Two-Stage Risk Adjusted Cross-Training Stochastic Programming Model for Improving Daily Operations Under Staff Absenteeism

by

Dania Ammar

**A thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science
(Industrial and Systems Engineering)
in the University of Michigan-Dearborn
2019**

Master's Thesis Committee:

**Associate Professor Jian Hu, Chair
Professor Armen Zakarian
Shanshan Qiu, Ford Motor Company**

Dedication

To my supportive parents who instilled in me the desire of learning and working hard…

Table of Contents

List of Tables

List of Figures

Abstract

During daily operation, workers are assigned to jobs on an assembly line. This assignment requires that each worker be skillfully qualified to operate the position he is appointed to. Nonetheless, several circumstances may arise at time of operation, of which workers' absenteeism is considered most critical. Absenteeism leads to a deficiency in the daily available skills keeping some jobs at the mercy of being filled by unqualified workers. To compensate for deficiencies in the obtainable skills under the uncertainty of absenteeism, managements have recourse to cross-training its workers. However, cross-training is costly and requires strategic decisions to identify key regions, workers, and jobs within the assembly line that are to be involved with the cross-training plan while coping with uncertainties. The aim of this study is to develop a cross-training process that resolves the issue of the uncertain daily shortage in skills while preserving the workplace requirements. It presents a two-stage stochastic cross-training model in which specific workers are suggested to be trained to specific jobs for the purpose of maximizing the future expected assignment of skillful workers to jobs under workers' indefinite inclination to absenteeism and under the varying level of this imposed risk. Research is presented on an assembly line that integrates cross-training plan in its labor-force planning. Results illustrate a reasonable improvement in the level of positions' occupancy varying according to available resources, target degree of protection against absenteeism and selective parameters that guide the assignment process.

Chapter 1 Introduction

Cross-training is the act of training a person at work to carry out new skills besides the person's present skills. It has become a common practice in different disciplines including manufacturing processes and service systems. It ensures the acquisition of each of the required skills of a certain work environment by more than one person to maintain the flow of its work and to fight the threat of being short on a certain skill during daily operation. This threat is the result of different means of uncertainty including work-demand fluctuations in addition to labor-behavior patterns driven by absenteeism, planned and unplanned occurrences.

In 2006, it was estimated that the cost associated with absenteeism in the United States to be as high as $118 billion (Weaver, 2006). This triggers researchers to analyze the different causes of absenteeism. Personal factors such as age, gender, place of residency, marital status, number of kids, and health condition are primary variables that drive absenteeism (Harrison & Martocchio, 1998). Another significant contributor to absenteeism is the weather status, especially when it comes to areas characterized by extreme cold and high precipitation (U.S Bureau of Labor Statistics, 2012). Further unscheduled absences have been attributed to illnesses, family issues such as problematic relationships, family obligations, emergencies, personal needs, and stress (Navarro & Bass, 2006; Prater & Smith, 2011). Other factors leading to absenteeism are associated with the working environment itself including lack of involvement, absence of recognition and motivation, unfair management, work safety issues, and inflexible shift scheduling (Navarro & Bass, 2006). Moreover, the type of responsibility held at work is another absenteeism determining factor (Harrison & Martocchio, 1998).

To reduce absenteeism, many companies have established incentive and goal-setting programs, childcare support facilities, positive workspace atmosphere, workers' assistantship programs, and some disciplinary proceedings in an attempt to increase workers' accountability for their absences (Kocakülâh, Kelley, Mitchell, & Ruggieri, 2011).

While many have focused on the causes of absenteeism and on means for reducing its occurrences, very limited research has focused on quantifying absenteeism consequences and on solving them. Industries, in particular, have concluded that 15% of payroll constitute the direct and indirect cost of workers absenteeism. Navarro and Bass (2006) concluded that each employee absenteeism instance costs the workplace an average of $660 in terms of paid unproductive time in addition to other indirect costs such as losses in productivity, quality and customer service. Blumenfeld and Inman (2009) concluded that quality defects of produced goods increase with absenteeism. A key point in this situation is that one instance of absence calls for the immediate replacement of the absentee making absenteeism a critical issue to solve.

In workplaces that highly depend on mechanized production forms, absence is compensated through integrating highly sophisticated technologies to decrease the dependence on manual work and thus decrease the dependence on workforce (Mateo, 2007). Other workplaces tend to invest in costly cross-training programs for their workers to become multi-skilled (Blumenfeld & Inman, 2009).

Since the daily distribution of work among workers depends on a combination of factors including number of available workers and positions, workers' acquired skills, workers' attendance status, workers' preferred positions, and positions' importance and complexity, then a well-planned cross-training strategy incorporating all these concerns should be carried out.

Chapter 2 Literature Review

While many employers resort to reserve pools of extra people when incurring shortages in uncovered duties, others adopt the notion of developing a multi-skilled worker by exploiting the potentials of existing workers. This idea of establishing a multi-skilled worker also helps in reducing the degree of boredom when performing a highly repetitive job and enhances innovation (W.J. Hopp & Spearman, 2011, pp. 163–165). However, full cross-training is not desirable as it is extremely costly and it decreases productivity. Further limitations are related to workers' efficiency and learning and forgetting aspects (Molleman & Slomp, 1999). Thus, carrying out a cross-training plan should not be a haphazard process but rather a well-planned strategy with underlying goals.

We will proceed to a thorough review on different workforce cross-training policies and models published earlier in literature.

2.1 Cross-training

Studies on cross-training are primarily focused on evaluating certain cross-training policies and on developing mathematical cross-training models.

Inman, Jordan, and Blumenfeld (2004) evaluated the performance of chaining, being a practical principle in cross-training, in the presence of different scenarios of unanticipated absences at an assembly line. In this study, chaining corresponds to training one worker from each section to only one task in the section located downstream. The research concludes to the effectiveness of chaining compared to multiple other cross-training policies, including (1) no cross-training between sections, (2) full cross-training, (3) all-for-one, (4) one-for-all, (5) reciprocal pairs, and (6) all-to-all in terms of reliability achieved per each cross-training opportunity proposed by each policy. Hopp, Tekin, and Oyen (2004) have also found out that skill chaining is more effective, robust, and flexible compared to another capacity-balancing approach called cherry picking. Iravani, Kolfal, and Oyen (2007) evaluated the use of the average shortest path length metric APL of a small world network SWN under three considerations of training limited budget being the financial aspect, the practical considerations for time required for training, and the stress from handling multiple jobs. The study, nevertheless, focused on proving the effectiveness of APL in service systems only and not in manufacturing systems.

A handful number of studies have focused on mathematical programming models to achieve optimal cross-trainings. Askin and Huang (2001) generated a worker assignment and training mixed integer goal programming model. The objective function has multiple purposes with the intention of establishing an interactive team that properly matches workers' abilities to task requirements while minimizing the training cost**.** Bokhorst, Slomp, and Molleman (2004) developed an integer goal programming model that aims at minimizing the deviation from optimal cross-training configurations represented by the deviation from suggested additional number of cross training, from equal multifunctionality, from equal machine coverage, from collective responsibility (i.e. deviation from obtaining balanced workload among all workers), and from equal working responsibilities. Another integer programming model that integrates cross-training along with shift scheduling, days-off scheduling, and breaks scheduling at call centers is discussed by Taskiran and Zhang (2017). The study focuses on two cross-training aspects being the optimal portion of employees to be cross-trained and the optimal number of skills attributed to each employee. These aspects are driven by the financial limitations and by the impact on secondary skills' efficiency. Yet, the model does not take into account stochastic demand fluctuations**.** In this sense, Slomp et al. (2005) developed an integer programming model that selects optimal workers to be cross-trained in a cellular manufacturing environment. The mathematical model presents a tradeoff between cross-training costs and operating costs. However, only two deterministic scenarios of demand (low or high) have been considered. Wirojanagud, Gel, Fowler and Cardy (2007) also developed a mixed integer programming model to determine the optimal number of workers to be fired, hired, and cross-trained in order to minimize associated costs accompanied with costs of uncovered production. The focus of this research is the inclusion of individuals' difference in general cognitive abilities translated through their productivity. Nevertheless, no other uncertain parameters have been considered. In this concern, Araz and Fowler (2008) developed a two stage stochastic integer programming model for cross-training workers in a wafer fab. The sources of uncertainty considered are product demand, workers' productivity, new workers' hiring and training costs, and current workers' cross-training and firing costs, all that vary according to workforce market. Billionnett (1999) formulated an integer program that saves workers' cost while creating the optimal hierarchical workforce. Seckinar, Gokcen, and Kurt (2007) extended Billionet's integer programming model to multiple shifts.

2.2 Research Gap and Contributions

Even though these studies have formulated models aiming at reducing cross-training costs and assessing the effectiveness of heuristic cross-training policies, none has developed a cross-training model that aims at achieving optimal daily work assignment in an assembly line while considering workers' characteristics represented by their unique skills and stochastic individual absenteeism pattern and while giving the flexibility to select different cross-training options that serve for different risk levels.

For the purpose of closing this gap, this paper presents the first two-stage stochastic cross-training programming model that attempts to (1) achieve optimal daily operation characterized by maximizing the assignment of a skilled worker to corresponding job under the risk of staff absenteeism, to (2) maintain desired effectiveness and quality, to (3) provide different cross-training options pertained to different risk levels, and to (4) ensure the optimal distribution of limited cross-training resources among different zones. It reports results of research performed on a local assembly line illustrating the application of this cross-training model.

The paper proceeds as follows: Section 3 tackles the problem description along with model key elements and assumptions. Section 4 discusses the solution approach while section 5 describes a case study on an assembly line located in the United States and discusses the accompanying results. Finally, conclusions are presented.

Chapter 3 Problem Description

When a workplace suffers from daily absence instances, a shortage in the combination of available skills occurs. As a result, several jobs lack a specialized worker capable of carrying work independently. In workplaces characterized by their serial flow of work such as an assembly line, the occupancy of all workstations is mandatory to ensure the continuity of work. Therefore, the management is coerced to daily fill up the remaining unoccupied positions by assigning one unspecialized worker or even sometimes by assigning two workers who are not qualified to do the job each on his own. However, this type of single or doubled-up assignment has its different drawbacks. One drawback is the decrease in quality delivered and thus the increase in rework instances. Another drawback is the slowing down of the assembly line and thus the decrease in productivity and its accompanying economic loses such as missing out on product delivery deadlines. In some cases, workers from distinct shifts are urged to work for overtime hours to cover absences, and these substitutions are usually accompanied by high replacement costs along with fatigue issues. Double-ups also require the availability of a large number of workers increasing the cost of daily operation. These facts trigger the management to develop a cross-training plan that would increase daily one-to-one specialized job occupancy in the presence of absenteeism through qualifying certain workers to do certain jobs independently as a function of multiple factors. Yet, workers' absenteeism is characterized by its undetermined and stochastic happenings thus rendering the cross-training model to a stochastic programming model in nature. This foreshadows for a two-stage stochastic operation necessary to implement the cross-training plan. The first stage is basically the training stage that is responsible for generating the new optimal skill combination. Periodically, the management allocates some budget to cross-training sufficient to train some workers to some jobs. The management ordinarily negotiates the goal behind cross-training as to whether it should defend against skills' shortage during overall days in general or during inferior days recognizing extreme shortage in skills. Depending on the determined need, more than one worker could be required to learn a particular job or even one worker could be required to learn more than one job. Nevertheless, it is the workers' responsibility to preserve his acquired skills while developing the tendency to learning new skills. There also exist some unique jobs that need the fulfillment of exceptional requirements to be promoted to them. Such requirements apply to positions such as the team leader and tag relief positions that are not allocated any specific machine or station in principle. During the working shift, each worker normally visits restrooms and goes on a lunch break leaving his job unoccupied for some time. However, workflow should be maintained necessitating the presence of a backup person, called a tag relief, who relieves each other worker and replaces them in such instances. Consequently, a tag relief must be knowledgeable of all other jobs as he occupies each of them for some time during the day in addition to being recognized by his fast changeover. In other type of instances such as emergent cases characterized by severe deficiency of people and unavailability of enough people to double up, the team leader, besides his managerial skills, is forced to substitute any vacant position to protect against line slowing or shutting down. Therefore, the team leader should possess skills needed to run all other jobs. Furthermore, his comprehensive knowledge is important to ensure that each person within his team is performing his job correctly and to supervise cross-training thus deciding whether any newly trained worker has become eligible to autonomously do a certain job.

