12-11-2014

Implementation of Cross-training for Multi-shift Worker Allocation

Aditya Gandhi

Rochester Institute of Technology

Implementation of Cross-training for Multi-shift Worker Allocation

A Thesis

Submitted in partial fulfillment of the

requirements for the degree of

Master of Science in Industrial Engineering

in the

Department of Industrial & Systems Engineering

Kate Gleason College of Engineering

by

Aditya Gandhi

December 11, 2014

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 Student's Name

has been examined and approved by the

thesis committee as satisfactory for the

thesis requirement for the

Master of Science degree

Approved by:

____

Dr. Scott E. Grasman, Thesis Advisor

____

Dr. Michael R. Hewitt

ACKNOWLEDGEMENTS:

I owe a great debt of gratitude to the following individuals for their assistance and motivation.

First and foremost, I want to thank Dr. Scott Grasman for giving me an opportunity to work on this project. In particular, I would like thank him for his never-ending patience, support and guidance throughout the duration of my master's degree at RIT. I have learnt numerous valuable lessons about the qualities of a successful leader from his words and actions. I would also like to thank Dr. Mike Hewitt for teaching me linear, non-linear and integer programming. I admire his approach towards operations research, and towards his classes, which were a blast for all involved.

I would also like to thank Dr. Shrikant Jarugumilli; without whom this thesis would not have been possible. Further, I would like to thank Naiping Keng for his guidance and industry-oriented feedback, which was invaluable to this research work. I also owe a debt of gratitude to Dr. Ruben Proano for his introductory course in operations research. His passion for operations research was fundamental in driving me to work in this area.

I would like to thank my friends Srinath, Raghava, Sivakumar, Vasudev, Arun and Bala for supporting me throughout the completion of this thesis.

Last, but not the least, I would like to dedicate this thesis to my mother, Mamta Gandhi, and my father, Deepak Gandhi, for their blessings and their unwavering faith in my abilities.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

In workforce allocation, gaps between workers available and workers needed at various operations result in production delays and a loss of profitability for the manufacturer. These gaps can be reduced by overtime assignments of workers from other shifts. However, for a multiple shift planning horizon, a mix of cross-training of workers over different tasks along with overtime assignments may be a good strategy. This work develops an industry-motivated cross-training framework that identifies workers and operations for normal, overtime, and training assignments. A mixed integer programming model that integrates all three assignment tasks is formulated and solved. The production scenario consists of skill level based qualifications for workers that need to be assigned to operations in every shift. Factory floor conditions such as limits on worker levels at specific operations, scheduling restrictions and worker training protocols are also considered. The data taken into account includes parameters such as man-machine ratio, tool count, and limits on skill qualifications for workers. The output of the model provides a cross-training schedule and an assignment schedule that can be used by floor managers on a shift-by-shift basis. The MIP model is implemented in C# using the .NET framework and the IBM ILOG CPLEX Optimizer.

1. INTRODUCTION

Due to changing production demands, large organizations spend a significant amount of resources to determine daily worker schedules. In a complex manufacturing environment, that is both high-performance and labor-intensive, manual scheduling methods are inefficient and time-consuming. Factors such as the use of state-of-the-art machinery, workforce efficiency and production cost are important parameters that need to be taken into consideration while designing worker schedules. Furthermore, intense competition existing in the industry necessitates a highly optimized manufacturing process. As shown in Figure 1.1, the advantages of optimized worker schedules are decreased expenses, increased customer retention and reduced pressure on floor managers.

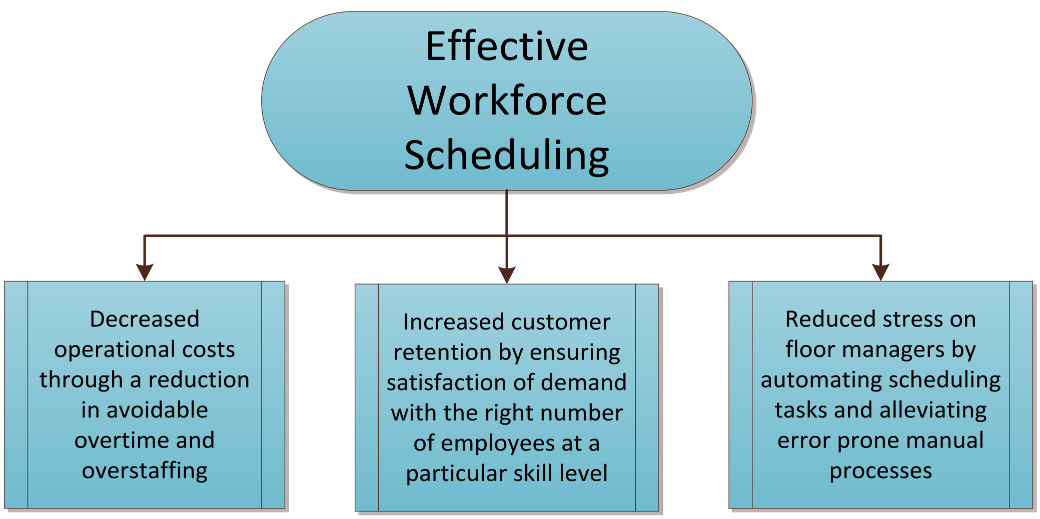

Figure 1.1: Advantages of Effective Workforce Scheduling

For effective scheduling, worker allocation models are popular. As mentioned in Celayix (2013), the benefits of these models range from lower operating costs, increased customer satisfaction, and the availability of better analysis tools for management. Worker allocation models assign workers to operations over a pre-determined duration of time. They may also take

into account factors such as allocation of workers with lower qualifications, personal preference of workers, and pre-allotment of specific workers to specific tasks. Also, parameters such as the operational cost and the efficiency of workers need to be considered before assignment of the worker. Sometimes, in labor-intensive tasks, multiple workers are needed for machines and/or overtime, along with training of workers. Workforce planning, in general, also involves strategic decisions such as the number of workers which are to be hired, trained or moved to other shifts for a larger planning horizon.

Worker allocation models must be developed considering different factors, such as type of work, planning periods, and demand patterns. A few of these factors are given as follows.

- Planning Horizon: We consider three planning horizons; single shift planning, multiple shift planning and strategic planning. Strategic planning models are generally long-term, whereas single-shift and multiple-shift planning are short-term.
- Demand Type: The demand can be expressed in a number of ways. For example, it can be the number of shifts of production needed, the number of workers needed at each operation, or the number of units produced in one shift.
- Shift Structure: Proper management of the labor force is a daunting task. In particular, assigning workers to different tasks involving regular work, overtime shifts and training procedures is a challenge.
- Other Factory Rules: These are rules that are specific to the factory floor under consideration. Some examples of factory rules are limits on cross-training/overtime per shift, step-by-step qualification training and number of worker allocations per shift.

Table 1.1 shows the some examples of planning horizons, shift structure and the demand type associated with workforce allocation models.

Factors Involved in Designing Workforce Allocation Models	
Planning Horizon	Long-term, Short-term
Shift Structure	5 days a week-8hr shifts, 4 days a week followed by 3 days a week (12 hrs/day)
Demand Type	Number of production shifts needed, Amount of produced units needed, Number of workers needed
Other Factory Rules	Training shifts needed, Max overtime/training per shift, training/overtime limits

Table 1.1: Factors considered while designing worker allocation models

In a competitive industry, it is imperative to achieve production targets on time, and for this purpose, workers need to be assigned to operations optimally, such that the demand at every operation is met. A mismatch between the available workers and the workers needed causes workforce gaps that result in a loss of profit and goodwill for the manufacturer. Similarly, each company has its own set of rules that need to be followed throughout the production process. These rules may be related to machinery, workforce, schedules, and skill qualification. This work is an industry-motivated project that aims to schedule regular work, overtime work and cross-training such that workforce gaps are minimized. Also, factory rules are given high importance, and we observe the effects that these have on individual worker-task allocations. Skill level considerations also form an important component of the mathematical model, along with standard factory floor limitations.

The rest of this thesis is organized as follows. Section 2 surveys the literature related to worker allocation and cross-training models and also highlights the research gap. Section 3 describes the problem and describes the industry-based system under consideration. This section also contains the mathematical notation, the multi-shift cross-training mathematical model, the

structure of cost parameters and the implementation architecture. Section 4 contains a numerical example of the mathematical model developed in Section 3. Section 5 presents the summary of test cases and experimentation. Also, we discuss the solving time associated with the model and its overall impact. Section 6 concludes this thesis and proposes future extensions to this research work.

2. LITERATURE REVIEW

The literature related to this research can be divided into two parts, worker allocation models, and cross-training models. The discussion of worker allocation models is presented in Section 2.1 and the discussion of cross-training models is presented in Section 2.2. Section 2.3 highlights the research gap that this work aims to fill.

2.1 Worker Allocation Models

Existing literature in the area of worker allocation can be divided broadly into three areas: single-shift models, multi-shift models and strategic models. Although we develop a multi-shift model in this thesis, we have reviewed relevant literature related to single shift models too, since they are building blocks for multi-shift allocation models. We also present an overview of related strategic models. Review papers such as Wang (2005) and Ernst *et al* (2004) perform an overview of the different personnel scheduling papers. Ernst *et al* (2004) divides allocation models according to various parameters such as demand modeling, days off, shift scheduling and task/shift assignment. Wang (2005) provides an overview of the different application areas such as healthcare, emergency services, airlines and hotels. A discussion of the individual papers directly relevant to this research work is provided in the sections below.

2.1.1 Single-Shift Models

Suer (1996) designs an optimal assignment policy for a single period allocation problem by designing a mixed integer program to determine the operator allotment for a variety of demand scenarios. The results show that before implementation of this methodology, all alternative operator skill levels should be considered. Aykin (1996) develops a single shift model that considers breaks in the duration of a work day. It was concluded from the paper that the

model is applicable for both continuous production and cyclic production. Brusco and Johns (1998) develop an integer programming model that minimizes the workforce staffing costs for a single work shift. Different cross-training configurations are evaluated and the model is used to analyze these configurations. The results indicate that asymmetric training configurations allow better operator-skill relationships. Campbell (1999) formulates a single shift assignment problem that considered various skill levels across different departments of an organization. The paper concludes that the approach used is better than the Lagrangian approach used by other authors. Askin and Huang (2001) develop two MILP formulations that determine allocation of workers with different costs for training on different skill levels. The output of the first model assigns workers to cells and gives the assignment of tasks for the worker at that particular cell. The second model determines the individual training schedule. Norman *et al* (2002) considers human skills such as leadership and decision making while assigning workers. A mixed integer program is developed to maximize the profit which is expressed in the form of a function of training cost and throughput. The results indicate that better results occur when only technical skills are considered. Wu and Fu (2005) present a linear programming model for operator staffing and assignment in a foundry fab. Their model minimizes the cost of staffing and chooses a staffing position which is assigned to an operator.

Kuo and Yang (2006) solve an operator allocation problem using mixed integer programming where the formulation assumes that there is no difference between the skill levels of operators. It was concluded that this assumption does not hold true when there is significant difference in between skill levels. Jarugumilli *et al* (2010) consider rules such as the lower limit on the number of workers at a machine group, the desired mix of workers with a certain skill level and the operational preference at each of the machine groups. Also, the emphasis is on

knowing the most effective allocation of resources despite not always being able to meet the worker requirements. A two-phase goal programming formulation is used which minimizes the cost of assigning workers and the slack/surplus of workers at each operation. The cost structure is specified for the parameters used in the optimization. The results indicate that operations and workers for cross-training can be identified using the model.

2.1.2 Multi-shift Models

In Burns *et al* (1998), an algorithm for scheduling workforce over 4-day or 3-day work week is developed. Employees work 4-days a week, have a fixed number of weekends off and work a maximum of 5 days in a row without a break. The results of the model calculate the minimum number of workers required to satisfy the demand. Azmat and Widmer (2004), develop a three-step process for workforce planning and scheduling. First, the minimum workforce that can satisfy the particular demand is computed. Second, the number of overtime hours and days off for the workforce is determined. And the last step is the actual allocation of the number of work days per week to each worker. The results show that the algorithm is a quick method to allocate workers. Laporte and Pesant (2004) develop a constraint programming algorithm that can handle a wide variety of constraints during a 24 hour, 7 days a week manufacturing facility. The algorithm is also validated with computational results which support their considerations of different shift structures, days off and breaks. Bhatnagar (2007) develops a linear programming model for optimal allocation of workers for companies such as computer manufacturers and cellphone makers. The results indicate that the cost of allocation increases with increase in demand variability.

Suer and Tummaluri (2008) expand their single-shift model (Suer, 1996) by considering a case of multiple periods, operator skill levels and operation times. A heuristic for operator allotment is developed and considers learning/forgetting issues for a worker. Results show that the proposed approaches in operator assignment outperform the classical approach of using standard times. Jaurugmilli (2011) extends his single-shift model (Jarugumilli *et al*, 2010) by linking the single-shift and multi-shift decision making scenarios. A multi-shift model is formulated that makes both single-shift and multi-shift decisions for a two-week planning horizon. The multi-shift decisions satisfy capacity plans and worker availability constraints, while the single-shift decisions satisfy qualification and skill level requirements. The results help workers in identifying operations and workers for cross-training. Also, the model considers partial allocations for any shift during the planning period.

2.1.3 Strategic Models

Strategic models generally consist of a longer time horizon and often have stochastic parameters. For instance, Ahn *et al* (2005) design a Markov decision process that determines the number of workers to hire while minimizing the cost. Hiring and firing of workers is considered along with a random demand scenario and linear costs. A dynamic programming based approach is followed to obtain the optimal staffing levels. Gans and Zhou (2002) also implement a Markov Decision Processes to determine an optimal policy to determine the number of workers to hire. The results indicate that the when learning is high, a state dependent policy outperforms a policy dependent on the number of workers in a system. Huang and Song (2007) use a successive convex approximation method to model a multi-stage stochastic model. Numerical experiments are conducted that suggest that the solutions obtained are near optimal.

