		61.8
								6	675.66	71	71	357		59.5
								7	2515.97	60	60	422		60.3
								8	224.5	58	58	437		54.6
								9	613.91	46	46	550		61.1
								10	23.95	45	45	563		56.3
								11	23.52	41	41	618		56.2
								12	19.03	41	41	618		51.5
								13	49.78	41	41	618		47.5
								14	73.92	41	41	618		44.1
								15	89.72	41	41	618		41.2
								16	522.75	41	41	618		38.6
								17	764.21	41	41	618		36.4

Configuration: 19 Stations

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output / Shift	R^2	Part/ Person
Sawyer	30	original	19	273.34	5.56	5.56	5.56	2	19.31	175	175	145	0.98	72.5
								3	327.98	118	118	215		71.7
								4	860.08	89	89	284		71
								5	15308	73	73	347		69.4
								6	11540	61	61	415		69.2
								7	27720	56	56	452		64.6
								8	212301	50	50	506		63.3
								9	63216.6	45	45	563		62.6
								10	40759	40	40	633		63.3
								11	239.73	40	40	633		57.5
								12	1709	35	35	723		60.3
								13	63.33	34	34	745		57.3
								14	266.98	32	32	791		56.5
								15	610.88	29	29	873		58.2
								16	937.42	28	28	904		56.5
								17		25		1013		59.6
								18		25		1013		56.3
Gunther	35	1*2	19	1529	15.85	13.12	n/a	2	21.91	264	264	96	0.58	48
								3	92.52	177	177	143		47.7
								4	1260.39	138	138	183		45.8
								5	1749	113	113	224		44.8
								6	24056	93	93	272		45.3
								7	15127	86	86	294		42
								8	6152	79	79	321		40.1
								9	1058.58	67	67	378		42
								10	66.47	58	58	437		43.7
								11	33.42	58	58	437		39.7
								12	30.97	58	58	437		36.4
								13	83.67	58	58	437		33.6
								14	507.27	58	58	437		31.2
								15	576.28	58	58	437		29.1
								16	1054	58	58	437		27.3
								17		58	58	437		25.7
								18		58	58	437		24.3
Warnecke	58	1*2	19	47.54	7.08	1.88	n/a	2	16.36	816	816	31	0.73	15.5
								3	348.92	540	540	47		15.7
								4	1485	422	422	60		15
								5	10580	332	332	76		15.2
								6	14153	285	285	89		14.8
								7	14437	240	240	106		15.1
								8	101584	220	220	115		14.4
								9	117.94	207	207	122		13.6
								10	155165	176	176	144		14.4
								11	55.06	166	166	153		13.9
								12	120.56	158	158	160		13.3
								13	59.53	158	158	160		12.3
								14	319.41	158	158	160		11.4
								15	581.88	158	158	160		10.7
								16	944.05	158	158	160		10
								17		158	158	160		9.4
								18		158	158	160		8.9

Configuration: 19 Stations- continued

DATA DESCRIPTION				STAGE 1 RESULTS				STAGE 2 RESULTS						
Name	T	L[i,j]	M	Solving Time (sec)	Best Bound	Lower Bound	Optimal Solution	N	Solving Time (sec)	Best Solution	Optimal Solution	Output / Shift	R^2	Part/ Person
Tonge	70	1*4	19	650	65.6	0.5	n/a	2	16.28	1801	1801	14	0.67	7
								3	334	1203	1203	21		7
								4	1258.44	922	922	27		6.8
								5	9911	743	743	34		6.8
								6	15366	630	630	40		6.7
								7	34614	566	566	45		6.4
								8	32435	484	484	52		6.5
								9	10204	450	450	56		6.2
								10	3506	393	393	64		6.4
								11	743.88	369	369	69		6.3
								12	106.75	367	367	69		5.8
								13	128.95	367	367	69		5.3
								14	84.28	367	367	69		4.9
								15	517.7	367	367	69		4.6
								16		367	367	69		4.3
								17		367	367	69		4.1
								18		367	367	69		3.8
Cell 842	79	1*2	19	12816	5.7	4.84	n/a	2	14.83	194	194	131	0.82	27.1
								3	307.69	134	134	189		63
								4	716.51	98	98	258		64.5
								5	1153.87	82	82	309		61.8
								6	1423	68	68	372		62
								7	11386	59	59	429		61.3
								8	20824	54	54	469		58.6
								9	2303	47	47	539		59.9
								10	17209	45	45	563		56.3
								11	28514	41	41	618		56.2
								12	45.34	40	40	633		52.8
								13	207	39	39	649		49.9
								14	631.91	38	38	666		47.6
								15	657	36	36	703		46.9
								16	1226.75	36	36	703		43.9
								17		36		703		41.4
								18		36		703		39.1

Appendix E: Percentages by Stations

These percentages are used in data collection. The percentages are computed from multiple iteration of one selected data set. Iteration with the highest R^2 is selected to compute the percentages. The following list demonstrates the R^2 value, stations times and percentages based on the achieved station times.