Nonetheless, the here and now cross-training decision taken during the first stage has its significant impact on daily assignment of work; in its ideal case, it should attempt to minimize the expected cost of the assigning stage, being the second stage, translating into the number of unoccupied jobs remaining after daily assignment of one worker to one well-known job. In this regards, workers' daily attendance directs the distribution of work among workers: each day, a certain number of workers show up to work after which every worker is assigned to one job which he knows how to do on his own. Part of this workers' knowledge to jobs is determined by the newly introduced cross-trainings. Now, even though different possible combinations of worker-job assignment may exist based on workers who show up (eg worker1-job1 and worker2-job2 vs worker1-job2 and worker2-job1) yielding the fact that a job can be carried out by more than one trained and available worker, the team leader may sometimes fix his selection to a specific worker to do a specific job on daily basis on the expense of all other workers and even sometimes on the expense of overall occupancy. What happens in this case is that the team leader might have developed some preference to a person to occupy a certain position because that worker has been doing this job for so long and has shown maximum effectiveness while doing it. At the same time, that worker might have preferred that particular job and might have felt most comfortable at running. The presence of such an assignment preference necessitates the construction of a multi-objective optimization model explicated by using the Chebyshev technique capable of handling the two goal constituents together, one being the overall job occupancy and the other being the prioritized assignment.

Both the training and assigning stages are applied within all sections of an assembly line. One common practice is distributing the cross-training budget offered by management among sections based on a deliberate need while allowing a worker to be trained to jobs found in the section that he belongs to only. This allows the one-to-one assignment of workers to jobs within their own section solely. Later, each unassigned worker is called by his unique section team leader along with another unassigned co-worker to fill up one job that ended up with no assigned specialized worker in that section. Workers who remain unassigned even after this doubling up procedure takes place are all gathered from the multiple sections in the assembly line. They wait in a reserve pool to help run the remaining unoccupied jobs in sections that have not been fully replenished even after exploiting their own workers through pairing up. Therefore, the above description distinguished with the stochastic happenings of events implies the essentials for developing a two-stage stochastic cross-training operation model.

Upon constructing a cross-training model, the workplace would be able to determine the best worker(s) that needs to be trained to the riskiest job(s) in the most insecure section(s). The factors affecting the choice are (1) the number of workers in each section, (2) the number of jobs in each section, (3) the attendance history of each worker, (4) the number of jobs known by each worker, (5) the number of workers capable of doing each job, (6) the number of available cross-training opportunities, and (7) the priority job for each worker.

3.1 Formulation

A two-stage risk-adjusted stochastic programming model is constructed and consists of two stages: (1) training stage and (2) assigning stage. The aim of this research is to help an assembly line to develop an effective cross-training plan for all its sections: recommending what employees be trained to what jobs within each section, in an attempt to increase the future expectancy of the daily number of occupied jobs by skilled workers while taking into account workers' uncertain inclination to absenteeism.

The two-stage model supplies an assembly line with all sections' updated skill matrices decided on during the first stage. The set of output skill matrices delivered in the first stage has its own impact on daily assignment taking place in the second stage in each section which is also at the mercy of its workers absenteeism. This set of resultant updated skill matrices keeps adjusting until maximum intended improvement in overall assembly line's functionality in terms of job occupancy is achieved taking into consideration the set of all possible attendance scenarios that have occurred in each section.

3.1.1 Notations

Sets and indices:

K	set of all sections $k \in K$, $K = \{1,2, 3 \ldots D\}$
I	set of all workers in the whole assembly line $i \in I$
I_k	set of all workers Section k such that $I_k \subset I$
J	set of all jobs in all sections $j \in J$, $J = \{1,2, 3 \ldots N\}$
J_k	set of all jobs $j \in J$ at Section k such that $J_k \subset J$, $J_k = \{1,2, 3 \ldots N^k\}$
J'	set of station-based jobs such that $J' \subset J$, $J' = \{1,2,3, \ldots S\}$
J'_k	set of station-based jobs at Section k such that $J_k' \subset J'$, $J_k' = \{1,2,3 \ldots S^k\}$
$\overline{J'}$	set of non-station-based jobs such that $\overline{J'} \subset J$, $\overline{J'} = \{TR, TL\}$
$\overline{J'_k}$	set of non-station-based jobs at Section k such that $\overline{J'_k} \subset \overline{J'}$, $\overline{J'_k} = \{TR_k, TL_k\}$

Parameters:

$V_{i,j}^k$	1 if worker i is able to do job j in Section k, 0 otherwise; where V is current skill matrix
$P_{i,j}^k$	1 if worker i has a priority job j in Section k, 0 otherwise; where P is priority matrix
$A_i^k(\omega)$	1 if worker i shows up to Section k, 0 otherwise; where A(ω) is stochastic attendance matrix
w_1	weight of objective function associated with overall assignment
w_2	weight of objective function associated with priority job assignment
α	risk- adjusting coefficient/ penalty cost associated with low-job occupancy days
B	total number of available cross-training opportunities
N, N^k	total number of jobs, total number of jobs in Section k

S, S^k	total number of stations, total number of stations in Section k
E, E^k	total number of workers, total number of workers in Section k
Q, Q^k	total number of priority-based jobs, total number of priority-based jobs in Section k
D	total number of sections in factory

Decision variables:

$x_{i,j}^k(\omega)$ 1 if worker i is assigned to work at job j at Section k,0 otherwise; where x(ω) is the assignment matrix

$u_{i,j}^k$ 1 if worker i is able to do job j in Section k,0 otherwise; where u is the updated skill matrix

The proposed cross-training model is formulated as a two-stage stochastic integer program:

$$Min_{u \in U} \ E\{g(u, \omega)\} \quad (1)$$

s.t.

$$\sum_{i \in I_k} \sum_{j \in J_k} \sum_{k \in K} (u_{i,j}^k - V_{i,j}^k) \leq B \quad (1a)$$

$$u_{i,j}^k \geq V_{i,j}^k, \quad \forall\, i \epsilon I_k, \quad \forall\, j \epsilon J_k, \quad \forall\, k \epsilon K \quad (1b)$$

$$u_{i,j}^k - \frac{\sum_{m \in J_k'} u_{i,j}^k}{S^k} \leq 0, \quad \forall\, i \epsilon I_k, \quad \forall\, n \epsilon \overline{J_k'}, \quad \forall\, k \epsilon K \quad (1c)$$

$$u_{i,j}^k \ binary, \quad \forall\, i \epsilon I_k, \quad \forall\, j \epsilon J_k, \quad \forall\, k \epsilon K \quad (1d)$$

$$g(u, \omega) = Min \ \sum_{k \in K} w_1 (N^k - \sum_{i \in I_k} \sum_{j \in J_k} x_{i,j}^k(\omega))^\alpha + \sum_{k \in K} w_2 \sum_{j \in J_k} \left(Q^k - \sum_{i \in I_k} P_{i,j}^k x_{i,j}^k(\omega)\right)^\alpha \quad (2)$$

s.t.

$$\sum_{j \in J_k} x_{i,j}^k(\omega) \leq A_i^k(\omega), \quad \forall\, i \in I_k, \quad \forall\, k \in K \quad (2a)$$

$$x_{i,j}^k(\omega) \leq u_{i,j}^k, \quad \forall\, i \in I_k, \quad \forall\, j \in J_k, \quad \forall\, k \in K \quad (2b)$$

$$\sum_{i \in I_k} x_{i,j}^k(\omega) \leq 1, \quad \forall\, j \in J_k, \quad \forall\, k \in K \quad (2c)$$

$$x_{i,j}^k(\omega) binary \quad \forall i \in I_k, \quad \forall j \in J_k, \quad \forall\, k \in K \quad (2d)$$

The function g(u, ω) in equation (1) resembles the cost incurred due to assigning workers to jobs according to the updated skill matrix u in the different set of scenarios ω. It is composed of three different parts as explicit in equation (2). The first term is responsible for the overall assignment of workers to jobs aiming to minimize the expected number of unoccupied jobs in all sections and among the overall stochastic scenarios. The significant feature in this two-stage stochastic model is the term α which importance will be examined in subsequent sections. The second term is responsible for minimizing the expected number of non-priority job assignment among the overall stochastic scenarios in all sections. Upper management can decide on the relative importance of the two above terms and manipulate their associated weights w_1 and w_2 accordingly thus inducing a tradeoff between overall and priority assignment.

The rules that govern the generation of the new distribution of workers' skills in the first stage are represented by constraints 1a, 1b, 1c, and 1d. Constraint 1a ensures that the number of the additional cross-training positions suggested by the outcome solution must not exceed the total number of available cross-training opportunities B provided by the management for all sections combined. Constraint 1b allows each worker to either adhere to his current available skill or to learn some new skills. Under this constraint, one worker is allowed to end up being trained to more than one new job such that there are at least 2 available cross-training opportunities. Equally plausible is the recommendation of cross-training more than one worker to the same job. Constraint 1c is the team leader specific and tag relief specific constraint. It prohibits training any worker for the team leader and tag relief positions until he satisfies the requirement of being trained for all other station-based jobs. Finally, constraint 1d requires that skill matrix elements be binary.

Constraints 2a, 2b, 2c, and 2d guide the daily assignment process. Constraint 2a is attendance-dependent assignment constraint such that any worker is allowed to be assigned to solely one job only if he shows up to work. Constraint 2b is skill-dependent assignment constraint in which a worker is allowed to be assigned to a job only if he has the current skill of that job or if he is to be cross-trained for that job. One-to-one assignment is represented by constraint 2c such that each job is to be run by one worker only. Finally, Constraint 2d requires that each assignment matrix element is either 0 or 1.

3.1.2 Importance of α

It is a risk adjusting continuous coefficient used as a penalty cost associated with the number of unoccupied jobs.

First, it aims at distinguishing the stochastic daily scenarios with high job-occupancy from those characterized by relatively low job-occupancy. In general, a workplace is expected to realize more daily scenarios with low absenteeism rate and thus a larger combination of available skills translating into a considerable frequency of high job-occupancy instances. The number of instances of meeting daily scenarios with high absenteeism rate characterized by severe shortage in the combination of available skills is very minimal with respect to scenarios with low absenteeism rate. Therefore, α is intended to give more weight for such low job-occupancy instances so that they would contribute to determining the optimal cross-training plan. Nevertheless, it is the management responsibility to decide on their target goal; that is, either choose low α for the sake of determining cross-training suggestions that protect all days in general or choose higher α for the sake of determining cross-training suggestions that are more in favor of protecting low-job occupancy days.

Another function for α is also to distinguish between different sections. The number of available cross-training resources are common across all sections and thus need to be distributed accordingly. These sections do also differ in the overwhelming trend of job occupancy: some sections are more prone to experiencing daily shortage in the combination of available skills due to its riskier pattern of workers' absenteeism, while some other sections experience a higher level of functionality due to relatively lower absenteeism occurrences. Consequently, the choice of α determines the distribution of available cross-training resources among the different sections such that it gives more weight to the protection of relatively riskier sections leading to the generation of a fair solution that allocates more cross-training suggestions to these sections rather than other sections. This reflects the balancing function of α as it reduces job unoccupancy in the section that experiences it most in an attempt to achieve the least average power-raised number of unoccupied jobs across all sections resembling the minimization of mean squared error. Having said that, α prevents the allocation of cross-training opportunities and thus the exploitation of skills in one section; it rather distributes the opportunities among sections to ensure a balanced enhancement of all sections.

Besides, the integration of this term α in both terms of the objective function is a fulfillment to the structure of Chebyshev scalarization for multi-objective optimization.

Chapter 4 Solution Approach

The constructed optimization model is a non-linear integer programming model due to the presence of the power function in the objective function rendering it non-linear and due to the binary requirement of all decision variables $x_{i,j}^k$ and $u_{i,j}^k$.