Techawiboonwong and Yenradee (2003) consider an aggregate production planning model using mixed integer programming where the worker resources can be transferred along production lines. The output provides benefits to managing the available production capacity by using workforce transfers among the lines. Fowler *et al* (2007) use MIP to determine different staffing decisions such as hiring, firing and cross-training to minimize workforce related costs over multiple periods. Kulkarni et al (2013) uses a scenario-based approach to solve a workforce planning problem formulated as a two-stage stochastic recourse program. Considerations include fluctuating demand over a long planning horizon, business and labor rules, e.g., hiring, firing, overtime, cross-training, and shift swapping. The results indicate that the cost of workforce formulation can be reduced by using the recourse problem.

2.2 Cross-training Models

Early literature consists of industry implementations of cross-training and its benefits to individual workers. Hackman and Oldham (1980) talk about the positive effect that cross-training has on a worker's quality of life and motivation. Jordan and Graves (1995) provide an actual guideline for management on the methods used to implement cross-training policies in a manufacturing environment. In Brandt (1997), an example of a cross-training policy was CTRAIN, used in semiconductor manufacturing at IBM. The results showed an increase in teamwork and reduced the pressure felt by workers to adapt to ever-changing demands. Another work, Bailey (1998), talks about the importance of forming cross-functional teams. This work also compares different team structures (with a particular level of cross-training) and their respective labor productivities. The results indicate that continuous improvement teams are

associated with the highest productivity. Slomp and Molleman (2002) distinguish four cross-training policies depending on their effect on important performance metrics such as load of the bottleneck workers and the number of new qualifications used. It is shown that a worker oriented cross-training policy, which spreads functions evenly among employees, performs well.

Hopp and Van Oyen (2003) prove that worker coordination, team structure and training efficiency are parameters essential in evaluating workforce agility. Also, workforce agility is supported by cross-training and leads to increased task responsiveness, improved quality of products, and an increased capability to produce a wider range of products. Slomp *et al* (2003) include cross-training into an integer programming formulation that minimizes the cost associated with training. Also, the model is tested with different values of redundancy of machines and multi-functionality of workers. The purpose of the formulation is to achieve equal distribution of workload among the workers. Once the amount of cross-training required has been determined, a strategy needs is developed to assign workers to training. Bokhorst, et al (2004) extends on Slomp et al (2003) using the concept of skill chaining to develop different cross-training configurations. Hopp et al (2004), goes in a different direction from Hopp and Van Oyen (2003), and analyzes the benefits of cross-training of workers for serial production lines. Bokhorst (2010) measures the impact of the work-in-process on the use of cross-training skills received by a worker.

Table 2.2.1 shows the comparison of selected worker allocation and cross-training works.

Literature	Single-shift /Multi-shift/ Strategic	Overtime Allocations	Cross-training	Multiskilling Considered	Non-stop Production	Fractional Allocation
Askin and Huang (2001)	Single-shift		√	√		√
Aykin (1996)	Single-shift				√	√
Azmat and Widmer (2001)	Single-shift	√				√
Bokhorst *et al* (2004)	Single-shift		√	√		
Brusco and Johns (1998)	Single-shift		√	√		
Campbell (1999)	Single-shift		√	√		
Kuo and Yang (2006)	Single-shift					√
Norman *et al* (2002)	Single-shift		√	√		
Slomp and Molleman (2002)	Single-shift		√	√		
Bhatnagar *et al* (2007)	Multi-shift	√		√		√
Fowler *et al* (2008)	Multi-shift		√	√		
Gomar *et al* (2002)	Multi-shift			√		
Jarugumilli (2011)	Multi-shift	√		√	√	√
Subramanian and An (2008)	Multi-shift	√	√	√		√
Ahn *et al* (2005)	Strategic	√		√		
Gans and Zhou (2002)	Strategic	√				
Huang and Song (2007)	Strategic	√		√		
Techawiboonwong and Yenradee (2003)	Strategic			√		
Fowler *et al* (2007)	Strategic		√	√		
Kulkarni *et al* (2013)	Strategic	√	√	√	√	

Table 2.2.1: Comparison of selected works of workforce allocation and cross-training

2.3 Research Gap

Figure 2.3.1 describes relevant papers from workforce allocation, cross-training and the intersection of both. From the papers in the intersection, i.e., Brusco and Johns (1998) and Askin and Huang (2001), skill levels are relevant, but both are single-shift models. Brusco and Johns (1998) do not consider overtime considerations or even different shift structures. Askin and

Huang (2001) propose two models developed for only cellular manufacturing systems, and focus on the synergy between the teams instead of minimizing allocation cost.

As mentioned in Jarugumilli (2011) - to satisfy the different demand scenarios, existing workforce capabilities might not be sufficient. This results in workforce gaps that result in unsatisfied demand at the manufacturing floor operations. In previous models such as Bhatnagar (2007), gaps have been minimized by the use of a large number of overtime workers (i.e. workers from other shifts) to satisfy the demand of the current shift. However, the use of overtime workers is a shift-by-shift solution. Over a larger time duration, implementation of cross-training is needed. Hence, this work integrates cross-training procedures along with regular and overtime allocation.

As can be observed from Table 2.2.1, we plan to develop a multi-shift model that integrates regular/ overtime/training allocations, multiskilling, non-stop production and fractional allocation in the same formulation. In addition, this work also considers many specific factory rules for the system under consideration.

Figure 2.3.1: Focus areas of existing literature

3. PROBLEM DESCRIPTION AND FORMULATION

Consider a system that has a fixed number of workers, operations, shifts and skill levels. The problem involves optimally allocating regular workers, assigning overtime, and assigning workers for cross-training such that the cost of all three actions along with the cost of the workforce gaps is minimized.

Depending on the system under consideration, a worker can have a number of skill levels. Lower skill levels involve doing regular work, and advanced skill levels are related to machine maintenance. Worker assignments can be made to only one operation at a time at one skill level, but the worker can be assigned to multiple operations in the same shift.

In case of worker assignment, a worker can work on an operation for some duration of his shift, while in case of cross-training, a worker has to work for the complete duration of the cross-training period. If lower skill level training is taking place on an operation, then that operation can be used by another worker. If higher skill level training is taking place, then that particular operation cannot be used for any other activity. Every operation has a minimum threshold limit for which a worker can be assigned to it

There is a set of specified workers for each shift, known as the regular workers. Overtime workers are regular workers from other shifts who can be contacted if the requirements are not met by regular workers. Different sets of workers work shifts on fixed days. Based on the workers working on that particular day, each shift can be classified as part of a shift type. Workers cannot work in back-to-back shifts, but they can do overtime work in a shift of another shift type.

3.1 System under Consideration

The aim is to minimize the costs associated with regular allocation, overtime allocation, cross-training and workforce gaps. These costs are not monetary costs, but values associated with the assignment of regular work, overtime, cross-training or gaps. The cost parameters have been determined based on industry feedback. The structure of these costs will be explained in Section 3.3.

For the testing and analysis of the model, we have considered a system that includes the different factors associated with the problem. The components of the system under consideration are explained below.

3.1.1 Workers/Shifts/Shift Types

This facility has two non-overlapping daily shifts of 12 hours each. There is a fixed planning horizon of two weeks, i.e., a total of 28 shifts. There are four types of shifts - ST1, ST2, ST3 and ST4. ST1 and ST3 are daytime shifts, while ST2 and ST4 are night time shifts. Workers in ST1 and ST2 work for forty-eight hours in week 1 of the planning period, and for thirty-six hours in week 2 of the planning period. Similarly, workers in ST3 and ST4 work for thirty-six hours in week 1 and forty-eight hours in week 2. Table 3.1.1 shows the shift distribution for the manufacturing facility. (Jarugumilli (2011))

Week1	Sun	Mon	Tue	Wed	Thu	Fri	Sat
Day Shift	ST11 (1)	ST12 (3)	ST13 (5)	ST14 (7)	ST31 (9)	ST32 (11)	ST33 (13)
Night Shift	ST21 (2)	ST22 (4)	ST23 (6)	ST24 (8)	ST41 (10)	ST42 (12)	ST43 (14)
Week2							
Day Shift	ST15 (15)	ST16 (17)	ST17 (19)	ST34 (21)	ST35 (23)	ST36 (25)	ST37 (27)
Night Shift	ST25 (16)	ST26 (18)	ST27 (20)	ST44 (22)	ST45 (24)	ST46 (26)	ST47 (28)

Table 3.1.1: Shift Structure Considered

To explain the notation, ST13 means that it is shift type 1 (ST1) and the third shift in the time horizon. So, the general notation is ST[Shift type][Shift number in shift type]. The numbers in parentheses show the shifts in the order of actual occurrence.

Every shift type contains a set of workers, and apart from overtime situations they work in shifts of that particular shift type itself. For example, let us say that there are 60 workers; and workers 1-15 only work in ST1, 16-30 only work in ST2, 31-45 only work in ST3, and 46-60 only work in ST4. These are called as the regular workers for each shift. ST1 and ST2 workers work day/night shifts on the same day, while ST3 and ST4 workers work day/night shifts on the same day. Workers in ST3 and ST4 can do overtime work only in shifts ST1 and ST2 (since they are on different days) and vice versa (i.e., ST1, ST2 workers can do overtime only in ST3 and ST4).

3.1.2 Skill Levels

A worker can have three skill level qualifications; L1, L2, and L3. L1 is a basic skill level; L2 and L3 are advanced skill levels that include machine repair and maintenance. Every operation needs workers that are qualified on L1, L2 and L3. Workers that are qualified on a

higher skill level (L2 or L3) of a particular operation are capable of doing work at lower skill levels for the same operation. For example, if a worker is qualified on OP7-L3, he can do OP7-L2 and OP7-L1. If he is qualified on OP7-L2, he can do OP7-L1 also. However, if he is qualified on OP7-L1, he cannot do work at OP7-L2 or OP7-L3. Also, to be cross-trained on OP7-L2, a worker needs to be qualified on OP7-L1; and to be cross-trained on OP7-L3, a worker needs to be qualified on OP7-L2.

3.1.3 Operations

Every operation has a fixed requirement, i.e., for workers qualified on L1, L2 and L3 on that operation. Overtime workers (regular workers from other shifts) can be contacted if the demand for workers is not met by the regular workers on any operation. Every operation is unique, and a worker that is qualified on an operation skill level combination, cannot automatically perform work of the same skill level on other operations. Hence, it is essential to determine the number of workers that need to be qualified on any particular operation. The solution to the problem involves optimally (at the lowest cost) assigning workers to operations such that worker requirement at every skill level, on every operation is met. If regular workers are not sufficient then overtime workers from other shifts (not shifts on the same day) are used. The two-week schedule for normal allocation of workers and that for cross-training of workers needs to be developed before the start of each planning period.

Figure 3.1.1 denotes the normal operating procedure of the factory floor. The initial scenario depicts workers slotted into the 4 different shift types, i.e., workers 1-4 in ST1, workers 5-7 in ST2, and so on. The same operations exist for each shift of a shift type, and each operation consists of L1/L2/L3 levels of workers. Once a shift starts, assignments such as regular

allocation of workers, cross-training of workers, and overtime allocations (if needed) from other shift types take place. For example, for ST1, overtime on an operation is obtained from workers in ST3 and/or ST4. ST2 cannot provide overtime workers because it would mean that a worker, if assigned for overtime in this shift, would work for 24 hours non-stop. Similarly, for ST2 overtime is obtained from ST3 and/or ST4 and for ST3 and ST4, overtime is obtained from ST1 and/or ST2. If even after assigning overtime, the demand is not satisfied, then workforce gaps are created.

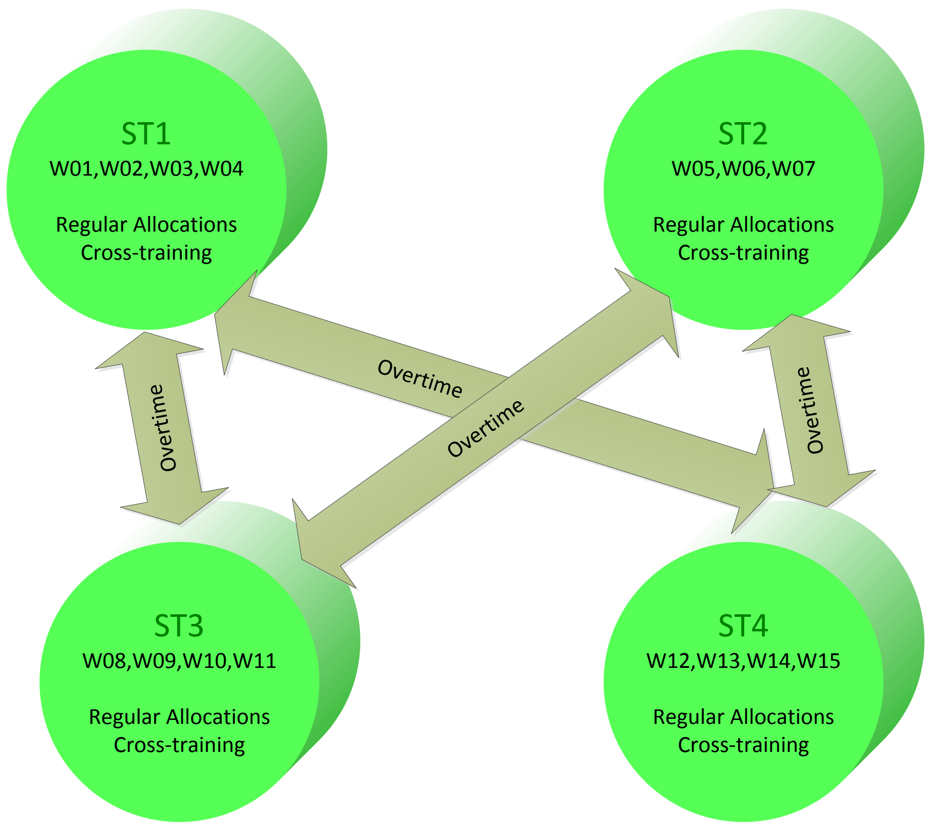

Figure 3.1.1: Normal functioning of Factory Floor

3.2 MULTI-SHIFT CROSS-TRAINING MODEL

To integrate regular, overtime and cross-training assignments, we develop a multi-shift cross-training model formulated as an MIP model. It is a deterministic formulation for a fixed time horizon and the assignment of workers and cross-training schedules are developed for each shift. The regular work requirements and the cross-training requirements are known entities, and the aim is to satisfy these values while minimizing the cost associated with regular allocations, overtime allocations, cross-training assignments and workforce gaps and cross-training gaps. Other factory floor considerations have also been taken into account in the formulation.