Exp #	M	R^2	STATION TIMES	%
18.5	5	0.95	stationtime[1] = 42.0000 stationtime[2] = 20.0000 stationtime[3] = 45.0000 stationtime[4] = 22.0000 stationtime[5] = 56.0000	22.70% 10.81% 24.32% 11.89% 30.27% **100.00%**
6.3	6	0.97	stationtime[1] = 11.0000 stationtime[2] = 17.0000 stationtime[3] = 17.0000 stationtime[4] = 17.0000 stationtime[5] = 3.0000 stationtime[6] = 10.0000	14.67% 22.67% 22.67% 22.67% 4.00% 13.33% **100.00%**
25.2	7	0.96	stationtime[1] = 26.0000 stationtime[2] = 13.0000 stationtime[3] = 11.0000 stationtime[4] = 11.0000 stationtime[5] = 22.0000 stationtime[6] = 22.0000 stationtime[7] = 25.0000	20.00% 10.00% 8.46% 8.46% 16.92% 16.92% 19.23% **100.00%**

Exp #	M	R^2	STATION TIMES	%
16.2	8	0.96	stationtime[1] = 8.0000	17.39%
			stationtime[2] = 5.0000	10.87%
			stationtime[3] = 8.0000	17.39%
			stationtime[4] = 8.0000	17.39%
			stationtime[5] = 3.0000	6.52%
			stationtime[6] = 5.0000	10.87%
			stationtime[7] = 5.0000	10.87%
			stationtime[8] = 4.0000	8.70%
				100.00%
27.2	9	0.92	stationtime[1] = 21.0000	16.15%
			stationtime[2] = 9.0000	6.92%
			stationtime[3] = 9.0000	6.92%
			stationtime[4] = 8.0000	6.15%
			stationtime[5] = 10.0000	7.69%
			stationtime[6] = 16.0000	12.31%
			stationtime[7] = 10.0000	7.69%
			stationtime[8] = 24.0000	18.46%
			stationtime[9] = 23.0000	17.69%
				100.00%
28.1	10	0.96	stationtime[1] = 12.0000	9.23%
			stationtime[2] = 14.0000	10.77%
			stationtime[3] = 17.0000	13.08%
			stationtime[4] = 11.0000	8.46%
			stationtime[5] = 9.0000	6.92%
			stationtime[6] = 9.0000	6.92%
			stationtime[7] = 10.0000	7.69%
			stationtime[8] = 15.0000	11.54%
			stationtime[9] = 17.0000	13.08%
			stationtime[10] = 16.0000	12.31%
				100.00%

Exp #	M	R^2	STATION TIMES	%
cell842	11	0.98	stationtime[1] = 42.0000 stationtime[2] = 41.0000 stationtime[3] = 25.0000 stationtime[4] = 39.0000 stationtime[5] = 21.0000 stationtime[6] = 45.0000 stationtime[7] = 16.0000 stationtime[8] = 45.0000 stationtime[9] = 37.0000 stationtime[10] = 16.0000 stationtime[11] = 19.0000	12.14% 11.85% 7.23% 11.27% 6.07% 13.01% 4.62% 13.01% 10.69% 4.62% 5.49% **100.00%**
30.1	12	0.95	stationtime[1] = 12.0000 stationtime[2] = 4.0000 stationtime[3] = 14.0000 stationtime[4] = 9.0000 stationtime[5] = 11.0000 stationtime[6] = 7.0000 stationtime[7] = 11.0000 stationtime[8] = 10.0000 stationtime[9] = 13.0000 stationtime[10] = 14.0000 stationtime[11] = 13.0000 stationtime[12] = 12.0000	9.23% 3.08% 10.77% 6.92% 8.46% 5.38% 8.46% 7.69% 10.00% 10.77% 10.00% 9.23% **100.00%**
31.1	13	0.98	stationtime[1] = 12.0000 stationtime[2] = 14.0000 stationtime[3] = 4.0000 stationtime[4] = 7.0000 stationtime[5] = 9.0000 stationtime[6] = 8.0000 stationtime[7] = 4.0000 stationtime[8] = 12.0000 stationtime[9] = 10.0000 stationtime[10] = 14.0000 stationtime[11] = 8.0000 stationtime[12] = 13.0000 stationtime[13] = 15.0000	9.23% 10.77% 3.08% 5.38% 6.92% 6.15% 3.08% 9.23% 7.69% 10.77% 6.15% 10.00% 11.54% **100.00%**