4.1 Reformulation

The linearization of this non-linear model requires the reformulation of its objective function. This is achieved through the introduction of intermediate variables and through the inclusion of a plus function as shown in the transformation below:

$$(N^k - \sum_{i \in I_k} \sum_{j \in J_k} x_{i,j}^k(\omega))^\alpha$$

$$= c_1(N^k - \sum_{i \in I_k} \sum_{j \in J_k} x_{i,j}^k(\omega)) \; + \; c_2^\alpha \left(N^k - \sum_{i \in I_k} \sum_{j \in J_k} x_{i,j}^k(\omega) - 1\right)_+ \; + \; c_3^\alpha \left(N^k - \sum_{i \in I_k} \sum_{j \in J_k} x_{i,j}^k(\omega) - 2\right)_+ + \cdots + c_{N^k-1}^\alpha \left(N^k - \sum_{i \in I_k} \sum_{j \in J_k} x_{i,j}^k(\omega) - (N^k - 1)\right)_+$$

Similarly,

$$(Q^k - \sum_{i\in I_k}\sum_{j\in J_k} P_{i,j}^k x_{i,j}^k(\omega))^\alpha$$

$$= c_1(Q^k - \sum_{i\in I_k}\sum_{j\in J_k} P_{i,j}^k x_{i,j}^k(\omega)) \; + \; c_2^\alpha\big(Q^k - \sum_{i\in I_k}\sum_{j\in J_k} P_{i,j}^k x_{i,j}^k(\omega) - 1\big)_+ \; + \; c_3^\alpha\big(Q^k - \sum_{i\in I_k}\sum_{j\in J_k} P_{i,j}^k x_{i,j}^k(\omega) - 2\big)_+ + \cdots + c_{Q^k-1}^\alpha\big(Q^k - \sum_{i\in I_k}\sum_{j\in J_k} P_{i,j}^k x_{i,j}^k(\omega) - (Q^k - 1)\big)_+$$

Where $c_1 = 1$ for all α and $c_2^\alpha, c_3^\alpha, \ldots c_{PN^k-1}^\alpha, \ldots c_{Q^k-1}^\alpha \geq 0$ are coefficients that are unique for each α.

4.2 Lagrangian Relaxation

The formulization of the first stage model constraints includes one hard constraint that binds all sections k together, being constraint (1a). To solve the complexity introduced by this constraint, we implement the Lagrangian relaxation approach in which constraint (1a) is relaxed through attaching it to one Lagrangian multiplier λ and through moving it to the objective function. This relaxation is described below:

Z=min cx		Z (λ)= min cx+ λ (Dx-e) where λ≥0
S.T. Ax=b		S.T. Ax=b
Dx≤e	➞	x≥0
x≥0		x is binary
x is binary		

Then, the Lagrangian dual of the two-stage stochastic programming model is presented below where λ≥0:

$$Z(\lambda) = Min_{u\in U} \; E\{g(u,\omega)\} \; + \; \lambda \left(\sum_{i\in I_k}\sum_{j\in J_k}\sum_{k\in K}(u_{i,j}^k - V_{i,j}^k) - B\right) \tag{3}$$

s.t.

$$u_{i,j}^k \geq V_{i,j}^k, \quad \forall\, i\epsilon I_k, \quad \forall\, j\epsilon J_k, \quad \forall\, k\epsilon K \tag{3a}$$

$$u^k_{i,j} - \frac{\sum_{m\in J'_k} u^k_{i,j}}{S^k} \leq 0, \quad \forall\ i\epsilon I_k, \quad \forall\ n\epsilon \overline{J'_k}, \quad \forall\ k\epsilon K \tag{3b}$$

$$u^k_{i,j}\ binary, \quad \forall\ i\epsilon I_k, \quad \forall\ j\epsilon J_k, \quad \forall\ k\epsilon K \tag{3c}$$

$$g(u,\omega) = Min \sum_{k\in K} w_1 [\ c_1 (N^k - \sum_{i\in I_k} \sum_{j\in J_k} x^k_{i,j}(\omega))\ +\ c^\alpha_2 \left(N^k - \sum_{i\in I_k} \sum_{j\in J_k} x^k_{i,j}(\omega) - 1\right)_+ + \cdots + c^\alpha_{N^k-1} \left(N^k - \sum_{i\in I_k} \sum_{j\in J_k} x^k_{i,j}(\omega) - \left(N^k - 1\right)\right)_+] + \sum_{k\in K} w_2 [\ c_1 (Q^k - \sum_{i\in I_k} \sum_{j\in J_k} P^k_{i,j} x^k_{i,j}(\omega))\ + c^\alpha_2 \left(Q^k - \sum_{i\in I_k} \sum_{j\in J_k} P^k_{i,j} x^k_{i,j}(\omega) - 1\right)_+ + \cdots + c^\alpha_{Q^k-1} \left(Q^k - \sum_{i\in I_k} \sum_{j\in J_k} P^k_{i,j} x^k_{i,j}(\omega) - \left(Q^k - 1\right)\right)_+] \tag{4}$$

s.t.

$$\sum_{j\in J_k} x^k_{i,j}(\omega) \leq A^k_i(\omega), \quad \forall\ i \in I_k, \quad \forall\ k \in K \tag{4a}$$

$$x^k_{i,j}(\omega) \leq u^k_{i,j}, \quad \forall\ i \in I_k, \quad \forall\ j \in J_k, \quad \forall\ k \in K \tag{4b}$$

$$\sum_{i\in I_k} x^k_{i,j}(\omega) \leq 1, \quad \forall\ j \in J_k, \quad \forall\ k \in K \tag{4c}$$

$$x^k_{i,j}(\omega) binary \quad \forall i \in I_k, \quad \forall j \in J_k, \quad \forall\ k \in K \tag{4d}$$

In this case, the optimal value of Z(λ) becomes a lower bound to Z since we are dealing with a minimization problem. The main interest becomes in finding the optimal Lagrangian multiplier λ that would maximize Z(λ) which is concaved on λ (Fisher, 2004). This search could be done through different techniques one of which is the Golden section search method. The algorithm explaining this method is shown in the chart below.

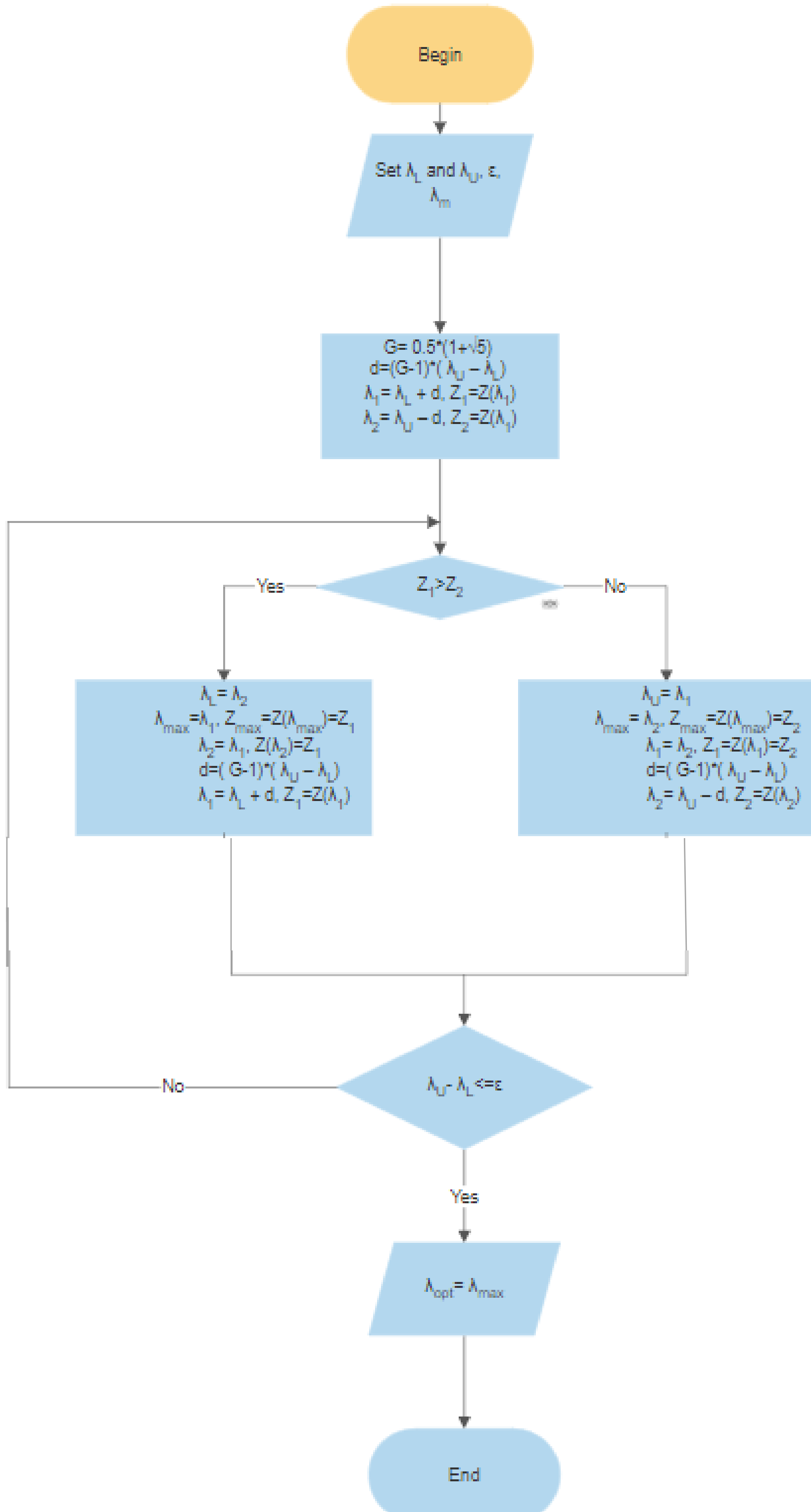

Figure 1: Golden Section Search Method

Chapter 5 Case Study

In this paper, one full manufacturing assembly line is put under study. This line has different shifts with distinct staff working for each shift. Only one staff is considered for our study. Furthermore, the line is composed of 7 different sections, such that each section has a specific number of jobs and a specific number of workers under the following conditions:

- Each section comprises of a number of workers that exceed the number of jobs N^k; that is $E^k > N^k$.
- Each section consists of N^k jobs out of which only S^k jobs have a designated physical station.
- The remaining N^k-S^k jobs in each section are basically 2 positions (team leader and tag relief positions) that are not station-based.
- By definition, a tag relief is the worker who relieves each other worker in a certain section while the latter is on his daily allocated break. Workers in one section go on break in a sequential manner so that each station is temporarily held by the tag relief one at a time.

In total, there are 100 workers and 78 jobs distributed uniquely among each section. Each worker is initially knowledgeable of a different set of jobs in his corresponding section denoted as Origin Section.

Moreover, filling all jobs happens in two steps practiced each day: in the first step, the assignment of each worker to one job, which he is trained for, takes place inside his Original Section only. In the second step, remaining unassigned jobs in each section adhere to the double-up policy which consists of occupying each unassigned job remaining from step 1 by any two workers who are not knowledgeable of this job but who belong to this section. If a particular section witnesses a shortage in the number of workers needed for double-up, it can borrow floating workers from other sections. Those floating workers are the surplus workers remaining from other sections after one-to-one assignments and pairing ups. Other workers could be borrowed from other lines as needed.

5.1 Data Collection

In order to run the formulated cross-training model, skill matrices characterizing each section are collected. Each of this matrix shows the list of jobs and list of workers relative to that section. It also shows the set of current skills acquired by each worker within his Original Section. Additionally, workers' preferred position within his Original Section designated by a priority matrix is collected from team leaders. For instance, the skill matrix in Figure 2 corresponds to one of the sections (Section 1) that consists of 13 jobs and 15 workers such that Jobs 12 and 13 are the tag relief and the team leader jobs respectively. Worker 13 is knowledgeable of Jobs 1 and 6. However, the associated priority matrix in Figure 3 indicates that he is preferred to be assigned to Job 6 leaving him as a primary person for that job.