3.2.1 Notation

The notation used in the mathematical model is given as follows:

Sets

$set\ I = \{1,2,\ldots..\mathrm{I}\}$ is the set of workers.

$set\ N = \{1,2,\ldots.\mathrm{N}\}$ is the set of operations.

$set\ L = \{1,2,\ldots..\mathrm{L}\}$ is the set of skill levels.

$set\ K = \{1,2,\ldots.\mathrm{K}\}$ is the set of shifts.

$set\ J = \{1,2,..\mathrm{J}\}$ is the set of shift types.

$set\ K_j$ = Set of shifts present in shift type $j \epsilon J$.

$set\ W_j$ = Set of workers present in shift type $j \epsilon J$.

$set\ T_j$ = Set of shifts that can possibly exist for training for in each shift type $j \epsilon J$.

Variables

S_{iln}^{jk} = Binary variable which is 1 if training starts for worker $i \epsilon W_j$, on skill level $l \epsilon L$, operation $n \epsilon N$, during shift type $j \epsilon J$ and shift number $k \epsilon K_j$, and 0 otherwise.

X_{iln}^{jk} = Continuous variable from 0 to 1 which denotes the percentage of the shift duration for worker $i \epsilon W_j$, on skill level $l \epsilon L$, operation $n \epsilon N$, during shift type $j \epsilon J$ and shift number $k \epsilon T_j$.

Y_{iln}^{jk} = Binary variable which is 1 if worker $i \epsilon W_j$ is being trained on skill level $l \epsilon L$, operation $n \epsilon N$, during shift type $j \epsilon J$ and shift number $k \epsilon T_j$, and 0 otherwise.

OT_{ln}^{jk} = Amount of overtime worker-shifts needed for operation $n \epsilon N$, skill level $l \epsilon L$ during shift type $j \epsilon J$ and shift number $k \epsilon K_j$.

G_{ln}^{jk} = Gap in the allocation at operation $n \epsilon N$, on skill level $l \epsilon L$ during shift type $j \epsilon J$ and shift number $k \epsilon K_j$.

E_{iln}^{jk} = Binary variable which is 1 when worker $i \epsilon W_j$ on operation $n \epsilon N$ and skill level $l \epsilon L$ is being trained during shift type $j \epsilon J$ and shift number $k \epsilon K_j$, and the training ends in shift k-1, and 0 otherwise.

SL_{ln}, SP_{ln} = Slack variable and surplus variable which denote the gap in cross-training on operation $n \epsilon N$ and skill level $l \epsilon L$.

Z_{iln}^{jk} = Binary variable that is 1 if worker $i \epsilon W_j$ is assigned to operation $n \epsilon N$, skill level $l \epsilon L$ during shift type $j \epsilon J$ and shift number $k \epsilon K_j$, and 0 otherwise.

TW_n^{jk} = Binary variable that is 1 if training of skill level L2 or L3 is taking place at n ϵN in shift type $j \epsilon J$, shift $k \epsilon K_j$. It is 0 otherwise.

ST_{ln}^{jk} = Binary variable that is 1 if training takes place on operation n, skill level l in shift type $j \epsilon J$, and shift $k \epsilon K_j$, and 0 otherwise.

Parameters

r_{ln}^{j} = Number of worker-shifts needed on operation $n \epsilon N$ and skill level $l \epsilon L$ during shift type $j \epsilon J$.

rc_{ln} = Number of worker-shifts of training needed on operation $n \epsilon N$ and skill level $l \epsilon L$.

$mshift_{ln}$ = Number of shifts of training needed for a worker to be completely trained on operation $n \epsilon N$ and skill level $l \epsilon L$.

mn/mx = Minimum/maximum number of qualifications a worker can have at any moment in the time horizon.

c_{iln} = Cost of assignment for worker $i \epsilon I$, on skill level $l \epsilon L$ and operation $n \epsilon N$.

ct_{iln} = Cost of training worker $i \epsilon I$, on skill level $l \epsilon L$ and operation $n \epsilon N$.

ϵ_{iln}^{jk} = Pre-allocation if training has been carried forward for worker $i \epsilon W_j$, on skill level $l \epsilon L$ and operation $n \epsilon N$, in shift type $j \epsilon J$ and shift number $k \epsilon K_j$.

φ_{iln}^{jk} = Partial allocation for cross-training that is given to the worker $i \epsilon W_j$, on skill level $l \epsilon L$ and operation $n \epsilon N$, in shift type $j \epsilon J$ and shift number $k \epsilon K_j$.

q_{iln} = Value is 1 if worker $i \epsilon W_j$, is qualified at skill level $l \epsilon L$ and operation $n \epsilon N$ at the beginning of the planning period, and 0 otherwise.

ot_{limit} = Limit on overtime for the entire factory.

$\propto_{shift}$ = Limit on training for every shift.

$\beta_{factory}$ = Limit for training for the complete factory.

p_{ln} = Cost of overtime on operation $n \epsilon N$ and skill level $l \epsilon L$.

ps_{ln}= Cost of a gap at operation $n \epsilon N$ on skill level $l \epsilon L$.

pp_{ln}, pn_{ln} = Cost of surplus/slack in cross-training on operation $n \epsilon N$ and skill level $l \epsilon L$.

M = A big number (condition : $M \geq 1/sm$)

sm = Lowest allocation that a worker can take if he is assigned (Eg: If assigned, the worker can take a value of greater than 0.1).

$count_i$ = Number of individual operation-skill level qualifications a worker $i \epsilon W_j$ possesses at the beginning of the shift.

$tl =$ Upper limit on the number of worker than can be trained on an operation in any particular shift.

$wl =$ Maximum number of operations that a worker can be assigned to in any particular shift.

Lim = Number of times a worker can be trained over the planning horizon.

3.2.2 Mathematical Model

The data in the previous section is used to formulate the mathematical model that can be given as follows.

Objective Function

The objective function [1] is a cost-parameter based formula that minimizes the sum of the following:

- Total cost of assignment of regular workers on all the operations over two weeks,
- Total cost of all overtime assignments over the two weeks,
- Penalty costs due to all the operations where workforce gaps exist over two weeks,
- Total cost of cross-training workers over two weeks, and
- Penalty costs associated with not meeting cross-training targets that are specified at the beginning of the two weeks.

$$Minimize \quad \sum_l \sum_i \sum_n \sum_k \sum_j c_{iln} . X_{iln}^{jk} + \sum_l \sum_n \sum_k \sum_j p_{ln} . OT_{ln}^{jk} + \sum_l \sum_n \sum_k \sum_j ps_{ln} . G_{ln}^{jk} +$$

$$\sum_i \sum_l \sum_n \sum_k \sum_j ct_{iln} . Y_{iln}^{jk} + \sum_l \sum_n pp_{ln} . SP_{ln} + \sum_l \sum_n pn_{ln} . SL_{ln} \qquad [1]$$

Constraints

Constraint [2] ensures that the total worker allocation at each operation (including overtime allocations) should satisfy the requirement at that operation in each shift. If it does not, then worker gaps are observed. Similarly, even the cross-training requirements are given. Constraint [3] ensures that the number of workers trained on every operation over the two weeks should be equal to the pre-established targets.

$$\sum_i X_{iln}^{jk} + OT_{lk}^{jk} + G_{ln}^{jk} = r_{ln}^{j} \quad \forall \; l, n, j, k \qquad [2]$$

$$\sum_i \sum_j \sum_k Y_{iln}^{jk} - SP_{ln} + SN_{ln} = rc_{ln} \quad \forall \; l, n \qquad [3]$$

Constraints [4] and [5] work together. If a worker is allocated, he can only be allocated for a shift duration that is greater than or equal the minimum limit, *sm*. Generally, for most test cases, we have assumed this value to be 25% of the total shift duration.

$$Z_{iln}^{jk} \geq X_{iln}^{jk} \quad \forall \; i, l, n, j, k \qquad [4]$$

$$Z_{iln}^{jk} \leq \frac{1}{sm} . \; X_{iln}^{jk} \quad \forall \; i, l, n, j, k \qquad [5]$$

Constraint [6] ensures that the total overtime allocation for the factory does not cross an upper limit. Constraint [7] ensures that a worker cannot be assigned to training while doing work, and vice versa. There are upper limits on the number of workers cross-trained per shift as given in Constraint [8], and the number of workers that can be cross-trained in the planning period as given in Constraint [9].

$$\sum_l \sum_n \sum_j \sum_{k \epsilon K_j} OT_{ln}^{jk} \leq ot\_limit \quad [6]$$

$$\sum_l \sum_n X_{iln}^{jk} + \sum_l \sum_n Y_{iln}^{jk} \leq 1 \qquad \forall \ i \, \epsilon \, W_j, j \, \epsilon \, J, k \, \epsilon \, K_j \quad [7]$$

$$\sum_i \sum_l \sum_n Y_{iln}^{jk} \leq \ \propto_{shift} \qquad \forall \ j \, \epsilon \, J, k \, \epsilon \, K_j \quad [8]$$

$$\sum_l \sum_n \sum_k \sum_j \sum_i Y_{iln}^{jk} \leq \beta_{factory} \quad [9]$$

Constraints [10] and [11] address the fact that if a worker starts, then the starting shift needs to be kept track of)

$$S_{iln}^{jk} = Y_{iln}^{jk} \qquad \forall \ i \, \epsilon \, W_j, l \, \epsilon \, L, n \, \epsilon \, N, j \, \epsilon \, J, \ k = k_{j[1]} \, \epsilon \, K_j \quad [10]$$

$$S_{iln}^{jk} \geq Y_{iln}^{jk} - Y_{iln}^{jk'} \qquad \forall \ i, l, n, j, k > k_{j[1]}, \ k' = Element \ of \ K_j \ before \ k \, \epsilon \, K_j \quad [11]$$

Constraint [12] satisfies the requirement that if a worker is has started training, then he must be trained for the pre-specified number of shifts needed to be qualified at that skill level.

$$mshift_{ln}. \ S_{iln}^{jk} \ \leq \ \sum_{r \geq k}^{r \leq k' = k + mshift_{ln} - 1} Y_{iln}^{jr} \qquad \forall \ \ i \, \epsilon \, W_j, \ l, n, j, k \, \epsilon \, K_j, \ k' \epsilon \, T_j \quad [12]$$

Constraint [13] prevents duplicate training of a particular worker on the same operation-skill level combination.

$$\sum_{k \, \epsilon \, K_j} S_{iln}^{jk} \leq 1 \qquad \forall \quad i, l, n, j \quad [13]$$

The shift in which training for a worker ends is kept track of using Constraints [14] and [15]. Constraint [14] recognizes the ending shift, and if training shifts are more than 1, Constraint [15] ensures that the variable denoting the end of training is 0 in the shifts between the start and end shifts.

$$S_{iln}^{jk} = E_{iln}^{jk\prime} \quad \forall \;\; i, l, n, j, \; k \in K_j \;, \; k' = (mshift_{ln} + k)^{th} \; member \; of \; T_j \qquad [14]$$

Only if $mshift_{ln}$> 1;

$$E_{iln}^{jk} = 0 \quad \forall \quad i, l, n, j, \; k \leq (mshift_{ln})^{th} \; member \; of \; T_j \;\; \&\& \; k \in K_j \qquad [15]$$

Constraints [16] and [17] put lower and upper limits on the number of possible qualifications a worker can have in any shift.

$$\sum_n \sum_l \sum_k S_{iln}^{jk} + count_i \geq \;\; mn \quad \forall \quad i, j \qquad [16]$$

$$\sum_n \sum_l \sum_k S_{iln}^{jk} + count_i \leq \;\; mx \quad \forall \quad i, j \qquad [17]$$

Constraint [18] enforces that a worker can only be trained on skill level L2 of an operation if he is qualified on skill level L1 of that same operation. Similarly, Constraint [19] enforces that condition for L3 and L2.

$$M. \; E_{iln}^{jk} \geq \sum_{k'>k} Y_{il'n}^{jk'} \quad \forall \;\; l = L1,; \; l' = L2; \; n, \; i, \; j \; \&\& \; q_{iln}^{jk} = 0 \; \forall \; l = L1, L2 \qquad [18]$$

$$M.\ E_{iln}^{jk} \geq \sum_{k'>k} Y_{il'n}^{jk'} \qquad \forall \quad l = L2,;\ l' = L3;\ n,\ i,\ j\ \&\&\ q_{iln}^{jk} = 0\ \forall\ l = L2, L3 \qquad [19]$$

Constraint [20] puts an upper limit on the number of workers that can be trained on the same operation-skill level combination in the same shift.