Exp #	M	R^2	STATION TIMES	%
32	14	0.91	stationtime[1] = 12.0000	9.23%
			stationtime[2] = 9.0000	6.92%
			stationtime[3] = 9.0000	6.92%
			stationtime[4] = 9.0000	6.92%
			stationtime[5] = 8.0000	6.15%
			stationtime[6] = 7.0000	5.38%
			stationtime[7] = 12.0000	9.23%
			stationtime[8] = 10.0000	7.69%
			stationtime[9] = 9.0000	6.92%
			stationtime[10] = 8.0000	6.15%
			stationtime[11] = 10.0000	7.69%
			stationtime[12] = 13.0000	10.00%
			stationtime[13] = 7.0000	5.38%
			stationtime[14] = 7.0000	5.38%
				100.00%
2	15	0.94	stationtime[1] = 32.0000	6.63%
			stationtime[2] = 30.0000	6.21%
			stationtime[3] = 22.0000	4.55%
			stationtime[4] = 28.0000	5.80%
			stationtime[5] = 29.0000	6.00%
			stationtime[6] = 37.0000	7.66%
			stationtime[7] = 41.0000	8.49%
			stationtime[8] = 40.0000	8.28%
			stationtime[9] = 31.0000	6.42%
			stationtime[10] = 27.0000	5.59%
			stationtime[11] = 24.0000	4.97%
			stationtime[12] = 35.0000	7.25%
			stationtime[13] = 23.0000	4.76%
			stationtime[14] = 42.0000	8.70%
			stationtime[15] = 42.0000	8.70%
				100.00%

Exp #	M	R^2	STATION TIMES	%
6	16	0.91	stationtime[1] = 138.0000	9.21%
			stationtime[2] = 101.0000	6.74%
			stationtime[3] = 56.0000	3.74%
			stationtime[4] = 147.0000	9.81%
			stationtime[5] = 135.0000	9.01%
			stationtime[6] = 73.0000	4.87%
			stationtime[7] = 60.0000	4.00%
			stationtime[8] = 78.0000	5.20%
			stationtime[9] = 75.0000	5.00%
			stationtime[10] = 75.0000	5.00%
			stationtime[11] = 64.0000	4.27%
			stationtime[12] = 153.0000	10.21%
			stationtime[13] = 89.0000	5.94%
			stationtime[14] = 120.0000	8.01%
			stationtime[15] = 60.0000	4.00%
			stationtime[16] = 75.0000	5.00%
				100.00%
35	17	0.96	stationtime[1] = 12.0000	9.23%
			stationtime[2] = 9.0000	6.92%
			stationtime[3] = 5.0000	3.85%
			stationtime[4] = 9.0000	6.92%
			stationtime[5] = 8.0000	6.15%
			stationtime[6] = 8.0000	6.15%
			stationtime[7] = 7.0000	5.38%
			stationtime[8] = 5.0000	3.85%
			stationtime[9] = 9.0000	6.92%
			stationtime[10] = 10.0000	7.69%
			stationtime[11] = 5.0000	3.85%
			stationtime[12] = 7.0000	5.38%
			stationtime[13] = 5.0000	3.85%
			stationtime[14] = 5.0000	3.85%
			stationtime[15] = 6.0000	4.62%
			stationtime[16] = 13.0000	10.00%
			stationtime[17] = 7.0000	5.38%
				100.00%