	Job1	Job2	Job3	Job4	Job5	Job6	Job7	Job8	Job9	Job10	Job11	Job12*	Job13**
Worker1	0	0	0	0	0	0	0	0	1	0	0	0	0
Worker2	0	0	0	0	0	0	0	0	0	0	1	0	0
Worker3	0	1	0	0	0	0	0	0	0	0	0	0	0
Worker4	1	0	0	1	1	0	0	1	0	0	0	0	0
Worker5	0	0	0	1	0	0	0	0	0	0	0	0	0
Worker6	1	1	1	1	1	0	1	0	0	1	1	0	0
Worker7	0	0	0	0	0	0	0	0	0	1	0	0	0
Worker8	0	0	0	0	0	0	1	0	0	0	0	0	0
Worker9	0	0	1	0	0	0	0	0	0	0	0	0	0
Worker10	1	0	0	0	0	0	0	0	0	0	0	0	0
Worker11	0	0	0	0	0	1	0	0	0	0	0	0	0
Worker12	0	0	0	0	1	0	0	0	0	0	0	0	0
Worker13	1	0	0	0	0	1	0	0	0	0	0	0	0
Worker14	1	1	1	1	1	1	1	1	1	1	1	1	1
Worker15	1	1	1	1	1	1	1	1	1	1	1	1	1

Figure 2: Skill matrix for Section 1

*Tag Relief Job

**Team Leader Job

	Job1	Job2	Job3	Job4	Job5	Job6	Job7	Job8	Job9	Job10	Job11	Job12*	Job13**
Worker1	0	0	0	0	0	0	0	0	1	0	0	0	0
Worker2	0	0	0	0	0	0	0	0	0	0	1	0	0
Worker3	0	1	0	0	0	0	0	0	0	0	0	0	0
Worker4	0	0	0	0	0	0	0	1	0	0	0	0	0
Worker5	0	0	0	1	0	0	0	0	0	0	0	0	0
Worker6	0	0	0	0	0	0	0	0	0	0	0	0	0
Worker7	0	0	0	0	0	0	0	0	0	1	0	0	0
Worker8	0	0	0	0	0	0	1	0	0	0	0	0	0
Worker9	0	0	1	0	0	0	0	0	0	0	0	0	0
Worker10	1	0	0	0	0	0	0	0	0	0	0	0	0
Worker11	0	0	0	0	0	1	0	0	0	0	0	0	0
Worker12	0	0	0	0	1	0	0	0	0	0	0	0	0
Worker13	0	0	0	0	0	1	0	0	0	0	0	0	0
Worker14	0	0	0	0	0	0	0	0	0	0	0	1	0
Worker15	0	0	0	0	0	0	0	0	0	0	0	0	1

Figure 3: Priority matrix for Section 1

Historical attendance data is another essential component of the cross-training model to resemble the stochastic daily scenarios of workers. For this purpose, attendance data for each worker for three consecutive years corresponding to 2016, 2017, and 2018 has been extracted. The attendance data shows huge uncertainty in worker's behavior creating a serious problem of absenteeism as shown in Figure 4. A considerable portion of workers is absent for more than 12% of the time.

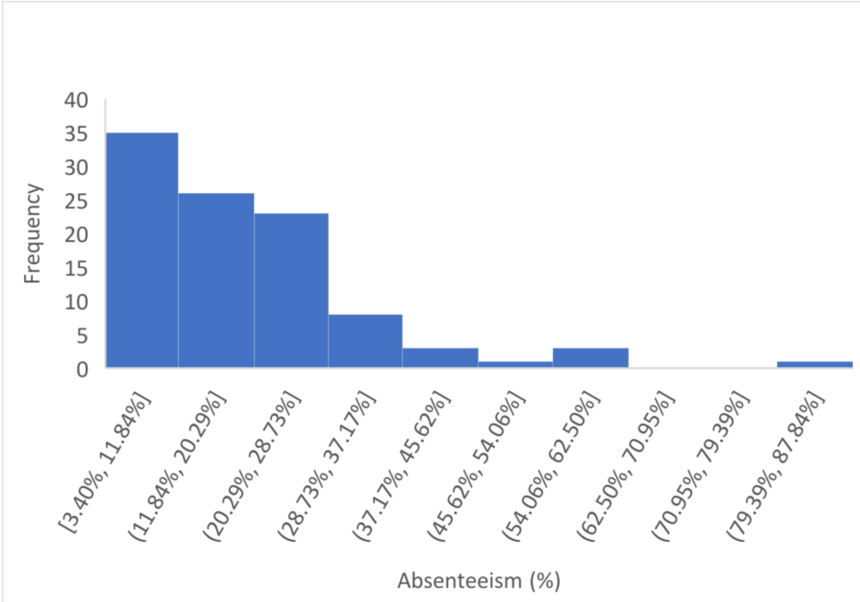

Figure 4: Distribution of Workers' Absenteeism

5.2 Results

Different values of the risk-adjustment coefficient α, the total number of available cross-training opportunities B, and the weights corresponding to overall and priority assignments w_1 and w_2 respectively have been tested.

Three governing combinations of w_1 and w_2 have been considered for this case study: $(w_1,w_2)=(1,0)$, $(w_1,w_2)=(0.5,0.5)$, and $(w_1,w_2)=(0.1,0.9)$ in which the first combination emphasizes overall assignment only, the second combination presents a trade-off between overall assignment and priority-based assignment, whereas the third one intensifies priority assignment.

The results below demonstrate (1) the number of unoccupied jobs depicting those that remain after one-to-one assignment takes place and (2) the number of non-priority assignment obtained after both one-to-one and paired-up assignment.

Combination 1 $(w_1, w_2) = (1,0)$:

For the first combination of w_1 and w_2, the outcomes are compared with initial distribution of daily scenarios among the number of unoccupied jobs before providing any opportunity for training, which is independent of α and B. This initial distribution is based on maximizing overall job occupancy and is shown in Table 1. It indicates that the assembly line fully serves all jobs using one-to-one specialized assignment during a total of 1722 scenarios distributed across all 7 sections. It also shows that the maximum number of unoccupied jobs that have been incurred during the 3 years data is 7, taking place during 4 daily scenarios in section 6. The second maximum number of unoccupied jobs is 5 which has been incurred twice in section 1 and once in section 4. These instances with such a relatively high number of unoccupied jobs resemble a portion of worst-case scenarios that are less likely to occur than normal daily scenarios characterized by a relatively low number of unoccupied jobs.

Table 1: Initial Distribution of Daily Scenarios among the number of unoccupied jobs for w_1=1, w_2=0

Num. of Unoccupied Jobs / Section	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
1	195	213	103	39	7	2	0	0
2	200	218	100	29	12	0	0	0
3	285	188	61	20	5	0	0	0
4	154	235	122	34	13	1	0	0
5	291	160	86	20	2	0	0	0
6	363	156	32	4	0	0	0	4
7	234	222	88	14	1	0	0	0
Total	1722	1392	592	160	40	3	0	4

Upon inputting the model with different combinations of α and B, different cross-training recommendation output is available for each case. Table 2 summarizes the resultant allocation of cross-training resources among the 7 different sections for all (α, β) combinations tested.

Table 2: Distribution of cross-training opportunities among sections for different combinations of parameters

Section / Parameters	1	2	3	4	5	6	7
α=1, B=2				1			1
α=2, B=2		1		1			
α=4, B=2		1				1	
α=6, B=2		1				1	
α=1, B=3		1		1			1
α=2, B=3		1		1			1
α=4, B=3		1		1		1	
α=6, B=3	1	1				1	
α=1, B=4		1	1	1			1
α=2, B=4		1	1	1			1
α=4, B=4	1	1		1		1	
α=6, B=4	1	1		1		1	
α=1, B=8		1	3	1		1	1
α=6, B=8	1	1	1	1	1	1	1

To demonstrate the output of running the two-stage stochastic programming model across the assembly line, a deeper look into the combination (α, B)=(4,2) is considered: in the presence of 2 cross-training opportunities and a risk adjusting coefficient of 4, the model recommends that one cross-training opportunity be given to Section 2 and the remaining other resource be given to Section 6. More specifically, the model advises that Worker 8 be cross-trained to Job 7 in Section 2 and Worker 13 be cross-trained to Job 3 in Section 6 as shown in Figures 5-8.

Abs Rate		Job1	Job2	Job3	Job4	Job5	Job6	Job7	Job8	Job9	Job10	Job11	Job12*	Job13**
15.38%	Worker1	0	0	0	1	0	0	0	0	0	0	0	0	0
25.22%	Worker2	0	0	0	0	0	0	0	0	0	1	1	0	0
20.04%	Worker3	0	1	0	1	0	1	0	1	1	0	0	0	0
24.15%	Worker4	0	0	0	0	0	1	0	0	0	0	0	0	0
9.48%	Worker5	0	0	1	0	0	0	0	0	0	0	0	0	0
23.26%	Worker6	1	0	1	0	0	0	0	1	0	0	0	0	0
18.96%	Worker7	1	1	1	1	1	1	1	1	1	1	1	1	1
7.69%	Worker8	0	0	0	0	0	1	0	0	0	0	0	0	0
15.56%	Worker9	1	1	1	1	1	1	1	1	1	1	1	1	1
34.17%	Worker10	0	0	0	0	0	0	0	0	0	1	1	0	0
15.74%	Worker11	1	0	1	0	1	0	1	0	0	1	1	0	0
7.16%	Worker12	0	0	0	0	1	0	0	0	0	0	0	0	0
22.36%	Worker13	0	1	0	0	0	0	0	0	0	0	0	0	0
23.26%	Worker14	1	1	0	0	0	0	0	0	1	0	0	0	0
26.12%	Worker15	0	0	0	0	0	0	0	0	1	0	0	0	0
20.21%	Worker16	1	0	1	0	0	1	0	0	0	1	1	0	0
46.15%	Worker17	0	0	0	0	0	0	0	0	0	0	1	0	0

Figure 5: Input/ Initial Skill Matrix of Section 2

Abs Rate		Job1	Job2	Job3	Job4	Job5	Job6	Job7	Job8	Job9	Job10	Job11	Job12*	Job13**
15.38%	Worker1	0	0	0	1	0	0	0	0	0	0	0	0	0
25.22%	Worker2	0	0	0	0	0	0	0	0	0	1	1	0	0
20.04%	Worker3	0	1	0	1	0	1	0	1	1	0	0	0	0
24.15%	Worker4	0	0	0	0	0	1	0	0	0	0	0	0	0
9.48%	Worker5	0	0	1	0	0	0	0	0	0	0	0	0	0
23.26%	Worker6	1	0	1	0	0	0	0	1	0	0	0	0	0
18.96%	Worker7	1	1	1	1	1	1	1	1	1	1	1	1	1
7.69%	Worker8	0	0	0	0	0	1	1	0	0	0	0	0	0
15.56%	Worker9	1	1	1	1	1	1	1	1	1	1	1	1	1
34.17%	Worker10	0	0	0	0	0	0	0	0	0	1	1	0	0
15.74%	Worker11	1	0	1	0	1	0	1	0	0	1	1	0	0
7.16%	Worker12	0	0	0	0	1	0	0	0	0	0	0	0	0
22.36%	Worker13	0	1	0	0	0	0	0	0	0	0	0	0	0
23.26%	Worker14	1	1	0	0	0	0	0	0	1	0	0	0	0
26.12%	Worker15	0	0	0	0	0	0	0	0	1	0	0	0	0
20.21%	Worker16	1	0	1	0	0	1	0	0	0	1	1	0	0
46.15%	Worker17	0	0	0	0	0	0	0	0	0	0	1	0	0

Figure 6: Output/ Updated Skill Matrix of Section 2 for (α, B) = (4,2)

Abs Rate		Job1	Job2	Job3	Job4	Job5	Job6	Job7	Job8	Job9	Job10*	Job11**
22.00%	Worker1	1	0	0	0	0	0	0	0	0	0	0
22.72%	Worker2	0	0	1	0	0	0	0	0	0	0	0
25.76%	Worker3	0	0	0	1	0	0	0	1	1	0	0
16.82%	Worker4	0	0	0	0	0	1	0	0	0	0	0
5.90%	Worker5	0	0	0	0	1	0	0	0	0	0	0
30.23%	Worker6	0	0	0	1	0	0	0	0	0	0	0
3.40%	Worker7	0	1	0	0	0	0	0	0	0	0	0
40.25%	Worker8	0	0	0	1	0	0	0	0	0	0	0
4.47%	Worker9	1	1	1	1	1	1	1	1	1	1	1
6.62%	Worker10	1	1	1	1	1	1	1	1	1	1	0
29.16%	Worker11	0	0	0	0	0	0	0	1	1	0	0
7.69%	Worker12	0	0	0	0	0	0	1	0	0	0	0
4.47%	Worker13	0	0	0	0	0	0	0	1	1	0	0
13.06%	Worker14	1	1	1	1	1	1	1	0	0	0	0
9.66%	Worker15	0	0	0	0	0	0	0	1	1	0	0