$$\sum_l \sum_i Y_{iln}^{jk} \leq tl \qquad \forall \qquad j \in J,\ k \in T_j,\ n \in N \qquad [20]$$

Constraint [21] puts an upper limit on the number of assignments of a worker that can be made in one shift.

$$\sum_l \sum_n Z_{iln}^{jk} \leq\ wl \qquad \forall \qquad j \in J,\ k \in K_j,\ i \in W_j \qquad [21]$$

Constraint [22] puts an upper limit on the number of times a worker can be trained during the planning horizon.

$$\sum_k \sum_n \sum_l S_{iln}^{jk} \leq\ \ Lim \qquad \forall \qquad i \in W_j,\ j \qquad [22]$$

Constraints [23] and [24] ensure that training of skill levels L2 and L3 only occurs if there is no work currently taking place at the particular operation in that shift.

$$\sum_i \sum_{l=L2,L3} Y_{iln}^{jk} \leq M.\ TW_n^{jk} \qquad \forall \quad j \in J, k \in K_j, n \in N \qquad [23]$$

$$\sum_i \sum_l Z_{iln}^{jk} \leq M.\left(1 - TW_n^{jk}\right) \qquad \forall \quad j \in J, k \in K_j, n \in N \qquad [24]$$

Constraints [25] and [26] enforce the condition that for any operation, in any shift of the planning horizon, training cannot occur at multiple skills levels.

$$\sum_i Y_{iln}^{jk} \leq \; M.\; ST_{ln}^{jk} \qquad \forall \quad l, n, j, k \in T_j \tag{25}$$

$$\sum_l ST_{ln}^{jk} \leq 1 \qquad \forall \;\; j, n, k \in T_j \tag{26}$$

If there are any pre-allocations for cross-training before the beginning of the planning horizon, then they can be assigned in the model using Constraint [27].

$$\epsilon_{iln}^{jk} \leq Y_{iln}^{jk} \qquad \forall \;\; i, l, n, j, k \tag{27}$$

Similarly, the partial allocations assigned in the model before the planning period starts can be assigned using Constraint [28].

$$\varphi_{iln}^{jk} \leq X_{iln}^{jk} \qquad \forall \;\; i, l, n, j, k \tag{28}$$

Constraint [29] ensures that the duration a worker can be assigned on any operation skill-level combination in any shift is between 0 and 1, i.e, from 0% of the worker’s shift to 100% of the worker’s shift.

$$0 \leq X_{iln}^{jk} \leq 1 \qquad \forall \;\; i, l, n, j, k \tag{29}$$

Constraint [30] specifies the binary variables used in the model.

$$S_{iln}^{jk},\ Z_{iln}^{jk},\ E_{iln}^{jk},\ Y_{iln}^{jk}, TW_{n}^{jk}, ST_{ln}^{jk} \in \{0,1\} \quad \forall\ i, l, n, j, k \qquad [30]$$

3.3 COST PARAMETER STRUCTURE

Here, we present the structure of the cost parameters used in the objective function presented in the previous sub-section. For every operation skill-level scenario, we can split the set of workers I in any shift type j, into two groups; I_{ln} – the set of workers qualified on skill level l and operation n and $I_{l'n'}$- the set of workers not qualified on skill level l and operation n. Also, if worker I_{ln} is present in shift type j, then we denote the worker as I_{ln}^{j}, and if the worker is not present in shift type j, then we denote the worker as $I_{ln}^{j'}$.

Definition 1: An operation-skill level combination is defined as a particular instance of the operation and the skill level. For example, if there are three operations - OP1, OP2 and OP3, and two skill levels – L1 and L2, then the set of operation-skill level combinations is {OP1-L1, OP2-L1, OP3-L1, OP1-L2, OP2-L2, OP3-L2}.

Definition 2: A skill level relaxation occurs when there exists a requirement on an operation of a specific skill level and it is filled by a worker qualified on the same operation but of a lower skill level.

Theorem 1: *This theorem is derived from a theorem in Jarugumilli (2011). If an optimal solution is to be obtained, the cost of assigning a worker that is qualified on a particular skill level operation combination, is less than the cost associated with assigning a worker that is not qualified on that same operation skill-level combination.*

Proof: We need the qualified workers to perform the task, not the ones that are not qualified. Since, the optimal value of the objective function is a minimization of the costs, we obtain $c_{iln} < c_{i'ln}$, *i'* is the worker that is not qualified on skill l and operation n.

Theorem 2: *If the cost of allocating any worker qualified on a particular operation-skill level combination is less than the cost of cross-training any worker on the same operation-skill level combination, i.e.,*$c_{iln} < ct_{xln}$*, for x = i and x≠i., then we obtain an optimal solution.*

Proof: This can be divided into 2 conditions, $(i) i = x = I_{ln}$ and $(ii) i = I_{ln}$, $x = I_{l'n'}$. Also, we can say that the objective function value for each case is W1 and W2.

For (i), we compare the costs of cross-training and allocation for the same worker. Assuming he is qualified on OP2-L1, we do not need to assign him to cross-training for OP2-L1. Hence, if $c_{iln} > ct_{xln}$, the value W1 would favor cross-training and not allocation. Hence, $c_{iln} < ct_{xln}$.

For (ii), we consider a worker that is not qualified on the operation-skill level combination to another worker that is qualified on it. From Theorem 1, for worker *i* we obtain $c_{iln} < c_{xln}$. Also, from part (i) of this theorem, we obtain that $c_{xln} < ct_{xln}$. Hence, we can prove that $c_{iln} < ct_{xln}$.

Theorem 3: *If a skill level relaxation takes place, the cost of assigning a worker to the different skill level on the same operation is always higher.*

Proof: If a skill level relaxation takes place for skill level l, it means that instead of assigning worker I_{ln} on operation n, we have assigned worker $I_{l'n}$ on operation n. This means that one less worker is available to be assigned on skill level l'. There is an extra overhead cost that needs to be assumed for this. Hence, $c_{iln} < c_{il'n}$.

Theorems 1, 2 and 3 were concerned with comparing the costs for single workers. Now, we will consider costs for overtime allocation, workforce gap, and slack/surplus in cross-training for the complete set of workers present in each shift type.

Definition 3: The workforce gap is defined the gap observed at every operation-skill level combination in each shift in the planning horizon.

Definition 4: The slack and surplus in cross-training is the number of shifts lesser than and greater than the cross-training targets (specified at the beginning of the planning horizon) respectively for every operation skill level combination.

Theorem 4: *The cost of cross-training surplus is less than the cost of cross-training slack, which is in turn less than the cost of total overtime per shift which is in turn less than the cost of the workforce gap per shift.*

Proof: We will prove this theorem in parts. (i) $p_{ln} < ps_{ln}$ (ii) $pp_{ln} < pn_{ln}$ (iii) $pn_{ln} < p_{ln}$

The cost of a workforce gap at a particular operation skill level combination is the highest. This is because a workforce gap causes a shortage in production, and a shortage causes a loss of profit and reputation for the company. Hence, $p_{ln} < ps_{ln}$ and (i) holds true.

Similarly, a surplus in cross-training targets is preferable as compared to a slack. Hence, $pp_{ln} < pn_{ln}$, and (ii) holds true.

We can prove (iii) by negation. The cost parameters need to be structured such that satisfying the operational demand is a higher priority as compared to worker training. If we assume $pn_{ln} > p_{ln}$, the cost of training slack will be higher than the cost of overtime allocation at skill level l on operation n. For scenarios in which the operational demand is large, instead of allocating

overtime to meet the extra demand, training assignments will be given a priority in the objective function. Hence, $pn_{ln} < p_{ln}$.

Tables 3.3.1 and 3.3.2 show an example that denotes the values taken for the cost parameters in the model. As we can observe from Table 3.3.1, the cost parameters for a worker qualified at a skill level and performing work at that skill level are the lowest values in the table. L1 at L1 is 270/271/272, L2 at L2 is 210/211/212 and L3 at L3 is 100/101/102. These three parameters are in descending order, because L3 work is generally considered more important and the workers qualified on L3 should be working at L3 instead of working on lower skill levels. It is to be noted that the specific values themselves are not important as the relationship between these values.

Table 3.3.2 highlights the difference in costs for critical operations. The cost parameters are close to each other, but the cost of workforce gaps is higher for critical operations, the cost of training slack is higher and the cost of training surplus is lower for critical operations.

Trained_Skill Level	Performs Work of	Preference 1	Preference 2	Preference 3
L1	L1	270	271	272
L1	L2	5900	5910	5920
L1	L3	6700	6710	6720
L2	L1	2300	2310	2320
L2	L2	210	211	212
L2	L3	5300	5310	5320
L3	L1	1900	1910	1920
L3	L2	1700	1710	1720
L3	L3	100	101	102

Table 3.3.1: Worker Assignment Cost Parameters

Critical Operation	Training Cost	Cost of Overtime	Cost of Training Slack	Cost of Training Surplus	Cost of Workforce Gap
Yes	5001	12000	11000	9000	1200000
No	5001	12500	10500	9500	1000000

Table 3.3.2: Other Cost Parameters

3.4 Implementation Architecture

Figure 3.4.1 shows the implementation architecture for the mathematical model. The input requirements are converted into a Derived Requirements document, which is then used by a C# code. The solver used to perform the optimization is IBM ILOG CPLEX 12.4.

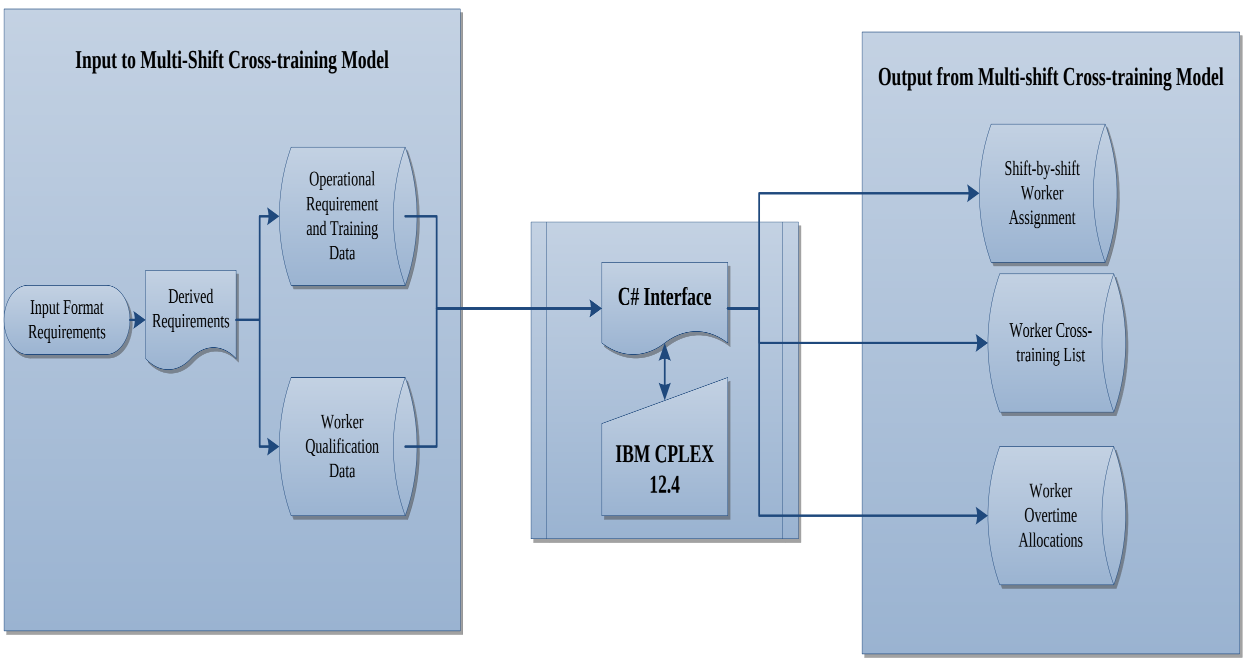

Figure 3.4.1: Implementation Architecture

The input requirements are given in terms of values such as man-machine ratio, L1:L2:L3 ratio, and the tool count needed for each shift. These values are then converted in a derived requirements excel sheet that contains the requirement in terms of number of shifts needed at

each operation-skill level combination (additional information on this procedure is given in Section 4). Similarly, even the training data is specified in the form of number of shifts. Furthermore, the workers qualification matrix at the beginning of the planning period is also needed as an input. An example of the input format is shown in the Table 3.4.1, and the corresponding derived requirements are shown in Tables 3.4.2(a), (b) and (c). The columns of Table 3.4.1 are described below.

Operation: It is the actual operation under consideration on the manufacturing floor.

MMR (Man-Machine Ratio): The value that is used to convert the amount of workers needed with respect to the amount actually needed at the machines.

L1:L2:L3: This is the ratio of the requirement for each skill level at every operation.

Tool Count: This denotes the actual machine requirement at each shift type. Hence, here we have values for ST1, ST2, ST3 and ST4 in each column.

The calculation for each cell in Table 3.4.3 is given as:

L1 Req. = (Tool Count)*MMR*L1 / (L1+L2+L3)

L2 Req. = (Tool Count)*MMR*L2 / (L1+L2+L3)

L3 Req. = (Tool Count)*MMR*L3 / (L1+L2+L3)

The values denote the total number of worker-shifts that are needed to satisfy the requirement for each skill and each operation for every shift type.