Exp #	M	R^2	STATION TIMES	%
2	18	0.94	stationtime[1] = 31.0000	6.42%
			stationtime[2] = 30.0000	6.21%
			stationtime[3] = 20.0000	4.14%
			stationtime[4] = 30.0000	6.21%
			stationtime[5] = 25.0000	5.18%
			stationtime[6] = 29.0000	6.00%
			stationtime[7] = 27.0000	5.59%
			stationtime[8] = 22.0000	4.55%
			stationtime[9] = 28.0000	5.80%
			stationtime[10] = 29.0000	6.00%
			stationtime[11] = 32.0000	6.63%
			stationtime[12] = 32.0000	6.63%
			stationtime[13] = 28.0000	5.80%
			stationtime[14] = 23.0000	4.76%
			stationtime[15] = 40.0000	8.28%
			stationtime[16] = 11.0000	2.28%
			stationtime[17] = 40.0000	8.28%
			stationtime[18] = 6.0000	1.24%
				100.00%
1	19	0.93	stationtime[1] = 7.0000	5.60%
			stationtime[2] = 9.0000	7.20%
			stationtime[3] = 5.0000	4.00%
			stationtime[4] = 9.0000	7.20%
			stationtime[5] = 4.0000	3.20%
			stationtime[6] = 8.0000	6.40%
			stationtime[7] = 7.0000	5.60%
			stationtime[8] = 4.0000	3.20%
			stationtime[9] = 6.0000	4.80%
			stationtime[10] = 5.0000	4.00%
			stationtime[11] = 13.0000	10.40%
			stationtime[12] = 5.0000	4.00%
			stationtime[13] = 6.0000	4.80%
			stationtime[14] = 6.0000	4.80%
			stationtime[15] = 5.0000	4.00%
			stationtime[16] = 7.0000	5.60%
			stationtime[17] = 8.0000	6.40%
			stationtime[18] = 6.0000	4.80%
			stationtime[19] = 5.0000	4.00%
				100.00%

10 GLOSSARY

Deterministic Model: A mathematical model which contains no random (stochastic) components; consequently, each component and input is determined exactly. (dssresources.com/glossary/dssglossary1999.html)

Stochastic Model: It is written in the form of a mathematical program extended to a parameter space whose values are random variables (generally with known distribution function). (http://glossary.computing.society.informs.org/index.php?page=S.html).

Integer Programming (IP): The variables are required to be integer-valued. (http://glossary.computing.society.informs.org/index.php?page=I.html)

Mixed Integer Programming (MIP): Some of the variables are required to be integer-valued. (http://glossary.computing.society.informs.org/second.php?page=M.html#MIP)

NP-Hard: It is a problem type that cannot be solved in polynomial time. An optimization problem that relies upon the solution of an NP-complete problem. In that sense, NP-hard problems are at least as hard as NP-complete problems.

http://glossary.computing.society.informs.org/index.php?page=N.html#NP-hard)

NP-Complete: Problems are divided into two categories: those for which there exists an algorithm to solve it with polynomial time complexity, and those for which there is no such algorithm. We denote the former class of problems by P. There are problems for which no known algorithm exists that solves it in polynomial time, but there is also no proof that no such algorithm exists. Among these problems that are not known to be in P (or in ~P), there is a subclass of problems known as *NP-complete*: those for which either all are solvable in polynomial time, or none are. Formally, a problem is NP if there exists an algorithm with

polynomial time complexity that can certify a solution. (http://glossary.computing.society.informs.org/index.php?page=N.html#NP-complete)

Heuristic: In mathematical programming, this usually means a procedure that seeks an optimal solution but does not guarantee it will find one, even if one exists. (http://glossary.computing.society.informs.org/index.php?page=H.html)

Search Tree: The tree formed by a branch and bound algorithm strategy. It is a tree because at each (forward) branching step the problem is partitioned into a disjunction. A common one is to dichotomize the value of some variable, x <=v or x >=v+1. This creates two nodes from the parent: (http://glossary.computing.society.informs.org/index.php?page=S.html)

```
              [parent node]
                  / \
          x <= v /   \ x >= v+1
                /     \
      [left child]   [right child]
```

Branch and Bound: A search strategy in which the `lowest cost' node in the agenda is always considered first. This strategy is guaranteed to find the lowest cost solution if more than one solution to a problem exists. (www.informatics.susx.ac.uk/books/computers-and-thought/gloss/node1.html)

Genetic Algorithm: A population containing a number of trial solutions each of which is evaluated (to yield a fitness) and a new generation is created from the better of them. The

process is continued through a number of generations with the aim that the population should evolve to contain an acceptable solution. (www.cs.bham.ac.uk/~wbl/thesis.glossary.html)