Figure 7: Input/ Initial Skill Matrix of Section 6

Abs Rate		Job1	Job2	Job3	Job4	Job5	Job6	Job7	Job8	Job9	Job10*	Job11**
22.00%	Worker1	1	0	0	0	0	0	0	0	0	0	0
22.72%	Worker2	0	0	1	0	0	0	0	0	0	0	0
25.76%	Worker3	0	0	0	1	0	0	0	1	1	0	0
16.82%	Worker4	0	0	0	0	0	1	0	0	0	0	0
5.90%	Worker5	0	0	0	0	1	0	0	0	0	0	0
30.23%	Worker6	0	0	0	1	0	0	0	0	0	0	0
3.40%	Worker7	0	1	0	0	0	0	0	0	0	0	0
40.25%	Worker8	0	0	0	1	0	0	0	0	0	0	0
4.47%	Worker9	1	1	1	1	1	1	1	1	1	1	1
6.62%	Worker10	1	1	1	1	1	1	1	1	1	1	0
29.16%	Worker11	0	0	0	0	0	0	0	1	1	0	0
7.69%	Worker12	0	0	0	0	0	0	1	0	0	0	0
4.47%	Worker13	0	0	1	0	0	0	0	1	1	0	0
13.06%	Worker14	1	1	1	1	1	1	1	0	0	0	0
9.66%	Worker15	0	0	0	0	0	0	0	1	1	0	0

Figure 8: Output/ Updated Skill Matrix of Section 6 for (α, B) = (4,2)

Furthermore, Tables 3-16 show the distribution of daily scenarios for each section among the number of overall unoccupied jobs as a function of the different combinations and their corresponding resultant updated skill matrices model output.

Table 3: Distribution of Daily Scenarios among the number of unoccupied jobs for w_1=1, w_2=0, B=2, α=1

Section \ Number of Unoccupied jobs	0	1	2	3	4	5	6	7
1	195	213	103	39	7	2	0	0
2	200	218	100	29	12	0	0	0
3	285	188	61	20	5	0	0	0
4*	270	175	74	31	9	0	0	0
5	291	160	86	20	2	0	0	0
6	363	156	32	4	0	0	0	4
7*	378	129	41	11	0	0	0	0
Total	1982	1239	497	154	35	2	0	4

Table 4: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=1, w2=0, B=2, α=2

Section \ Number of Unoccupied Jobs	0	1	2	3	4	5	6	7
1	195	213	103	39	7	2	0	0
2*	279	202	58	16	4	0	0	0
3	285	188	61	20	5	0	0	0
4*	270	175	74	31	9	0	0	0
5	291	160	86	20	2	0	0	0
6	363	156	32	4	0	0	0	4
7	234	222	88	14	1	0	0	0
Total	1917	1316	502	144	28	2	0	4

Table 5: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=1, w2=0, B=2, α=4

Section \ Number of Unoccupied Jobs	0	1	2	3	4	5	6	7
1	195	213	103	39	7	2	0	0
2*	279	202	58	16	4	0	0	0
3	285	188	61	20	5	0	0	0
4	154	235	122	34	13	1	0	0
5	291	160	86	20	2	0	0	0
6*	408	125	19	3	0	0	4	0
7	234	222	88	14	1	0	0	0
Total	1846	1345	537	146	32	3	4	0

Table 6: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=1, w2=0, B=2, α=6

Section \ Number of Unoccupied Jobs	0	1	2	3	4	5	6	7
1	195	213	103	39	7	2	0	0
2*	279	202	58	16	4	0	0	0
3	285	188	61	20	5	0	0	0
4	154	235	122	34	13	1	0	0
5	291	160	86	20	2	0	0	0
6*	408	125	19	3	0	0	4	0
7	234	222	88	14	1	0	0	0
Total	1846	1345	537	146	32	3	4	0

Table 7: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=1, w2=0, B=3, α=1

Section \ Number of Unoccupied Jobs	0	1	2	3	4	5	6	7
1	195	213	103	39	7	2	0	0
2*	279	202	58	16	4	0	0	0
3	285	188	61	20	5	0	0	0
4*	270	175	74	31	9	0	0	0
5	291	160	86	20	2	0	0	0
6	363	156	32	4	0	0	0	4
7*	378	129	41	11	0	0	0	0
Total	2061	1223	455	141	27	2	0	4

Table 8: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=1, w2=0, B=3, α=2

Section \ Number of Unoccupied Jobs	0	1	2	3	4	5	6	7
1	195	213	103	39	7	2	0	0
2*	279	202	58	16	4	0	0	0
3	285	188	61	20	5	0	0	0
4*	270	175	74	31	9	0	0	0
5	291	160	86	20	2	0	0	0
6	363	156	32	4	0	0	0	4
7*	375	133	41	10	0	0	0	0
Total	2058	1227	455	140	27	2	0	4

Table 9: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=1, w2=0, B=3, α=4

Section \ Number of Unoccupied Jobs	0	1	2	3	4	5	6	7
1	195	213	103	39	7	2	0	0
2*	279	202	58	16	4	0	0	0
3	285	188	61	20	5	0	0	0
4*	270	175	74	31	9	0	0	0
5	291	160	86	20	2	0	0	0
6*	408	125	19	3	0	0	4	0
7	234	222	88	14	1	0	0	0
Total	1962	1285	489	143	28	2	4	0

Table 10: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=1, w2=0, B=3, α=6

Section \ Number of Unoccupied Jobs	0	1	2	3	4	5	6	7
1*	213	206	105	29	6	0	0	0
2*	279	202	58	16	4	0	0	0
3	285	188	61	20	5	0	0	0
4	154	235	122	34	13	1	0	0
5	291	160	86	20	2	0	0	0
6*	408	125	19	3	0	0	4	0
7	234	222	88	14	1	0	0	0
Total	1864	1338	539	136	31	1	4	0

Table 11: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=1, w2=0, B=4, α=1

Section \ Number of Unoccupied Jobs	0	1	2	3	4	5	6	7
1	195	213	103	39	7	2	0	0
2*	279	202	58	16	4	0	0	0
3*	320	187	39	10	3	0	0	0
4*	270	175	74	31	9	0	0	0
5	291	160	86	20	2	0	0	0
6	363	156	32	4	0	0	0	4
7*	378	129	41	11	0	0	0	0
Total	2096	1222	433	131	25	2	0	4

Table 12: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=1, w2=0, B=4, α=2

Section \ Number of Unoccupied Jobs	0	1	2	3	4	5	6	7
1	195	213	103	39	7	2	0	0
2*	279	202	58	16	4	0	0	0
3*	320	187	39	10	3	0	0	0
4*	270	175	74	31	9	0	0	0
5	291	160	86	20	2	0	0	0
6	363	156	32	4	0	0	0	4
7*	375	133	41	10	0	0	0	0
Total	2093	1226	433	130	25	2	0	4

Table 13: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=1, w2=0, B=4. α=4

Section \ Number of Unoccupied Jobs	0	1	2	3	4	5	6	7
1*	213	206	105	29	6	0	0	0
2*	279	202	58	16	4	0	0	0
3	285	188	61	20	5	0	0	0
4*	270	175	74	31	9	0	0	0
5	291	160	86	20	2	0	0	0
6*	408	125	19	3	0	0	4	0
7	234	222	88	14	1	0	0	0
Total	1980	1278	491	133	27	0	4	0

Table 14: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=1, w2=0, B=4, α=6

Section \ Number of Unoccupied Jobs	0	1	2	3	4	5	6	7
1*	213	206	105	29	6	0	0	0
2*	279	202	58	16	4	0	0	0
3	285	188	61	20	5	0	0	0
4*	270	175	74	31	9	0	0	0
5	291	160	86	20	2	0	0	0
6*	408	125	19	3	0	0	4	0
7	234	222	88	14	1	0	0	0
Total	1980	1278	491	133	27	0	4	0

Table 15: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=1, w2=0, B=8, α=1

Section \ Number of Unoccupied Jobs	0	1	2	3	4	5	6	7
1*	235	192	91	35	5	1	0	0
2*	279	202	58	16	4	0	0	0
3***	418	107	23	8	3	0	0	0
4*	270	175	74	31	9	0	0	0
5	291	160	86	20	2	0	0	0
6*	408	125	19	3	0	0	4	0
7*	378	129	41	11	0	0	0	0
Total	2279	1090	392	124	23	1	4	0

Table 16: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=1, w2=0, B=8, α=6

Section \ Number of Unoccupied Jobs	0	1	2	3	4	5	6	7
1**	254	183	92	25	5	0	0	0
2*	279	202	58	16	4	0	0	0
3*	320	187	39	10	3	0	0	0
4*	270	175	74	31	9	0	0	0
5*	308	156	81	12	2	0	0	0
6*	408	125	19	3	0	0	4	0
7*	375	133	41	10	0	0	0	0
Total	2214	1161	404	107	23	0	4	0

As a further matter, Tables 17-29 show the distribution of scenarios among number of non-priority assigned jobs resulting after the attempt to minimize overall number of unoccupied jobs.

Table 17: Initial Distribution of Daily Scenarios among the number of non-priority assigned jobs for w_1=1, w_2=0

Section \ Number of non-priority assignment	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	143	173	116	66	15	4	0	1
3	35	148	178	136	46	13	3	0	0
4	110	227	153	53	14	2	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	219	119	43	9	0	0	0	0
Total	810	1360	1025	493	174	38	8	4	1

Table 18: Distribution of Daily Scenarios among the number of non-priority assignment for w1=1, w2=0, B=2, α=1

Section \ Number of non-priority assignment	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	143	173	116	66	15	4	0	1
3	35	148	178	136	46	13	3	0	0
4	110	114	219	94	18	4	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	102	65	140	67	15	1	0	0
Total	810	1130	1037	631	236	55	9	4	1

Table 19: Distribution of Daily Scenarios among the number of non-priority assignment for w1=1, w2=0, B=2, α=2

Section \ Number of non-priority assignment	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	127	128	93	81	47	34	5	3
3	35	148	178	136	46	13	3	0	0
4	110	114	219	94	18	4	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	219	119	43	9	0	0	0	0
Total	810	1231	1046	511	193	72	38	9	3

Table 20: Distribution of Daily Scenarios among the number of non-priority assignment for w1=1, w2=0, B=2, α=4

Section \ Number of non-priority assignment	0	1	2	3	4	5	6	7	8	9
1	78	183	180	81	29	7	1	0	0	0
2	41	127	128	93	81	47	34	5	3	0
3	35	148	178	136	46	13	3	0	0	0
4	110	227	153	53	14	2	0	0	0	0
5	159	215	132	46	6	1	0	0	0	0
6	218	183	82	58	13	1	0	0	0	4
7	169	219	119	43	9	0	0	0	0	0
Total	810	1302	972	510	198	71	38	5	3	4

Table 21: Distribution of Daily Scenarios among the number of non-priority assignment for w1=1, w2=0, B=2, α=6

Section \ Number of non-priority assignment	0	1	2	3	4	5	6	7	8	9
1	78	183	180	81	29	7	1	0	0	0
2	41	127	128	93	81	47	34	5	3	0
3	35	148	178	136	46	13	3	0	0	0
4	110	227	153	53	14	2	0	0	0	0
5	159	215	132	46	6	1	0	0	0	0
6	218	183	82	58	13	1	0	0	0	4
7	169	219	119	43	9	0	0	0	0	0
Total	810	1302	972	510	198	71	38	5	3	4

Table 22: Distribution of Daily Scenarios among the number of non-priority assignment for w1=1, w2=0, B=3, α=1

Section \ Number of non-priority assignment	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	127	128	93	81	47	34	5	3
3	35	148	178	136	46	13	3	0	0
4	110	114	219	94	18	4	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	102	65	140	67	15	1	0	0
Total	810	1114	992	608	251	87	39	9	3

Table 23: Distribution of Daily Scenarios among the number of non-priority assignment for w1=1, w2=0, B=3, α=2