Operation	MMR	L1:L2:L3	Tool Count ST1	Tool Count ST2	Tool Count ST3	Tool Count ST4
Op1	2:3	1:1:1	0.45	0.45	0.63	0.45
Op2	1:2	1:1:2	3.6	1.2	2	2.4
Op3	2:5	1:1:3	2.5	4	2	3.5

Table 3.4.1: Input Format

L1 Req.	ST1	ST2	ST3	ST4
OP1	0.1	0.1	0.14	0.1
OP2	0.45	0.15	0.25	0.3
OP3	0.2	0.32	0.16	0.28

Table 3.4.2(a): Derived Requirements (L1)

L2 Req.	ST1	ST2	ST3	ST4
OP1	0.1	0.1	0.14	0.1
OP2	0.45	0.15	0.25	0.3
OP3	0.2	0.32	0.16	0.28

Table 3.4.2(b): Derived Requirements (L2)

L2 Req.	ST1	ST2	ST3	ST4
OP1	0.1	0.1	0.14	0.1
OP2	0.9	0.3	0.5	0.6
OP3	0.6	0.96	0.48	0.84

Table 3.4.2(c): Derived Requirements (L3)

The output of the model provides the shift-by-shift worker assignment, the worker cross-training list and the worker overtime allocations. The output tables for a numerical example are

described in detail in Section 4. Section 4 also contains a shift type based view of these allocations and also an individual shift based view of each worker during the shift.

4. NUMERICAL EXAMPLE

In this section, we discuss a numerical example that has been solved using the integrated multi-shift cross-training model formulated in Section 3. The specifications of this test case are given in Table 4.1.

Shifts	28
Shift Types	4
Number of shifts in each Shift Type	7
Workers	15
Operations	5
Skill levels	L1, L2, L3
Regulars Worker / Shift type	1-6 (ST1) 7 (ST2) 8-11 (ST3) 12-15 (ST4)

Table 4.1: Test Case Specifications

The tables that act as input to the model and output to the model are discussed below.

Tables 4.2 (a), (b) and (c) show the number of shifts of regular allocation (parameter r_{ln} needed for each operation skill level combination in a shift of each shift type, i.e, these are the derived requirements.

L1 Req.	ST1	ST2	ST3	ST4
OP1	0.50	1.19	0.51	0.96
OP2	0.51	0.48	0.72	0.50
OP3	0.47	0.42	0.72	0.39
OP4	0.30	0.47	0.60	0.65
OP5	0.33	0.43	0.45	0.49
Total	**2.11**	**3.00**	**3.00**	**3.00**

Table 4.2(a): L1 Requirements for Test Case

L2 Req.	ST1	ST2	ST3	ST4
OP1	0.25	0.25	0.25	0.25
OP2	0.33	0.26	0.25	0.25
OP3	0.27	0.25	0.25	0.25
OP4	0.31	0.25	0.40	0.38
OP5	0.25	0.33	0.34	0.25
Total	**1.40**	**1.34**	**1.49**	**1.38**

Table 4.2(b): L2 Requirements for Test Case

L3 Req.	ST1	ST2	ST3	ST4
OP1	0.28	0.25	0.25	0.25
OP2	0.25	0.31	0.25	0.25
OP3	0.31	0.25	0.25	0.25
OP4	0.25	0.25	0.28	0.25
OP5	0.25	0.33	0.26	0.42
Total	**1.35**	**1.39**	**1.28**	**1.42**

Table 4.2(c): L3 Requirements for Test Case

Table 4.3 denotes the critical operations present on the manufacturing floor. The critical operations are important to obtain effective production in the manufacturing process. Hence, a gap at these operations provides more of a problem than a gap faced at other operations.

Operation	Critical Operation	Cross Training Area
Op1	No	Assembly
Op2	No	Assembly
Op3	Yes	Assembly
Op4	No	Assembly
Op5	No	Assembly

Table 4.3: Critical Operations

Table 4.4 denotes the workers assigned to each shift type.

Name	Shift Type
Worker01	ST1
Worker02	ST1
Worker03	ST1
Worker04	ST1
Worker05	ST1
Worker06	ST1
Worker07	ST2
Worker08	ST3
Worker09	ST3
Worker10	ST3
Worker11	ST3
Worker12	ST4
Worker13	ST4
Worker14	ST4
Worker15	ST4

Table 4.4: Worker-Shift Type Details

Table 4.5 shows the qualifications of each worker on each operation-skill level combination. The numbers 1, 2 and 3 denote the preference of the worker on that particular operation. Preference 1 means that the worker prefers to work on this operation as compared to any other operation for that particular skill level. (Similarly for preference 2 means the worker prefers to work on one operation before this operation). The fields that do not have any number present indicate that the worker is not qualified on any of these operations. The workers are eligible to be trained on these operation-skill level combinations.

Table 4.6 shows the shifts of training needed for each operation-skill level combination. The numerical value indicates the number of worker-shifts of training that need to be completed in this time horizon. This number can be obtained from strategic level models that make use of the long-term results to extrapolate the training levels needed for two weeks.

Table 4.7 shows the number of shifts needed to complete training at a particular skill level, denoted as Mshifts. This number remains constant over all operations.

Name	Cert Type	Op1	Op2	Op3	Op4	Op5
Worker01	L1		1			
Worker02	L1	1		2	3	
Worker03	L1	1				
Worker04	L1				1	
Worker05	L1			2		1
Worker06	L1					
Worker07	L1	2			3	1
Worker08	L1			1		
Worker09	L1		1			2
Worker10	L1					
Worker11	L1			1		
Worker12	L1	1				
Worker13	L1					
Worker14	L1		1		2	3
Worker15	L1		1			
Worker01	L2			1		
Worker02	L2					
Worker03	L2					1
Worker04	L2					
Worker05	L2					
Worker06	L2	1	2			
Worker07	L2					1
Worker08	L2		1			
Worker09	L2			1		2
Worker10	L2					
Worker11	L2					
Worker12	L2					
Worker13	L2					1
Worker14	L2			1		
Worker15	L2				1	2
Worker01	L3					
Worker02	L3					
Worker03	L3			1		
Worker04	L3		1			
Worker05	L3					
Worker06	L3					
Worker07	L3					
Worker08	L3				1	
Worker09	L3					
Worker10	L3		2	1		3
Worker11	L3					
Worker12	L3					
Worker13	L3			1		
Worker14	L3	1				
Worker15	L3			1		

Table 4.5: Worker Operation-Skill Level Qualifications

Operation	L1	L2	L3
OP1	18	16	18
OP2	6	5	15
OP3	2	20	9
OP4	11	19	9
OP5	10	5	15

Table 4.6: Number of shifts of training for each Operation-Skill Level Combination

Skill levels	Mshifts
L1	1
L2	2
L3	2

Table 4.7: Number of Shifts needed to train each worker on a skill level

The cross-training model is solved using CPLEX and a raw excel sheet output is obtained. The raw output has been formatted in Tables 4.8 (a), (b), (c), (d). The output is shown for ST1, ST2, ST3 and ST4 in each sub-table. Every column in Table 4.8 denotes parameters essential to the output. These parameters are described below.

Number of Qualified Workers: Provides a reference to the floor manager regarding the number of workers available to perform a particular task.

Qualified Workers for Current Shift: Displays the qualified workers available in the current shift.

Qualified Workers for Overtime: Shows the number of workers in other shift types that are capable of doing the task.

Worker: The actual worker number that is doing the particular task.

Allocation in Shifts: The individual value of the fraction of the shift duration that the particular worker has been assigned on the operation-skill level combination. This is given for each of the seven shifts present in the shift type. Hence, for ST1, it is given for shift 1, 3, 5, 7, 15, 17, 19. The values in red denote overtime or training.

Requirement: Denotes the number of worker-shifts needed (values in Table 4.2).

Gap: Denotes the workforce gap present at each operation, skill level combination after solving the model.

Tables 4.8(a)-(d) contain the output for regular, overtime and cross-training allocations. Overtime allocations are indicated by an “OT” in the Worker column, while training allocations are indicated by a “Tr” in the Allocation in Shifts column. The following paragraphs summarize the observations from Tables 4.8(a)-(d).

As we can see from Table 4.8(a), regular work allocations take place for Worker02 and Worker03. These satisfy the requirement at the operation-skill level combination OP1-L1. Also, Worker05 is trained on OP1-L1 in shift number 17. Similarly for skill level OP1-L3, there are overtime allocations. This means that a worker from either Worker09, Worker14 or Worker15 is assigned to the task in this shift. (since these are the workers qualified on OP1-L3 in ST3 and ST4.)

For OP2, we can see that Worker04 is actually qualified on L2, but performs on OP2-L1 as well. This is because a worker that is qualified at a higher skill level L2 on an operation can also do a task on the same operation at L1, since this is a lower skill level. For OP3-L1, we see that there are two workers that are being trained, i.e, Worker04 in shift 5, and Worker 01 in shift 19. Similarly, OP2 has Worker02 and Worker06 assigned to training. Overall, the amounts of overtime for this shift type are low, and there are no workforce gaps at any of the operation-skill level combinations.

In Table 4.8(b), there are a larger number of overtime assignments. This is due to the fact that this ST2 consists of only regular worker, i.e. Worker07, and most of the work is completed using overtime workers.

Tables 4.8(c) and 4.8(d) consist of a mix of regular and overtime assignments. Both ST3 and ST4 have 4 regular workers, Workers 8-11 for ST3 and Workers 12-15 for ST4. There are no workforce gaps observed for any shifts in ST3 and ST4. In Table 4.8(c), for OP1-L1, OP2-L2 and OP3-L3, no regular workers are qualified. Hence, overtime workers are being used to satisfy the operational requirements. Furthermore all 4 workers are assigned to other operations, and hence training on OP1 does not take place. There is also a skill level relaxation where Worker11 is qualified on OP3-L1 but is also assigned to OP3-L2.

Operation	Skill Level	Number of Qualified workers	Qualified Workers FOR CURRENT SHIFT	Qualified workers for overtime: Worker no. (Shift number)	Worker	Allocation in Shifts							Requirement	Gap
			W01-W06	*W08-W15*		1	3	5	7	15	17	19		
OP1	L1	3	W02, W03	W12 (ST4)	W02	0.5			0.5	0.5	0.5	0.5	0.5	0
					W03		0.5	0.5						
					W05						Tr			
	L2	1	W06	-	W06	0.25	0.25	0.25	0.25		0.25	0.25	0.25	0
					W02					0.25				
	L3	1	-	W14 (ST4)	W06	0.28	0.28	0.28	0.28				0.28	0
					W03					0.28	0.28	0.28		
OP2	L1	4	W01	W09 (ST3), W14, W15 (ST4)	W01	0.51	0.51	0.5	0.51	0.51	0.51	0.45	0.51	0
					OT			0.01				0.06		
					W02			Tr						
					W06					Tr				
	L2	2	W06	W08 (ST3)	W06	0.33	0.33	0.33	0.33		0.33	0.33	0.33	0
					W04					0.33				
	L3	2	W04	W10 (ST3)	W04	0.25	0.25		0.25	0.25	0.25	0.25	0.25	0
					W01			0.25						
OP3	L1	4	W02, W05	W08, W11 (ST3)	W05	0.47	0.42	0.42	0.47	0.42		0.42	0.47	0
					W02						0.47			
					OT		0.05	0.05		0.05		0.05		
					W04			Tr						
					W01							Tr		
	L2	3	W01	W09 (ST3), W14 (ST4)	W01	0.27	0.27	0.25	0.27	0.27	0.27		0.27	0
					W03							0.27		
					OT			0.02						
	L3	4	W03	W10 (ST3), W13, W15 (ST4)	W03	0.31	0.25	0.25	0.31	0.31	0.31	0.31	0.31	0
					OT		0.06	0.06						
OP4	L1	4	W02, W04	W07 (ST3), W14 (ST4)	W04	0.3	0.3		0.3	0.3	0.3	0.3	0.3	0
					W02			0.3						
	L2	1	-	W15 (ST4)	W04	0.31	0.31		0.31		0.31		0.31	0
					W02			0.31		0.25		0.25		
					OT					0.06		0.06		
	L3	1	-	W08 (ST3)	W02	0.25		0.25	0.25			0.25	0.25	0
					OT		0.25			0.25	0.25			
OP5	L1	3	W05	W09 (ST3), W14 (ST4)	W05	0.33	0.33	0.33	0.33	0.33		0.33	0.33	0
					W03						0.33			
	L2	4	W03	W09 (ST3), W13, W15 (ST4)	W03	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0
	L3	1	-	W10 (ST3)	W03	0.25			0.25				0.25	0
					W05		0.25	0.25		0.25		0.25		
					OT						0.25			

Table 4.8(a): Output for ST1

Operation	Skill Level	Number of Qualified workers	Qualified Workers FOR CURRENT SHIFT	Qualified workers for overtime: Worker no. (Shift number)	Worker	Allocation in Shifts							Requirement	Gap
			W07	*W08-W15*		2	4	6	8	16	18	20		
OP1	L1	2	W07	W12 (ST4)	W07	0.25	0.25	0.25	0.25	0.25	0.25	0.25	1.19	0
					OT	0.94	0.94	0.94	0.94	0.94	0.94	0.94		
	L2	0	-	-	OT	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0
	L3	1	-	W14 (ST4)	OT	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0
OP2	L1	3	-	W09 (ST3), W14, W15 (ST4)	OT	0.48	0.48	0.48	0.48	0.48	0.48	0.48	0.48	0
	L2	1	-	W08 (ST3)	OT	0.26	0.26	0.26	0.26	0.26	0.26	0.26	0.26	0
	L3	1	-	W10 (ST3)	OT	0.31	0.31	0.31	0.31	0.31	0.31	0.31	0.31	0
OP3	L1	2	-	W08, W11 (ST3)	OT	0.42	0.42	0.42	0.42	0.42	0.42	0.42	0.42	0
	L2	2	-	W09 (ST3), W14 (ST4)	OT	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0
	L3	3	-	W10 (ST3), W13, W15 (ST4)	OT	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0
OP4	L1	3	W07	W07 (ST3), W14 (ST4)	OT	0.47	0.47	0.47	0.47	0.47	0.47	0.47	0.47	0
	L2	1	-	W15 (ST4)	OT	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0
	L3	1	-	W08 (ST3)	OT	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0
OP5	L1	3	W07	W09 (ST3), W14 (ST4)	W07	0.42	0.42	0.42	0.42	0.42	0.42	0.42	0.43	0
					OT	0.01	0.01	0.01	0.01	0.01	0.01	0.01		
	L2	4	W07	W09 (ST3), W13, W15 (ST4)	W07	0.33	0.33	0.33	0.33	0.33	0.33	0.33	0.33	0
	L3	1	-	W10 (ST3)	OT	0.33	0.33	0.33	0.33	0.33	0.33	0.33	0.33	0