Section \ Number of non-priority assignment	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	127	128	93	81	47	34	5	3
3	35	148	178	136	46	13	3	0	0
4	110	114	219	94	18	4	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	122	155	86	25	2	0	0	0
Total	810	1134	1082	554	209	74	38	9	3

Table 24: Distribution of Daily Scenarios among the number of non-priority assignment for w1=1, w2=0, B=3, α=4

Section \ Number of non-priority assignment	0	1	2	3	4	5	6	7	8	9
1	78	183	180	81	29	7	1	0	0	0
2	41	127	128	93	81	47	34	5	3	0
3	35	148	178	136	46	13	3	0	0	0
4	110	114	219	94	18	4	0	0	0	0
5	159	215	132	46	6	1	0	0	0	0
6	218	183	82	58	13	1	0	0	0	4
7	169	219	119	43	9	0	0	0	0	0
Total	810	1189	1038	551	202	73	38	5	3	4

Table 25: Distribution of Daily Scenarios among the number of non-priority assignment for w1=1, w2=0, B=3, α=6

Section \ Number of non-priority assignment	0	1	2	3	4	5	6	7	8	9
1	78	177	171	79	26	16	10	2	0	0
2	41	127	128	93	81	47	34	5	3	0
3	35	148	178	136	46	13	3	0	0	0
4	110	227	153	53	14	2	0	0	0	0
5	159	215	132	46	6	1	0	0	0	0
6	218	183	82	58	13	1	0	0	0	4
7	169	219	119	43	9	0	0	0	0	0
Total	810	1296	963	508	195	80	47	7	3	4

Table 26: Distribution of Daily Scenarios among the number of non-priority assignment for w1=1, w2=0, B=4, α=1

Section \ Number of non-priority assignment	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	127	128	93	81	47	34	5	3
3	35	145	162	106	42	44	20	5	0
4	110	114	219	94	18	4	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	102	65	140	67	15	1	0	0
Total	810	1111	976	578	247	118	56	14	3

Table 27: Distribution of Daily Scenarios among the number of non-priority assignment for w1=1, w2=0, B=4, α=2

Section \ Number of non-priority assignment	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	127	128	93	81	47	34	5	3
3	35	145	162	106	42	44	20	5	0
4	110	114	219	94	18	4	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	122	155	86	25	2	0	0	0
Total	810	1131	1066	524	205	105	55	14	3

Table 28: Distribution of Daily Scenarios among the number of non-priority assignment for w1=1, w2=0, B=4, α=4

Section \ Number of non-priority assignment	0	1	2	3	4	5	6	7	8	9
1	78	177	171	79	26	16	10	2	0	0
2	41	127	128	93	81	47	34	5	3	0
3	35	148	178	136	46	13	3	0	0	0
4	110	114	219	94	18	4	0	0	0	0
5	159	215	132	46	6	1	0	0	0	0
6	218	183	82	58	13	1	0	0	0	4
7	169	219	119	43	9	0	0	0	0	0
Total	810	1183	1029	549	199	82	47	7	3	4

Table 29: Distribution of Daily Scenarios among the number of non-priority assignment for w1=1, w2=0, B=4, α=6

Section \ Number of non-priority assignment	0	1	2	3	4	5	6	7	8	9
1	78	177	171	79	26	16	10	2	0	0
2	41	127	128	93	81	47	34	5	3	0
3	35	148	178	136	46	13	3	0	0	0
4	110	114	219	94	18	4	0	0	0	0
5	159	215	132	46	6	1	0	0	0	0
6	218	183	82	58	13	1	0	0	0	4
7	169	219	119	43	9	0	0	0	0	0
Total	810	1183	1029	549	199	82	47	7	3	4

Combination 2 (w_1, w_2) = (0.5,0.5):

For the second combination, in which equal weights are given for priority assignment and overall assignment (w1,w2)=(0.5,0.5), Tables 30-53 display the distribution of the number of unoccupied jobs along with the number of non-priority assignment upon providing different number of cross-training opportunities B and different risk-adjusting coefficient α.

Table 30: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=0.5, w2=0.5, B=2, α=1

Section \ Number of Unoccupied Jobs	0	1	2	3	4	5	6	7
1	195	213	103	39	7	2	0	0
2*	227	208	92	25	7	0	0	0
3*	312	177	55	11	4	0	0	0
4	154	235	122	34	13	1	0	0
5	291	160	86	20	2	0	0	0
6	363	156	32	4	0	0	0	4
7	234	222	88	14	1	0	0	0
Total	1776	1371	578	147	34	3	0	4

Table 31: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=0.5, w2=0.5, B=2, α=2

Section \ Number of Unoccupied Jobs	0	1	2	3	4	5	6	7
1	195	213	103	39	7	2	0	0
2*	218	213	92	27	9	0	0	0
3*	312	177	55	11	4	0	0	0
4	154	235	122	34	13	1	0	0
5	291	160	86	20	2	0	0	0
6	363	156	32	4	0	0	0	4
7	234	222	88	14	1	0	0	0
Total	1767	1376	578	149	36	3	0	4

Table 32: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=0.5, w2=0.5, B=2, α=4

Section \ **Number of Unoccupied Jobs**	0	1	2	3	4	5	6	7
1*	210	214	101	26	8	0	0	0
2	189	219	107	31	13	0	0	0
3	274	191	68	21	5	0	0	0
4	154	235	122	34	13	1	0	0
5	291	160	86	20	2	0	0	0
6*	391	142	19	3	0	0	4	0
7	234	222	88	14	1	0	0	0
Total	1743	1383	591	149	42	1	4	0

Table 33: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=0.5, w2=0.5, B=2, α=6

Section \ **Number of Unoccupied Jobs**	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
1*	210	214	101	26	8	0	0	0
2	189	219	107	31	13	0	0	0
3	274	191	68	21	5	0	0	0
4	154	235	122	34	13	1	0	0
5	291	160	86	20	2	0	0	0
6*	391	142	19	3	0	0	4	0
7	234	222	88	14	1	0	0	0
Total	1743	1383	591	149	42	1	4	0

Table 34: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=0.5, w2=0.5, B=3, α=1

Section \ **Number of Unoccupied Jobs**	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
1	195	213	103	39	7	2	0	0
2*	227	208	92	25	7	0	0	0
3*	312	177	55	11	4	0	0	0
4	154	235	122	34	13	1	0	0
5	291	160	86	20	2	0	0	0
6	363	156	32	4	0	0	0	4
7*	272	204	71	12	0	0	0	0
Total	1814	1353	561	145	33	3	0	4

Table 35: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=0.5, w2=0.5, B=3, α=2

Section \ **Number of Unoccupied Jobs**	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**
1*	218	210	95	28	7	1	0	0
2*	218	213	92	27	9	0	0	0
3*	312	177	55	11	4	0	0	0
4	154	235	122	34	13	1	0	0
5	291	160	86	20	2	0	0	0
6	363	156	32	4	0	0	0	4
7	234	222	88	14	1	0	0	0
Total	1790	1373	570	138	36	2	0	4

Table 36: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=0.5, w2=0.5, B=3, α=4

Section \ Number of Unoccupied Jobs	0	1	2	3	4	5	6	7
1*	210	214	101	26	8	0	0	0
2*	199	220	105	29	6	0	0	0
3	274	191	68	21	5	0	0	0
4	154	235	122	34	13	1	0	0
5	291	160	86	20	2	0	0	0
6*	391	142	19	3	0	0	4	0
7	234	222	88	14	1	0	0	0
Total	1753	1384	589	147	35	1	4	0

Table 37: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=0.5, w2=0.5, B=3, α=6

Section \ Number of Unoccupied Jobs	0	1	2	3	4	5	6	7
1*	210	214	101	26	8	0	0	0
2*	199	220	105	29	6	0	0	0
3	274	191	68	21	5	0	0	0
4	154	235	122	34	13	1	0	0
5	291	160	86	20	2	0	0	0
6*	391	142	19	3	0	0	4	0
7	234	222	88	14	1	0	0	0
Total	1753	1384	589	147	35	1	4	0

Table 38: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=0.5, w2=0.5, B=4, α=1

Section \ Number of Unoccupied Jobs	0	1	2	3	4	5	6	7
1*	218	210	95	28	7	1	0	0
2*	227	208	92	25	7	0	0	0
3*	312	177	55	11	4	0	0	0
4	154	235	122	34	13	1	0	0
5	291	160	86	20	2	0	0	0
6	363	156	32	4	0	0	0	4
7*	272	204	71	12	0	0	0	0
Total	1837	1350	553	134	33	2	0	4

Table 39: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=0.5, w2=0.5, B=4, α=2

Section \ Number of Unoccupied Jobs	0	1	2	3	4	5	6	7
1*	218	210	95	28	7	1	0	0
2*	218	213	92	27	9	0	0	0
3*	312	177	55	11	4	0	0	0
4*	174	235	107	33	10	0	0	0
5	291	160	86	20	2	0	0	0
6	363	156	32	4	0	0	0	4
7	234	222	88	14	1	0	0	0
Total	1810	1373	555	137	33	1	0	4

Table 40: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=0.5, w2=0.5, B=4, α=4

Number of Unoccupied Jobs / Section	0	1	2	3	4	5	6	7
1*	210	214	101	26	8	0	0	0
2*	199	220	105	29	6	0	0	0
3	274	191	68	21	5	0	0	0
4*	174	235	107	33	10	0	0	0
5	291	160	86	20	2	0	0	0
6*	391	142	19	3	0	0	4	0
7	234	222	88	14	1	0	0	0
Total	1773	1384	574	146	32	0	4	0

Table 41: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=0.5, w2=0.5, B=4, α=6

Number of Unoccupied Jobs / Section	0	1	2	3	4	5	6	7
1*	210	214	101	26	8	0	0	0
2*	199	220	105	29	6	0	0	0
3	274	191	68	21	5	0	0	0
4*	174	235	107	33	10	0	0	0
5	291	160	86	20	2	0	0	0
6*	391	142	19	3	0	0	4	0
7	234	222	88	14	1	0	0	0
Total	1773	1384	574	146	32	0	4	0

Table 42: Distribution of Daily Scenarios among the number of non-priority assignment for w_1=0.5, w_2=0.5, B=2, α=1

Number of non-priority assignment / Section	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	148	172	120	57	15	4	1	1
3	35	151	189	124	45	12	3	0	0
4	110	227	153	53	14	2	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	219	119	43	9	0	0	0	0
Total	810	1368	1035	485	164	37	8	5	1

Table 43: Distribution of Daily Scenarios among the number of non-priority assignment for w_1=0.5, w_2=0.5, B=2, α=2

Number of non-priority assignment / Section	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	151	176	117	56	13	4	0	1
3	35	151	189	124	45	12	3	0	0
4	110	227	153	53	14	2	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	219	119	43	9	0	0	0	0
Total	810	1371	1039	482	163	35	8	4	1

Table 44: Distribution of Daily Scenarios among the number of non-priority assignment for w_1=0.5, w_2=0.5, B=2, α=4

Section \ Number of non-priority assignment	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	151	176	117	56	13	4	0	1
3	35	151	189	124	45	12	3	0	0
4	110	227	153	53	14	2	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	219	119	43	9	0	0	0	0
Total	810	1371	1039	482	163	35	8	4	1

Table 45: Distribution of Daily Scenarios among the number of non-priority assignment for w_1=0.5, w_2=0.5, B=2, α=6

Section \ Number of non-priority assignment	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	151	176	117	56	13	4	0	1
3	35	151	189	124	45	12	3	0	0
4	110	227	153	53	14	2	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	219	119	43	9	0	0	0	0
Total	810	1371	1039	482	163	35	8	4	1

Table 46: Distribution of Daily Scenarios among the number of non-priority assignment for w_1=0.5, w_2=0.5, B=3, α=1

Section \ Number of non-priority assignment	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	148	172	120	57	15	4	1	1
3	35	151	189	124	45	12	3	0	0
4	110	227	153	53	14	2	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	219	119	43	9	0	0	0	0
Total	810	1368	1035	485	164	37	8	5	1

Table 47: Distribution of Daily Scenarios among the number of non-priority assignment for w_1=0.5, w_2=0.5, B=3, α=2

Section \ Number of non-priority assignment	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	151	176	117	56	13	4	0	1
3	35	151	189	124	45	12	3	0	0
4	110	227	153	53	14	2	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	219	119	43	9	0	0	0	0
Total	810	1371	1039	482	163	35	8	4	1

Table 48: Distribution of Daily Scenarios among the number of non-priority assignment for w_1=0.5, w_2=0.5, B=3, α=4

Section \ Number of non-priority assignment	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	151	176	117	56	13	4	0	1
3	35	151	189	124	45	12	3	0	0
4	110	227	153	53	14	2	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	219	119	43	9	0	0	0	0
Total	810	1371	1039	482	163	35	8	4	1

Table 49: Distribution of Daily Scenarios among the number of non-priority assignment for w_1=0.5, w_2=0.5, B=3, α=6

Section \ Number of non-priority assignment	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	151	176	117	56	13	4	0	1
3	35	151	189	124	45	12	3	0	0
4	110	227	153	53	14	2	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	219	119	43	9	0	0	0	0
Total	810	1371	1039	482	163	35	8	4	1

Table 50: Distribution of Daily Scenarios among the number of non-priority assignment for w_1=0.5, w_2=0.5, B=4, α=1

Section \ Number of non-priority assignment	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	148	172	120	57	15	4	1	1
3	35	151	189	124	45	12	3	0	0
4	110	227	153	53	14	2	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	219	119	43	9	0	0	0	0
Total	810	1368	1035	485	164	37	8	5	1

Table 51: Distribution of Daily Scenarios among the number of non-priority assignment for w_1=0.5, w_2=0.5, B=4, α=2

Section \ Number of non-priority assignment	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	151	176	117	56	13	4	0	1
3	35	151	189	124	45	12	3	0	0
4	110	227	153	53	14	2	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	219	119	43	9	0	0	0	0
Total	810	1371	1039	482	163	35	8	4	1

Table 52: Distribution of Daily Scenarios among the number of non-priority assignment for w_1=0.5, w_2=0.5, B=4, α=4

Number of non-priority assignment / Section	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	151	176	117	56	13	4	0	1
3	35	151	189	124	45	12	3	0	0
4	110	227	153	53	14	2	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	219	119	43	9	0	0	0	0
Total	810	1371	1039	482	163	35	8	4	1

Table 53: Distribution of Daily Scenarios among the number of non-priority assignment for w_1=0.5, w_2=0.5, B=4, α=6

Number of non-priority assignment / Section	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	151	176	117	56	13	4	0	1
3	35	151	189	124	45	12	3	0	0
4	110	227	153	53	14	2	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	219	119	43	9	0	0	0	0
Total	810	1371	1039	482	163	35	8	4	1

Combination 3 (w_1, w_2) = (0.1,0.9):

For the third combination, in which more weight is given for priority assignment on the expense of overall assignment, Tables 54-77 display the distribution of the number of unoccupied jobs along with the number of non-priority assignment upon providing different number of cross-training opportunities B and different risk-adjusting coefficient α.

Table 54: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=0.1, w2=0.9, B=2, α=1

Number of Unoccupied Jobs / Section	0	1	2	3	4	5	6	7
1	195	213	103	39	7	2	0	0
2*	218	213	92	27	9	0	0	0
3*	312	177	55	11	4	0	0	0
4	154	235	122	34	13	1	0	0
5	291	160	86	20	2	0	0	0
6	363	156	32	4	0	0	0	4
7	234	222	88	14	1	0	0	0
Total	1767	1376	578	149	36	3	0	4

Table 55: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=0.1, w2=0.9, B=2, α=2

Section \ Number of Unoccupied Jobs	0	1	2	3	4	5	6	7
1	195	213	103	39	7	2	0	0
2*	218	213	92	27	9	0	0	0
3*	312	177	55	11	4	0	0	0
4	154	235	122	34	13	1	0	0
5	291	160	86	20	2	0	0	0
6	363	156	32	4	0	0	0	4
7	234	222	88	14	1	0	0	0
Total	1767	1376	578	149	36	3	0	4

Table 56: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=0.1, w2=0.9, B=2, α=4

Section \ Number of Unoccupied Jobs	0	1	2	3	4	5	6	7
1*	210	214	101	26	8	0	0	0
2	189	219	107	31	13	0	0	0
3	274	191	68	21	5	0	0	0
4	154	235	122	34	13	1	0	0
5	291	160	86	20	2	0	0	0
6*	391	142	19	3	0	0	4	0
7	234	222	88	14	1	0	0	0
Total	1743	1383	591	149	42	1	4	0

Table 57: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=0.1, w2=0.9, B=2, α=6

Section \ Number of Unoccupied Jobs	0	1	2	3	4	5	6	7
1*	210	214	101	26	8	0	0	0
2	189	219	107	31	13	0	0	0
3	274	191	68	21	5	0	0	0
4	154	235	122	34	13	1	0	0
5	291	160	86	20	2	0	0	0
6*	391	142	19	3	0	0	4	0
7	234	222	88	14	1	0	0	0
Total	1743	1383	591	149	42	1	4	0

Table 58: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=0.1, w2=0.9, B=3, α=1

Section \ Number of Unoccupied Jobs	0	1	2	3	4	5	6	7
1	195	213	103	39	7	2	0	0
2*	218	213	92	27	9	0	0	0
3*	312	177	55	11	4	0	0	0
4	154	235	122	34	13	1	0	0
5	291	160	86	20	2	0	0	0
6	363	156	32	4	0	0	0	4
7*	272	204	71	12	0	0	0	0
Total	1805	1358	561	147	35	3	0	4

Table 59: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=0.1, w2=0.9, B=3, α=2

Number of Unoccupied Jobs / Section	0	1	2	3	4	5	6	7
1*	218	210	95	28	7	1	0	0
2*	218	213	92	27	9	0	0	0
3*	312	177	55	11	4	0	0	0
4	154	235	122	34	13	1	0	0
5	291	160	86	20	2	0	0	0
6	363	156	32	4	0	0	0	4
7	234	222	88	14	1	0	0	0
Total	1790	1373	570	138	36	2	0	4

Table 60: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=0.1, w2=0.9, B=3, α=4

Number of Unoccupied Jobs / Section	0	1	2	3	4	5	6	7
1*	210	214	101	26	8	0	0	0
2*	199	220	105	29	6	0	0	0
3	274	191	68	21	5	0	0	0
4	154	235	122	34	13	1	0	0
5	291	160	86	20	2	0	0	0
6*	391	142	19	3	0	0	4	0
7	234	222	88	14	1	0	0	0
Total	1753	1384	589	147	35	1	4	0

Table 61: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=0.1, w2=0.9, B=3, α=6

Number of Unoccupied Jobs / Section	0	1	2	3	4	5	6	7
1*	210	214	101	26	8	0	0	0
2*	199	220	105	29	6	0	0	0
3	274	191	68	21	5	0	0	0
4	154	235	122	34	13	1	0	0
5	291	160	86	20	2	0	0	0
6*	391	142	19	3	0	0	4	0
7	234	222	88	14	1	0	0	0
Total	1753	1384	589	147	35	1	4	0

Table 62: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=0.1, w2=0.9, B=4, α=1

Number of Unoccupied Jobs / Section	0	1	2	3	4	5	6	7
1*	218	210	95	28	7	1	0	0
2*	218	213	92	27	9	0	0	0
3*	312	177	55	11	4	0	0	0
4	154	235	122	34	13	1	0	0
5	291	160	86	20	2	0	0	0
6	363	156	32	4	0	0	0	4
7*	272	204	71	12	0	0	0	0
Total	1828	1355	553	136	35	2	0	4

Table 63: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=0.1, w2=0.9, B=4, α=2

Section \ Number of Unoccupied Jobs	0	1	2	3	4	5	6	7
1*	218	210	95	28	7	1	0	0
2*	218	213	92	27	9	0	0	0
3*	312	177	55	11	4	0	0	0
4*	174	235	107	33	10	0	0	0
5	291	160	86	20	2	0	0	0
6	363	156	32	4	0	0	0	4
7	234	222	88	14	1	0	0	0
Total	1810	1373	555	137	33	1	0	4

Table 64: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=0.1, w2=0.9, B=4, α=4

Section \ Number of Unoccupied Jobs	0	1	2	3	4	5	6	7
1*	210	214	101	26	8	0	0	0
2*	199	220	105	29	6	0	0	0
3	274	191	68	21	5	0	0	0
4*	174	235	107	33	10	0	0	0
5	291	160	86	20	2	0	0	0
6*	391	142	19	3	0	0	4	0
7	234	222	88	14	1	0	0	0
Total	1773	1384	574	146	32	0	4	0

Table 65: Distribution of Daily Scenarios among the number of unoccupied jobs for w1=0.1, w2=0.9, B=4, α=6

Section \ Number of Unoccupied Jobs	0	1	2	3	4	5	6	7
1*	210	214	101	26	8	0	0	0
2*	199	220	105	29	6	0	0	0
3	274	191	68	21	5	0	0	0
4*	174	235	107	33	10	0	0	0
5	291	160	86	20	2	0	0	0
6*	391	142	19	3	0	0	4	0
7	234	222	88	14	1	0	0	0
Total	1773	1384	574	146	32	0	4	0

Table 66: Distribution of Daily Scenarios among the number of non-priority assignment for w_1=0.1, w_2=0.9, B=2, α=1

Section \ Number of non-priority assignment	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	151	176	117	56	13	4	0	1
3	35	151	189	124	45	12	3	0	0
4	110	227	153	53	14	2	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	219	119	43	9	0	0	0	0
Total	810	1371	1039	482	163	35	8	4	1

Table 67: Distribution of Daily Scenarios among the number of non-priority assignment for $w_1=0.1$, $w_2=0.9$, B=2, $\alpha=2$

Section \ Number of non-priority assignment	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	151	176	117	56	13	4	0	1
3	35	151	189	124	45	12	3	0	0
4	110	227	153	53	14	2	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	219	119	43	9	0	0	0	0
Total	810	1371	1039	482	163	35	8	4	1

Table 68: Distribution of Daily Scenarios among the number of non-priority assignment for w1=0.1, w2=0.9, B=2, $\alpha=4$

Section \ Number of non-priority assignment	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	151	176	117	56	13	4	0	1
3	35	151	189	124	45	12	3	0	0
4	110	227	153	53	14	2	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	219	119	43	9	0	0	0	0
Total	810	1371	1039	482	163	35	8	4	1

Table 69: Distribution of Daily Scenarios among the number of non-priority assignment for w1=0.1, w2=0.9, B=2, $\alpha=6$

Section \ Number of non-priority assignment	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	151	176	117	56	13	4	0	1
3	35	151	189	124	45	12	3	0	0
4	110	227	153	53	14	2	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	219	119	43	9	0	0	0	0
Total	810	1371	1039	482	163	35	8	4	1

Table 70: Distribution of Daily Scenarios among the number of non-priority assignment for $w_1=0.1$, $w_2=0.9$, B=3, $\alpha=1$

Section \ Number of non-priority assignment	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	151	176	117	56	13	4	0	1
3	35	151	189	124	45	12	3	0	0
4	110	227	153	53	14	2	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	219	119	43	9	0	0	0	0
Total	810	1371	1039	482	163	35	8	4	1

Table 71: Distribution of Daily Scenarios among the number of non-priority assignment for w_1=0.1, w_2=0.9, B=3, α=2

Number of non-priority assignment / Section	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	151	176	117	56	13	4	0	1
3	35	151	189	124	45	12	3	0	0
4	110	227	153	53	14	2	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	219	119	43	9	0	0	0	0
Total	810	1371	1039	482	163	35	8	4	1

Table 72: Distribution of Daily Scenarios among the number of non-priority assignment for w1=0.1, w2=0.9, B=3, α=4

Number of non-priority assignment / Section	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	151	176	117	56	13	4	0	1
3	35	151	189	124	45	12	3	0	0
4	110	227	153	53	14	2	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	219	119	43	9	0	0	0	0
Total	810	1371	1039	482	163	35	8	4	1

Table 73: Distribution of Daily Scenarios among the number of non-priority assignment for w_1=0.1, w_2=0.9, B=3, α=6