Table 4.8(b): Output for ST2

Operation	Skill Level	Number of Qualified workers	Qualified Workers FOR CURRENT SHIFT	Qualified workers for overtime: Worker no. (Shift number)	Worker	Allocation in Shifts							Requirement	Gap
			W08-W11	*W01-W07*		9	11	13	21	23	25	27		
OP1	L1	3	-	W02, W03, W07	OT	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0.51	0
	L2	1	-	W06	OT	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0
	L3	0	-	-	OT	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0
OP2	L1	2	W09	W01	W09	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.72	0
					W10	0.47	0.47	0.47	0.47	0.47	0.47	0.47		
	L2	2	W08	W06	W08	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0
	L3	2	W10	W04	W10	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0
OP3	L1	4	W08, W11	W02, W05	W11	0.72	0.72	0.72	0.72	0.72	0.72	0.72	0.72	0
	L2	2	W09	W01	W11	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0
	L3	2	W10	W03	OT	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0
OP4	L1	3	-	W02, W04, W07	W08	0.47	0.47	0.47	0.47	0.47	0.47	0.47	0.6	0
					OT	0.13	0.13	0.13	0.13	0.13	0.13	0.13		
	L2	0	-	-	OT	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.6	0
	L3	1	W08	-	W08	0.28	0.28	0.28	0.28	0.28	0.28	0.28	0.28	0
OP5	L1	3	W09	W05, W07	W09	0.41	0.41	0.41	0.41	0.41	0.41	0.41	0.45	0
					OT	0.04	0.04	0.04	0.04	0.04	0.04	0.04		
	L2	3	W09	W03, W07	W09	0.34	0.34	0.34	0.34	0.34	0.34	0.34	0.34	0
	L3	1	W10	-	W10	0.26	0.26	0.26	0.26	0.26	0.26	0.26	0.26	0

Table 4.8(c): Output for ST3

Operation	Skill Level	Number of Qualified workers	Qualified Workers FOR CURRENT SHIFT	Qualified workers for overtime: Worker no. (Shift number)	Worker	Allocation in Shifts							Requirement	Gap
			W12-W15	*W01-W07*		10	12	14	22	24	26	28		
OP1	L1	4	W12	W02, W03, W07	W12	0.96	0.96	0.96	0.96	0.96	0.96	0.96	0.96	0
	L2	1	-	W06	OT	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0
	L3	1	W14	-	W14	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0
OP2	L1	3	W14, W15	W01	W15	0.37	0.37	0.37	0.37	0.37	0.37	0.37	0.37	0
	L2	1	-	W06	OT	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0
	L3	1	-	W04	OT	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0
OP3	L1	2	-	W02, W05	OT	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0.13	0
	L2	2	W14	W01	W14	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0
	L3	3	W13, W15	W03	W15	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0
OP4	L1	4	W14	W02, W04, W07	W14	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.65	0
					OT	0.15	0.15	0.15	0.15	0.15	0.15	0.15		
	L2	1	W15	-	W15	0.38	0.38	0.38	0.38	0.38	0.38	0.38	0.38	0
	L3	0	-	-	OT	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0
OP5	L1	3	W14	W05, W07	W13	0.49	0.49	0.49	0.49	0.49	0.49	0.49	0.49	0
	L2	4	W13, W15	W03, W07	W13	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0.25	0
	L3	0	-	-	OT	0.42	0.42	0.42	0.42	0.42	0.42	0.42	0.42	0

Table 4.8(d): Output for ST4

Table 4.9 shows the output for a particular shift of ST1, ST13. Similar tables are available for every single shift, providing the floor manager a status update of which worker is assigned to which task.

Current Op - Skill (Allocation): This denotes the operation, skill level combination separated by a dash, the actual allocation in brackets, e.g., OP2-L1 (0.5).

New Op-Skill (Training): This denotes the training performed in the particular shift in the same format as the previous column.

As can be observed, there is only one worker being trained in this shift, i.e. Worker04 on OP3-L1. Workers 07-15 are not part of this shift, but they may do some work since overtime is associated with the results. The selection of the worker is left up to the floor manager present in the shift. However, for OP4-L3, Worker08 is the only worker that is qualified for overtime, and hence it is assigned for 0.25 of its shift duration to OP4-L3.

Shift	Worker	Current Op - Skill (Allocation)	New Op -Skill (Cross-training)
ST13	W01	OP2-L1 (0.5), OP2-L3 (0.25), OP3-L2 (0.25)	-
ST13	W02	OP4-L1 (0.3), OP4-L2 (0.31), OP4-L3 (0.25)	-
ST13	W03	OP1-L1 (0.5), OP3-L3 (0.25), OP5-L2 (0.25)	-
ST13	W04	-	OP3-L1 (1)
ST13	W05	OP3-L1 (0.42), OP5-L1 (0.33), OP5-L3 (0.25)	-
ST13	W06	OP1-L2 (0.25), OP1-L3 (0.28), OP2-L2 (0.33)	-
ST13	W07	Not part of Shift Type (unless assigned OT)	-
ST13	W08	OP4-L3 (0.25) (assigned as OT)	-
ST13	W09	Not part of Shift Type (unless assigned OT)	-
ST13	W10	Not part of Shift Type (unless assigned OT)	-
ST13	W11	Not part of Shift Type (unless assigned OT)	-
ST13	W12	Not part of Shift Type (unless assigned OT)	-
ST13	W13	Not part of Shift Type (unless assigned OT)	-
ST13	W14	Not part of Shift Type (unless assigned OT)	-
ST13	W15	Not part of Shift Type (unless assigned OT)	-

Table 4.9: Output of One Shift in the time horizon

5. TESTING SUMMARY

Additional testing was conducted for the following fifteen data sets. The worker qualification matrices for these data sets were generated using a C++ code developed to assist in the creation of test cases.

The description of these test cases is shown in Tables 5.1(a) – 5.1(e):

Shifts	28
Shift Type	4
Number of shifts in each shift type	7
Workers	15
Operations	5 / 10 / 23
Skill levels	L1,L2, L3
Regulars Worker / shift type	1-6 (ST1) 7 (ST2) 8-11 (ST3) 12-15 (ST4)

Table 5.1(a): Test Cases 1, 2, 3

Shifts	28
Shift Type	4
Number of shifts in each shift type	7
Workers	26
Operations	5 / 10 / 23
Skill levels	L1,L2, L3
Regulars Worker / shift type	1-6 (ST1) 7-13 (ST2) 14-19 (ST3) 20-26 (ST4)

Table 5.1(b): Test Case 4, 5, 6

Shifts	28
Shift Type	4
Number of shifts in each shift type	7
Workers	57
Operations	5 /10 / 23
Skill levels	L1,L2, L3
Regulars Worker / shift type	1-15 (ST1) 16-30 (ST2) 31-45 (ST3) 46-57 (ST4)

Table 5.1(c): Test Cases 7, 8, 9

Shifts	28
Shift Type	4
Number of shifts in each shift type	7
Workers	100
Operations	5 / 10 / 23
Skill levels	L1,L2,l3
Regulars Worker / shift type	1-25 (ST1) 26-50 (ST2) 51-75 (ST3) 76-100 (ST4)

Table 5.1(d): Test Case 10, 11, 12

Shifts	28
Shift Type	4
Number of shifts in each shift type	7
Workers	150
Operations	5 / 10 / 23
Skill levels	L1,L2,l3
Regulars Worker / shift type	1-37 (ST1) 38-75 (ST2) 76-112 (ST3) 113-150 (ST4)

Table 5.1(e): Test Cases 13, 14, 15

The technical specifications of the processor used in computing the cases were: Intel Core-i7-3770 @ 3.4 GHz, Installed Memory (RAM): 16.0 GB. The solving time for each test case is shown in Table 5.2. The solving time was observed in terms of seconds and for the test cases that were not terminated until the processor ran out of memory the relative optimality gap was observed. Every test case was computed within 2% optimality in less than an hour of computation. Since the model needs to be used for allocation once every two weeks, the solving times are practically feasible.

Workers	Operations	Solving Time
15	5	6 secs
15	10	2 secs
15	23	1 sec
26	5	25 secs
26	10	11 secs
26	23	9 secs
57	5	3400 secs (1.93%)*
57	10	2628 secs (0.44%)*
57	23	12 secs
100	5	4008 secs (0.17%)*
100	10	3830 secs (0.01%)*
100	23	1261 secs
150	5	22 secs
150	10	4221 secs (0.09%)*
150	23	3839 secs (0.12%)*

***** *Indicates that the processor ran out of memory during the computation*

Table 5.2: Solving Times for all Test Cases (optimality gap)

To analyze the dependencies of the solving time, we develop a multiple regression model in R (version 3.0.1). The independent variable is the logarithm of the solving time, while the dependent variables are the logarithm of the number of workers, the number of operations, and three other parameters: the average skill levels per operation (before solving), the worker to

operation ratio, and the requirements/worker. These parameters have been selected because they determine the size of the problem. For instance, the number of variables and constraints are directly proportional to the initial number of qualifications since regular allocation variables are not created for the workers that are not qualified (value of 1 or 2 or 3 in the worker-qualification matrix) on a particular operation. Hence, we consider the initial average number of skill levels per operation while analyzing the dependencies for the solving time. Similarly, increases in the requirements/worker parameter and the worker/operation ratio increases the size of the model.

Figure 5.1 shows the summary of the multiple regression model in R. The three significant factors (p-value < 0.05) are the logarithm of the number of workers, the number of operations and the initial average number of skill levels per operation. Also, from the R-squared value of the model, we can say that 92.39% of the variability associated with the solving time is depicted by the current fit.

```
Call:
lm(formula = log(SolvingTime) ~ log(Workers) + Operations + SkPerOp +
    Req + WPerO)

Residuals:
    Min      1Q  Median      3Q     Max
-1.5611 -0.4682  0.2704  0.3415  2.2505

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept)  -8.08213    3.75499  -2.152  0.05981 .
log(Workers)  4.92544    0.64053   7.690 3.03e-05 ***
Operations   -0.19313    0.06584  -2.933  0.01667 *
SkPerOp      -0.04928    0.01253  -3.933  0.00344 **
Req          -1.47190    2.22565  -0.661  0.52497
WPerO        -0.08286    0.09441  -0.878  0.40293
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.157 on 9 degrees of freedom
Multiple R-squared:  0.9239,    Adjusted R-squared:  0.8817
F-statistic: 21.86 on 5 and 9 DF,  p-value: 8.676e-05
```

Figure 5.1: Summary of Multiple Regression Model

Figure 5.2 shows the linear relationship between the logarithms of the number of workers and the solving time. There are 2 outliers that are present below the line. These are highlighted by points P1 and P2. P1 depicts a case with 57 workers and 23 operations where the solving time is reduced due to the individual operational requirements being below the minimum threshold available to assign a worker at that task. P2 is the outlier with 150 workers and 5 operations (same as the one mentioned in the preceding paragraph)

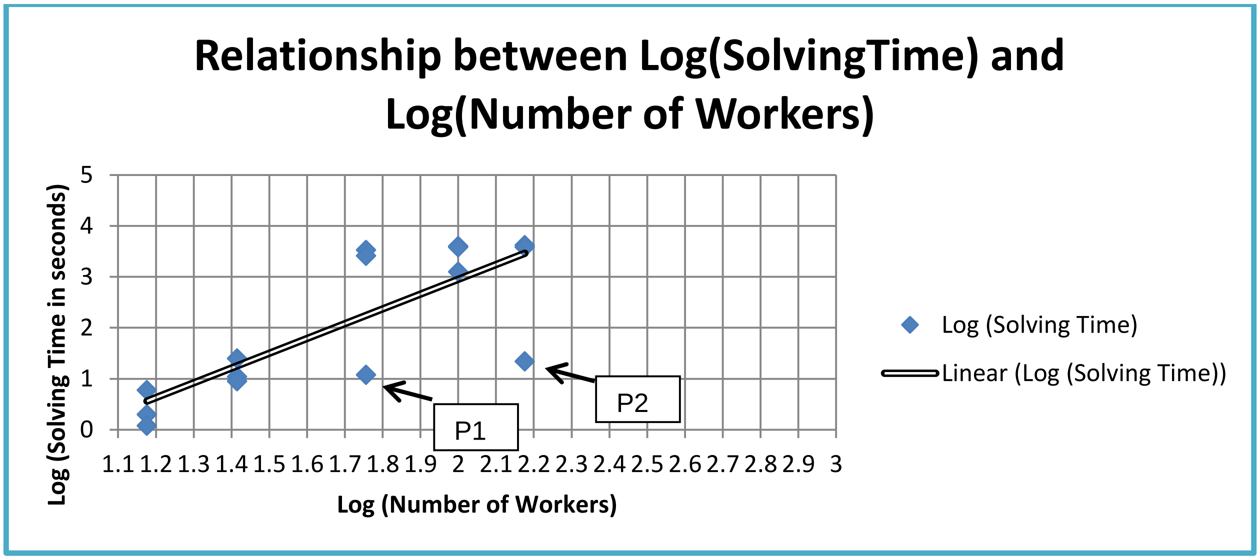

Figure 5.2: Logarithm (Solving Time) v/s Logarithm (Number of Workers)

Table 5.3 shows the relation between the solving time and the average number of skill per operation. As seen, for a fixed number of workers, the solving times reduce as the number of skills per operation reduces (apart from the outlier highlighted).

Workers	Skills Per Operation	Solving Time
15	8.19	6
	7.40	2
	2.92	1
26	24.60	25
	12.30	11
	5.96	9
57	20.63	3400
	12.83	2628
	12.17	12
100	92.00	4008
	51.30	3830
	21.70	1261
150	141.30	22
	73.95	4221
	33.98	3839

Table 5.3: Number of Skills per operation and solving time

Figure 5.3 shows the graph between the logarithm of the solving time versus the number of skill levels per operation. This has an exponential distribution with one outlier that is depicted by point P2. P2 depicts the test case with 150 workers and 5 operations and no overtime allocations.

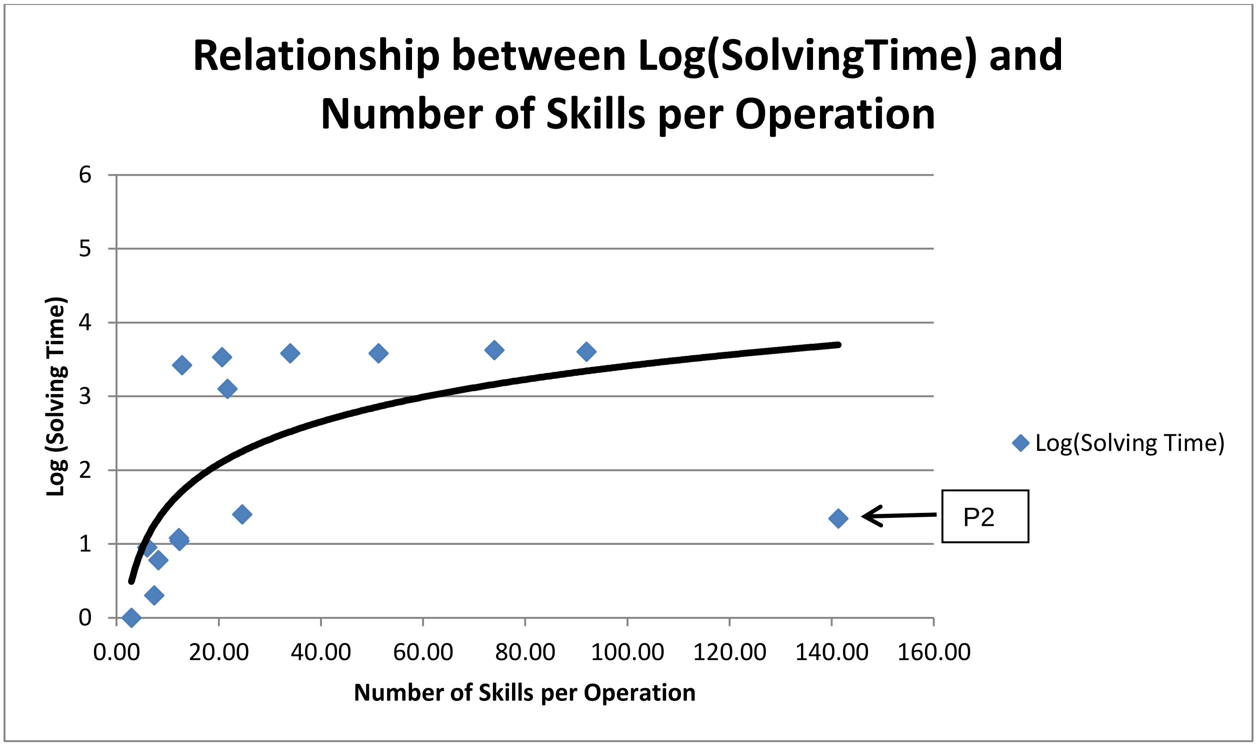

Figure 5.3: Logarithm (Solving Time) v/s Number of Skills Per Operation

To produce a better fit, points P1 and P2 - linked to the outliers in figures 5.2 and 5.3, have been removed from the multiple regression model, and new graphs that compare the logarithm of the solving time to the logarithm of the number of workers and the average number of skill levels per operation have been plotted.

Figure 5.4 depicts the results in R of the improved regression model. As seen, the R-squared value increases from 92.39% to 97.11%. Similarly, Figure 5.5 shows the improved linear fit between the log of the number of workers and the solving time, and Figure 5.6 shows the better exponential fit between the solving time and the average number of skills per operation.

```
Call:
lm(formula = log(SolvingTime1) ~ log(Workers1) + Operations1 +
    SkPerOp1 + Req1 + WPerO1)

Residuals:
     Min       1Q   Median       3Q      Max
-1.09695 -0.25173  0.06554  0.34649  1.00703

Coefficients:
               Estimate Std. Error t value Pr(>|t|)
(Intercept)   -6.431625   2.348225  -2.739  0.02549 *
log(Workers1)  3.870403   0.468823   8.256 3.48e-05 ***
Operations1   -0.119485   0.045929  -2.602  0.03154 *
SkPerOp1      -0.029447   0.007841  -3.756  0.00558 **
Req1          -1.483497   1.389193  -1.068  0.31673
WPerO1         0.074497   0.081895   0.910  0.38958
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.7202 on 8 degrees of freedom
Multiple R-squared: 0.9711,     Adjusted R-squared:  0.9531
F-statistic: 53.84 on 5 and 8 DF,  p-value: 6.043e-06
```

Figure 5.4: Improved Fit for the Multiple Regression model

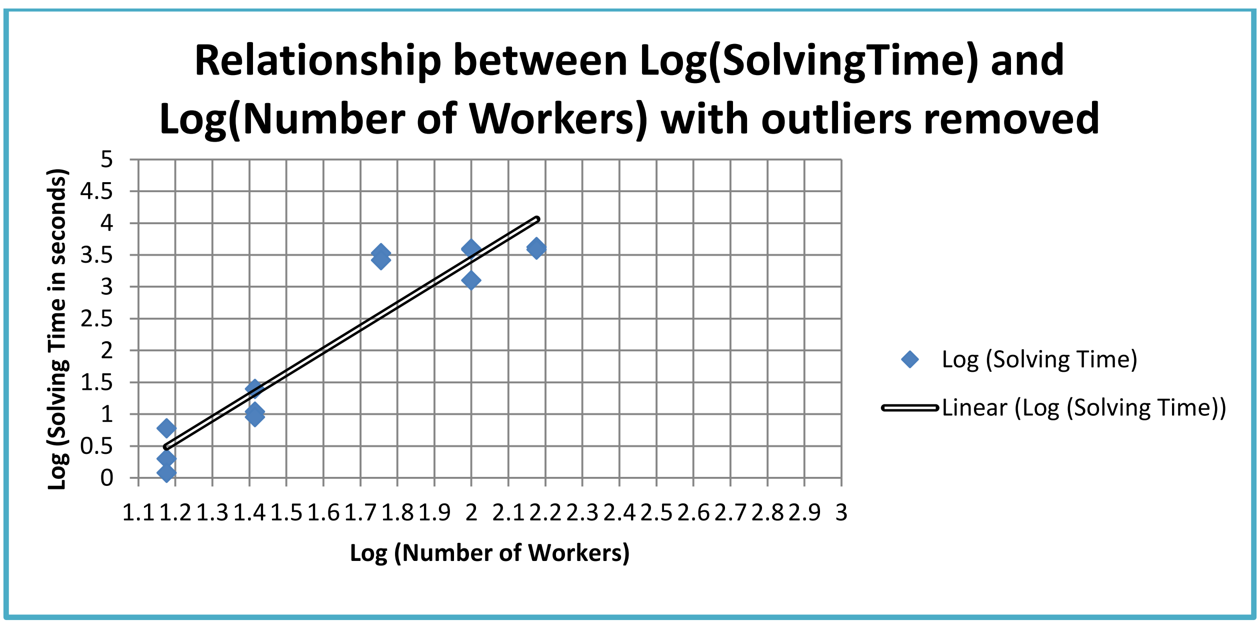

Figure 5.5: Improved Linear Relationship between solving time and number of workers

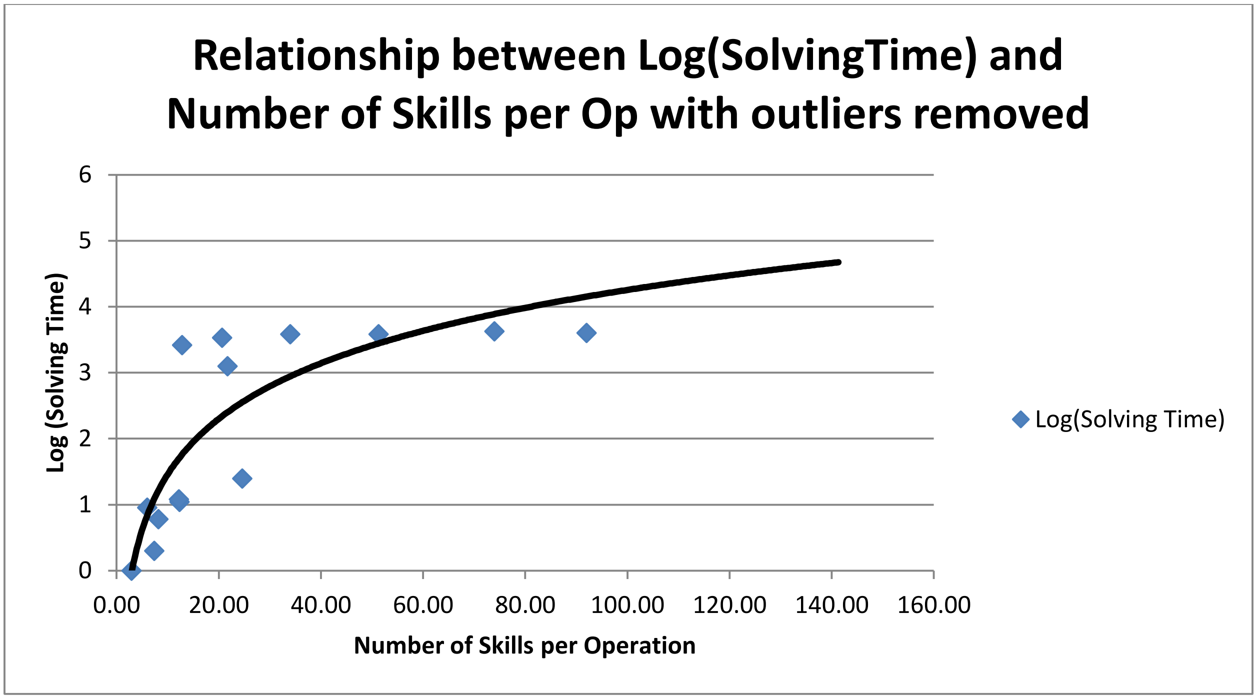

Figure 5.6: Improved Exponential Relationship between Solving Time and the Average no. of skill levels/op

5.1 Testing for rolling horizon

To observe changes in overtime costs, testing has been performed for 84 shifts, i.e., 3 planning periods for each test case. The worker qualifications have been updated after every planning period and considered as input for the next planning period. The model is then resolved with the updated input qualification matrix. The cross-training requirements have been kept uniform across each planning period. To ensure that the overall run time for each test case was within reasonable time durations, a time limit of 1000 seconds was added to the CPLEX parameters for every run.

The percentage overtime cost savings per worker observed are shown in Table 5.1.1. The values in the percentage overtime cost reduction per worker column are obtained when the model is solved for the set of cost parameters given in Section 3.3. As seen in Table 5.1.1, there are

significant overtime cost savings for cases with 26, 57 and 100 workers. When the number of workers was increased to 150, the savings in worker overtime drop off. This is due to factory limitations associated with cross-training of workers. In particular, the limits on the amount of cross-training allowed per shift and the amount of cross-training allowed for the time horizon of two weeks are the parameters that limit the improvements in overtime costs due to cross-training. For the case with 15 workers the reductions observed are low because ST2 for this set of test cases consists of only one regular worker, hence the amount of overtime assigned before and after training is the same for ST2.

In general, if the training limits are set higher than the number of workers available, then the reduction in OT cost is significant. Also, certain test cases with 15/26 workers have an unequal distribution of workers over different shift types, and hence the reduction in OT costs is lower due to particular shift types using greater overtime workers than others.

Workers	Percentage Reduction in OT Cost/Worker
15	1.04%
26	7.74%
57	28.81%
100	16.23%
150	0.01%

Table 5.1.1: Percentage Overtime Cost Reduction / Worker

Table 5.1.2 shows the initial and final average number of skill levels per worker for all the cases before and after three planning periods. There is a noticeable reduction in the percentage skill level increase for the last three rows (which are the readings for cases with 150 workers) as compared to the earlier instances. This is due to the low cross-training taking place in these cases. For cases where the initial number of skill levels per workers is low, such as for a few

cases with 15 and 57 workers (row 1 and row 7 in Figure 5.1.2), there are higher values observed for the percentage skill level increase per worker. This is because of a lower initial worker qualification level as compared to other cases.

Workers	Operations	Number of Skills/Worker (Initial)	Number of Skills/Worker (Final)	% Increase
15	5	2.73	3.80	39.02
15	10	4.93	6.20	25.67
15	23	4.47	6.87	53.73
26	5	4.73	5.50	16.26
26	10	4.73	5.69	20.32
26	23	5.27	7.85	48.90
57	23	1.81	4.07	125.24
57	5	2.25	4.58	103.90
57	10	4.91	6.35	29.28
100	10	4.60	5.86	27.39
100	5	5.13	5.97	16.37
100	23	4.99	6.35	27.25
150	23	4.71	5.13	8.92
150	10	4.93	5.20	5.54
150	5	5.21	5.41	3.83

Table 5.1.2: Increase in Worker Qualification after 3 Planning Periods

5.2 Changing cross-training requirements

Testing was also conducted by varying the cross-training requirements for each test case and solving the case for a single planning period. The parameter in the model that has been changed is rc_{ln} for every l, n. rc_{ln} takes three different values in each test case; low (25% of the total number of workers available), moderate (50% of the total number of workers available) and high (100% of the number of workers available). The regular worker requirement is kept at 80% of the total worker capacity in each case. To allow for a higher number of cross-training shifts in

order not to limit the calculation, the limit on training a worker only once in a planning period has been relaxed to thrice in a period. Also each case has been solved with a CPLEX time limit of 1000 seconds to maintain consistency with the testing in Section 5.1.

The results for training with different requirements are shown in Table 5.2.1.

Amount of Cross-training Needed	Average Percentage of cross-training performed
Low	70.06%
Moderate	60.34%
High	50.47%

Table 5.2.1: Amount of cross-training performed for each variation

Forty-five observations have been taken. (Fifteen - a combination of number of workers – 15, 26, 57, 100, 150 and number of operations – 5, 10, 23, times three - low, moderate and high cross-training requirements) The average percentage of cross-training performed was obtained by observing the number of shifts that were actually performed (by the end of the planning period) divided by the number of shifts that were needed (at the beginning of the planning period). As seen in Table 5.6, the amount of cross-training needed is increased, the percentage of that cross-training actually performed reduces. This is attributed to the fact that as the cross-training requirements increase, the number of training shifts needed of higher skill levels (L2/L3) increase significantly. However, the step-by-step qualification constraints only allow limited shifts of cross-training on L2/L3 skill levels to be performed. This results in a lower percentage of cross-training shifts being performed.

6. CONCLUSIONS AND FUTURE WORK

This thesis develops a multi-shift workforce allocation model that optimally allocates workers for regular work, overtime work and cross-training procedures. An applicable deterministic formulation using mixed integer programming is tested and analyzed. The objective function of the model minimizes the cost associated with the three worker allocation actions, as well as the costs associated with workforce gaps and training slack/surplus. The mathematical formulation also takes into account factory floor conditions and ensures that training and overtime rules are followed for the duration of the planning horizon. A software implementation of the model has been performed using the C# .NET framework in conjunction with the IBM ILOG CPLEX 12.4 Optimizer. A numerical example is discussed in detail along with a description of the input and output data associated with model. Additional testing consisting of larger data sets has also been performed.

As observed in Section 4, the model provides individual worker-operation based schedules to the floor manager for every shift of production. From Section 5, it was observed that for cases where the number of workers is not sufficient to meet operational requirements, a fewer number of shifts of training are performed as compared to situations when there are sufficient workers to meet the workforce demand. Occasionally, workers that are qualified at higher skill levels on an operation are needed to perform work of lower skill levels on the same operation. In many instances, a worker’s shift involves working on different operations for partial shift durations. From the solving time analysis in Section 5, it was observed that the mathematical model obtains near optimal solutions in practically implementable time durations (since the model only needs to be run once every two weeks). Furthermore, testing over rolling horizon and

testing after changing the cross-training requirements is also performed in this section.

This mathematical model can also be used in conjunction with larger scale workforce planning, particularly in integration with strategic models such as Kulkarni et al. (2013) that deal with a longer time horizon. Also, since this is an industry-oriented work, the model is developed taking factory floor rules into account. Considerations such as the maximum number of worker assignments per shift, the maximum number of training assignments per shift, minimum allocation for a worker, continuous shift-by-shift training procedures and step-by-step skill level training are part of the constraints in the mathematical model. If this work was to be applied to other areas, these constraints would need to be modified in order to make the model transferable. Lastly, since this is a cost-based model, the relationship between the cost parameters has been determined using industry-feedback for the system under consideration. If this model was to be applied to another area, a new cost structure based on the internal policies of user would need to be developed.

Future work could involve applying these principles to areas such as U-lines and work sharing systems. Also, cross-training policies could be specifically designed to consider the positive and negative impacts on team performance and morale. Another possible extension is the idea that worker learning could highly impact the cross-training schedule and assignment requirements. Lastly, this work could also be extended to include conditions where the training procedures were determined to be non-continuous and if the workers were made to be available in overlapping shifts.

REFERENCES:

Askin, R. G., & Huang, Y. (2001). Forming effective worker teams for cellular manufacturing. International Journal of Production Research, 39(11), 2431-2451.

Aykin, T. (1996). Optimal shift scheduling with multiple break windows. Management Science, 42(4), 591-602.

Bailey, D. E. (1998). Comparison of manufacturing performance of three team structures in semiconductor plants. Engineering Management, IEEE Transactions on, 45(1), 20-32.

Bhatnagar, R., Saddikutti, V., & Rajgopalan, A. (2007). Contingent manpower planning in a high clock speed industry. International journal of production research, 45(9), 2051-2072.

Bokhorst, J. A. C. (2011). The impact of the amount of work in process on the use of cross-training. International Journal of Production Research, 49(11), 3171-3190.

Bokhorst, J. A. C., Slomp, J., & Molleman, E. (2004). Development and evaluation of cross-training policies for manufacturing teams. IIE Transactions, 36(10), 969-984.

Brandt, L., & Fife, G. (1997, September). CTRAIN. A cross-training concept. In Advanced Semiconductor Manufacturing Conference and Workshop, 1997. IEEE/SEMI (pp. 341-343). IEEE.

Brusco, M. J. (2008). An exact algorithm for a workforce allocation problem with application to an analysis of cross-training policies. [Article]. IIE Transactions, 40(5), 495-508.

Brusco, M. J., & Johns, T. R. (1996). Staffing a multi-skilled workforce with varying levels of productivity: An analysis of cross-training policies. Paper presented at the Proceedings of the 1996 27th Annual Meeting of the Decision Sciences Institute. Part 2 (of 3), November 24, 1996 - November 26, 1996, Orlando, FL, USA.

Burns, R. N., Narasimhan, R., & Smith, L. D. (1998). A set-processing algorithm for scheduling staff on 4-day or 3-day work weeks. Naval Research Logistics (NRL), 45(8), 839-853.

Campbell, G. M. (1999). Cross-utilization of workers whose capabilities differ. Management Science, 45(5), 722-732.

Celayix (2013). Collaborative scheduling: revolutionary yet undervalued. Retrieved from: http://www.celayix.com/wpcontent/uploads/01_Collaborative_Scheduling_Revolutionary_yet_Undervalued_-_White_Paper_2.pdf

Ernst, A. T., Jiang, H., Krishnamoorthy, M., and Sier, D. (2004). Staff scheduling and rostering: A review of applications, methods and models. European journal of Operational research, 153(1), 3-27.

Gomar, J. E., Haas, C. T., & Morton, D. P. (2002). Assignment and allocation optimization of partially multiskilled workforce. Journal of construction Engineering and Management, 128(2), 103-109.

Hopp, W. J., Tekin, E., & Van Oyen, M. P. (2004). Benefits of skill chaining in serial production lines with cross-trained workers. Management Science, 50(1), 83-98.

Hopp, W. J., & Van Oyen, M. P. (2004). Agile workforce evaluation: A framework for cross-training and coordination. IIE Transactions (Institute of Industrial Engineers), 36(10), 919-940.

Jarugumilli, S. (2011). Integrated Workforce Planning Considering Regular and Overtime Decisions. Proceedings of the 2011 Industrial Engineering Research Conference (IERC), Reno, NV.

Jarugumilli, S., & Grasman, S. E. (2010). Single Shift Worker Allocation Using Two-Phase Goal Programming. Proceedings of the 4th Annual ISC Research Symposium. ISCRS 2010, Rolla, Missouri.

Jordan, W. C., & Graves, S. C. (1995). Principles on the benefits of manufacturing process flexibility. Management Science, 41(4), 577-594.

Kulkarni, A., Grasman, S., Jarugumilli, S., and Keng, N. (2013). Workforce planning and Facility utilization using two-stage stochastic recourse approach. Proceedings of the 2013 Industrial Engineering Research Conference (IERC), San Juan, Puerto Rico

Kum, K. Y. (2007). A comparison of cross-training policies in different job shops. International Journal of Production Research, 45(6), 1279-1295.

Kuo, Y., & Yang, T. (2006). A case study on the operator allocation decision for TFT-LCD inspection and packaging process. Journal of Manufacturing Technology Management, 17(3), 363-375.

Nembhard, D. A. (Ed.). (2007). Workforce cross training. CRC Press.

Chicago Norman, B. A., Tharmmaphornphilas, W., Needy, K. L., Bidanda, B., & Warner, R. C. (2002). Worker assignment in cellular manufacturing considering technical and human skills. International Journal of Production Research, 40(6), 1479-1492.

Oldham, G. R., & Hackman, J. R. (1981). Relationships between organizational structure and employee reactions: Comparing alternative frameworks. Administrative Science Quarterly, 66-83.

Slomp, J., Bokhorst, J. A. C., & Molleman, E. (2005). Cross-training in a cellular manufacturing environment. Computers & Industrial Engineering, 48(3), 609-624.

Slomp, J., & Molleman, E. (2002). Cross-training policies and team performance. International Journal of Production Research, 40(5), 1193-1219.

Subramanian, D., and An, L. (2008, July). Optimal Resource Action Planning Analytics for Services Delivery Using Hiring, Contracting and Cross-Training of Various Skills. In Services Computing, 2008. SCC'08. IEEE International Conference on (Vol. 1, pp. 433-440). IEEE.

Suer, G. A., & Tummaluri, R. R. (2008). Multi-period operator assignment considering skills, learning and forgetting in labour-intensive cells. Paper presented at the Selected papers from the

2nd Group Technology/Cellular Manufacturing World Symposium, 4 Park Square, Milton Park, Abingdon, Oxfordshire, OX14 4RN, United Kingdom.

Süer, G. A., & Tummaluri, R. R. (2008). Multi-period operator assignment considering skills, learning and forgetting in labour-intensive cells. International journal of production research, 46(2), 469-493.

Wang, J. (2005). A Review of Operations Research Applications in Workforce Planning and Potential Modeling of Military Training (No. DSTO-TR-1688).

APPENDIX: Description of the C# Code

The C# code is provided in the CD associated with this thesis. The code has been developed in the .NET framework. It is comprised of three components, the Data Interface, the Model solver and the Program. These are described below.

Data Interface:

This is a set of classes that are created to store the different parameters to be kept track of from the input excel sheet. The files associated with the data interface are:

- CostFactorList.cs: The CostFactorList class is defined here. This class stores the cost parameters for regular allocation, i.e., it stores the costs associated with doing work on different skill levels.
- Costs.cs: The Costs class stores the values of cost of training, cost of overtime, cost of slack and surplus in cross-training, and whether an operation is a critical operation or not.
- CostList.cs: This class creates a list which contains the elements stored in the Costs class.
- HCInfo.cs: Each regular worker and its designated shift type is stored in the HCInfo class.
- HC_CertList.cs: The HC_CertList class stores the elements of a worker-qualification matrix in a dictionary.
- HC_Info_List.cs: The HC_Info_List class stores the regular workers and shift involved in each shift type.
- Misc.cs: The Misc class stores values such as the minimum allocation that a worker assignment needs to be given for, the limit on overtime and cross-training for each shift and for the complete factory, the number of allocations that a worker can be given in a

shift, and the maximum/minimum number of qualifications a worker can possess at the same time.

- MiscList.cs: The MiscList class creates a list of all the information stored in the Misc class.
- Operation.cs: The Operation class stores each operation and whether it is a critical operation or not.
- OperationList.cs: The OperationList class creates a list of the information stored in the Operation class.
- PreAllocTraining.cs: The PreAllocTraining class stores all the pre-allocations specified at the beginning of the planning period.
- PreAllocWorkMap.cs: The PreAllocaWorkMap class stores all the partial allocations at the beginning of the planning period.
- Shifts-ShiftType.cs: The Shifts-ShiftType class stores the shift numbers of the shifts present in each shift type.
- Shifts-ShiftTypeList.cs: This class stores the information in class Shift-ShiftType in a list.
- Skill_Levels.cs: This class stores the skill levels present in the test case, and the number of shifts needed to complete training on that skill level.
- Skill_LevelsList.cs: This class stores the information present in the Skill_Levels class in a list.
- Tool-Operation Matrix.cs: This class stores the individual elements of the requirements such as MMR, L1:L2:L3 and Tool Count.

- Tool-Operation MatrixList.cs: This class stores the information present in Tool-Operation Matrix class in a list.
- TrainingNeeded.cs: The TrainingNeeded class stores the number of shifts of training needed on each operation skill level combination.
- Training_Needed_List: This class stores the information present in the TrainingNeeded class in the form of a list.

Model Solver (Model.cs and ReadTable.cs):

The model solver contains the optimization model implemented using the CPLEX C# API. ILOG Concert and ILOG CPLEX are given as references to this file. A class called Model is created which holds all the variables and parameters. A function called createConstraints() is defined with Individual constraints being framed within the createConstraints() function. This function also holds the objective function and includes commands for CPLEX parameterization. All the requisite data tables are created in Model.cs, and the output excel file is developed using the ExcelLibrary namespace. This file also contains a function called InitializeVariables(), that initializes every variable and creates the dictionaries needed to store their values.

ReadTable.cs contains the ReadTable class that actually reads the input excel sheet into the model solver.

Program (Program.cs):

The Program.cs file is a short run script that calls an object of the ReadTable class, an object of the Model class, initializes variables, creates constraints and displays the output.