Number of non-priority assignment / Section	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	151	176	117	56	13	4	0	1
3	35	151	189	124	45	12	3	0	0
4	110	227	153	53	14	2	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	219	119	43	9	0	0	0	0
Total	810	1371	1039	482	163	35	8	4	1

Table 74: Distribution of Daily Scenarios among the number of non-priority assignment for w_1=0.1, w_2=0.9, B=4, α=1

Number of non-priority assignment / Section	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	151	176	117	56	13	4	0	1
3	35	151	189	124	45	12	3	0	0
4	110	227	153	53	14	2	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	219	119	43	9	0	0	0	0
Total	810	1371	1039	482	163	35	8	4	1

Table 75: Distribution of Daily Scenarios among the number of non-priority assignment for w_1=0.1, w_2=0.9, B=4, α=2

Section \ Number of non-priority assignment	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	151	176	117	56	13	4	0	1
3	35	151	189	124	45	12	3	0	0
4	110	227	153	53	14	2	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	219	119	43	9	0	0	0	0
Total	810	1371	1039	482	163	35	8	4	1

Table 76: Distribution of Daily Scenarios among the number of non-priority assignment for w1=0.1, w2=0.9, B=4, α=4

Section \ Number of non-priority assignment	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	151	176	117	56	13	4	0	1
3	35	151	189	124	45	12	3	0	0
4	110	227	153	53	14	2	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	219	119	43	9	0	0	0	0
Total	810	1371	1039	482	163	35	8	4	1

Table 77: Distribution of Daily Scenarios among the number of non-priority assignment for w_1=0.1, w_2=0.9, B=4, α=6

Section \ Number of non-priority assignment	0	1	2	3	4	5	6	7	8
1	78	183	180	81	29	7	1	0	0
2	41	151	176	117	56	13	4	0	1
3	35	151	189	124	45	12	3	0	0
4	110	227	153	53	14	2	0	0	0
5	159	215	132	46	6	1	0	0	0
6	218	225	90	18	4	0	0	4	0
7	169	219	119	43	9	0	0	0	0
Total	810	1371	1039	482	163	35	8	4	1

5.2.1 Computational Analysis

5.2.1.1 Algorithm Performance

The aim behind solving the Lagrangian dual is to find the optimal λ that would yield the highest possible value of Z(λ) in an attempt to make it as close as possible to Z. In order to evaluate the performance of this implemented Lagrangian Relaxation algorithm in terms of providing near-optimal solutions, the duality gap being the difference between the optimal values of Z(λ) and Z is therefore examined. For all possible combinations of α, B, w_1, and w_2 presented in the previously reported results, optimal λ for each combination have been reached such that it satisfies Z(λ^*)=Z(λ^*) - $\lambda^* \times \left(\sum_{i\in I_k}\sum_{j\in J_k}\sum_{k\in K}\left(u_{i,j}^k - V_{i,j}^k\right) - B\right)$. In other words, while $\lambda^* \geq 0,$ conditions $\lambda^* \times \left(\sum_{i\in I_k}\sum_{j\in J_k}\sum_{k\in K}\left(u_{i,j}^k - V_{i,j}^k\right) - B\right) = 0$ and $\left(\sum_{i\in I_k}\sum_{j\in J_k}\sum_{k\in K}\left(u_{i,j}^k - V_{i,j}^k\right) - B\right) = 0$ are satisfied yielding a zero duality gap conveying that the optimal solution of the Lagrangian dual Z(λ) is also an optimal solution to the primary MILP.

5.2.1.2 Running Time

In order to relief the model from unnecessary computational effort, the assignment stage is carried out prior to the inclusion of any cross-training opportunity to determine the daily scenarios with full job occupancy and full commitment to priority-job assignment; these scenarios have no impact in directing the allocation of cross-training opportunities as they have a null contribution to the value of the objective function and can thus be separated from the model input. However, it is good to note that the number of scenarios complying with full job occupancy only is greater than that complying with full job occupancy and priority-job assignment at the same time; and thus, different computational time is attained for the cases of w_2=0 and w_2≠0. Applying these practices, the programming tool used is Matlab while integrating CPLEX 12.8 as a solver for the rendered mixed integer linear programming model. The time needed to solve the dual Langrangian of this two-stage stochastic model when w_2≠0 including all 7 sections is an average of 68 seconds compared to 49 seconds of solver time for the case when w_2=0 explained by the difference in number of scenarios considered for each case.

5.2.2 Model Performance

5.2.2.1 Impact of B

First, results reflect the fact that at a constant α, as the management provides a higher number of cross-training opportunities, the overall number of unoccupied jobs across the total sections' scenarios decreases. For (w_1, w_2, α)=(1, 0, 1) and for rising values in B, there is an apparent increase in the number of occupied jobs manifested in boosting the number of scenarios meeting zero shortage in one-to-one job occupancy from a total of 1722 scenarios for zero provided cross-training opportunities to 2279 scenarios for 8 cross-training opportunities (Figure 9).

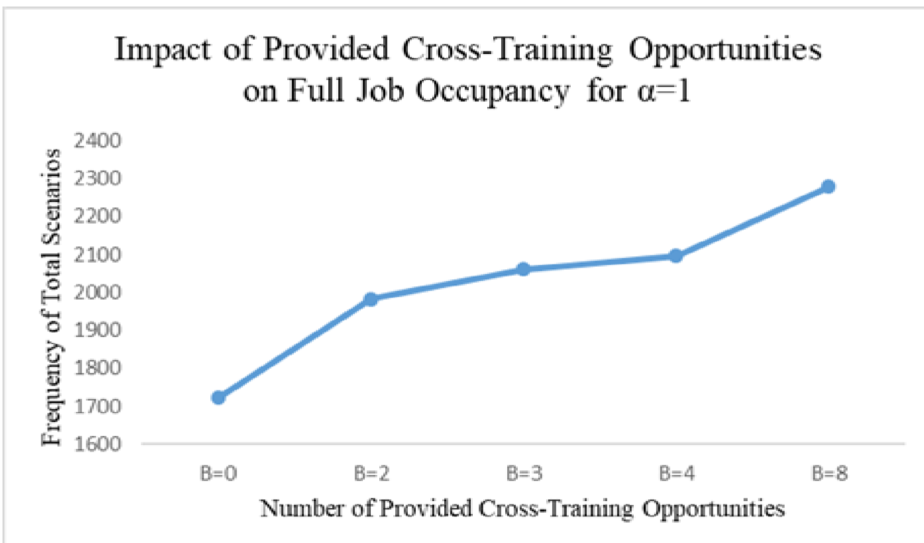

Figure 9: Impact of B on Full Job Occupancy

This is accompanied with a decrease in all frequencies of occurrence of unoccupied jobs (other than zero unoccupied jobs) as shown in Figure 10.This indicates that the more workers are cross-trained, the more rich is the combination of skills, and the more the assembly line is capable of fighting the shortage in skills under the uncertainty of absenteeism.

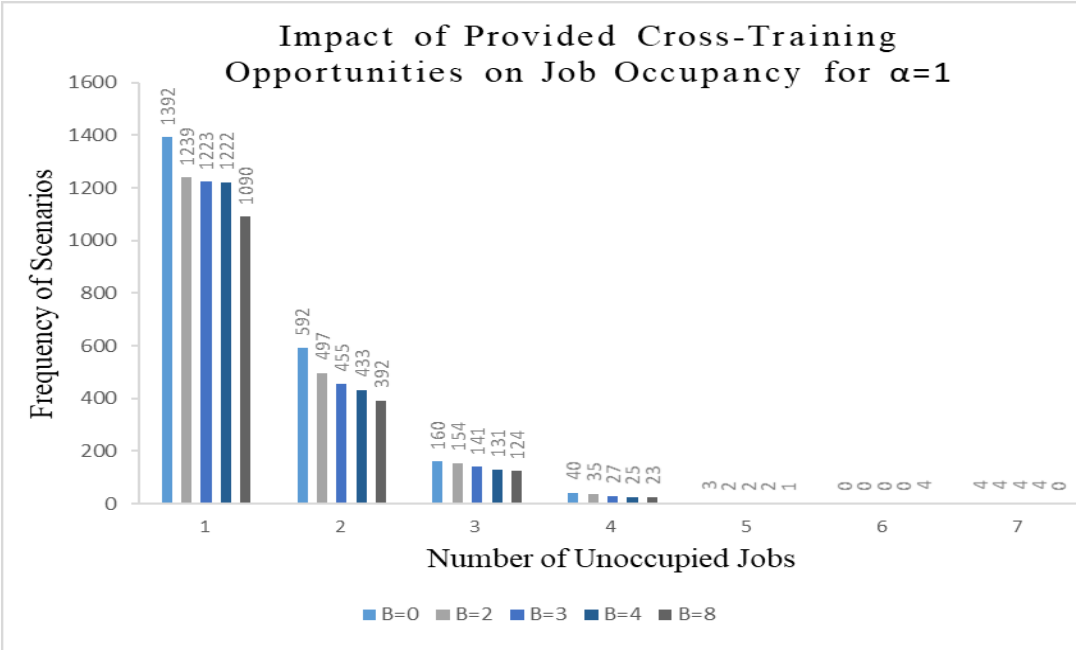

Figure 10: Impact of B on Job Occupancy

5.2.2.2 Impact of α

Additionally, for a fixed number of cross-training opportunities, results show that as α increases from 1 to 6 at (w_1, w_2) = (1,0), there is an evident decrease in the number of instances of incurring extremely high numbers of unoccupied jobs. When α=1 and α=2, there is an enhancement to the overall number of unoccupied jobs for all three common cases of provided cross-training opportunities (B=2, 3, 4), however, with neither an enhancement to the maximal number of unoccupied jobs being 7 nor to the number of instances that this worst case assignment is met. Nevertheless, as α increases from 1 to 2, there is an enhancement to the number of instances of meeting 3 and 4 unoccupied jobs during daily operation. This is strongly depicted for the case when B=2 in which the 154 and 35 instances of meeting 3 and 4 unoccupied jobs respectively at α =1 is reduced to 144 and 28 instances of meeting 3 and 4 unoccupied jobs at α =2; this is attended by a sacrifice to the frequency of lower number of unoccupied jobs (Figures 11 & 12).

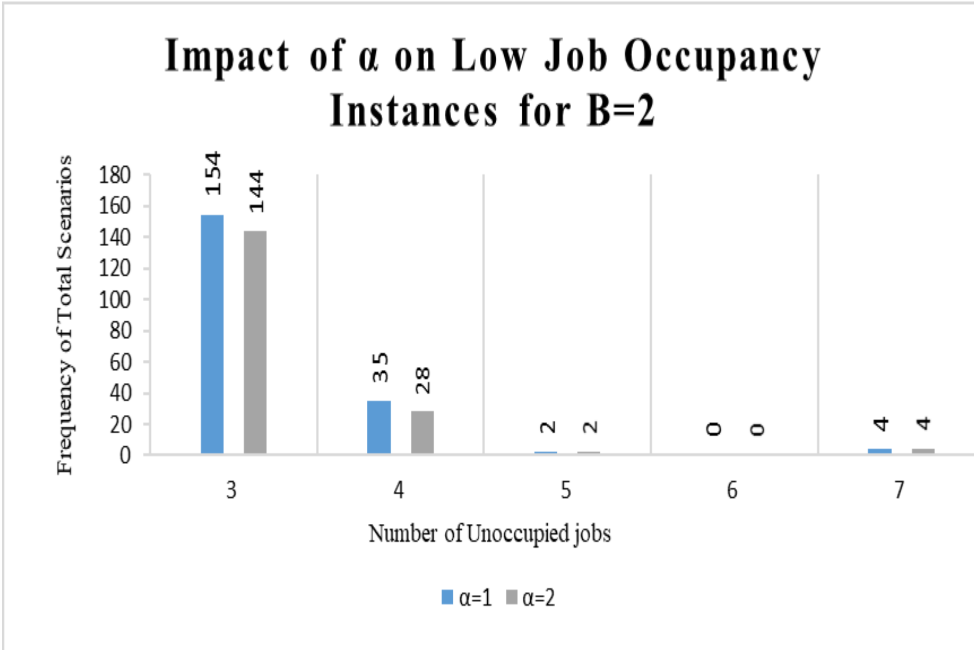

Figure 11: Impact of α on Low Job Occupancy Instances

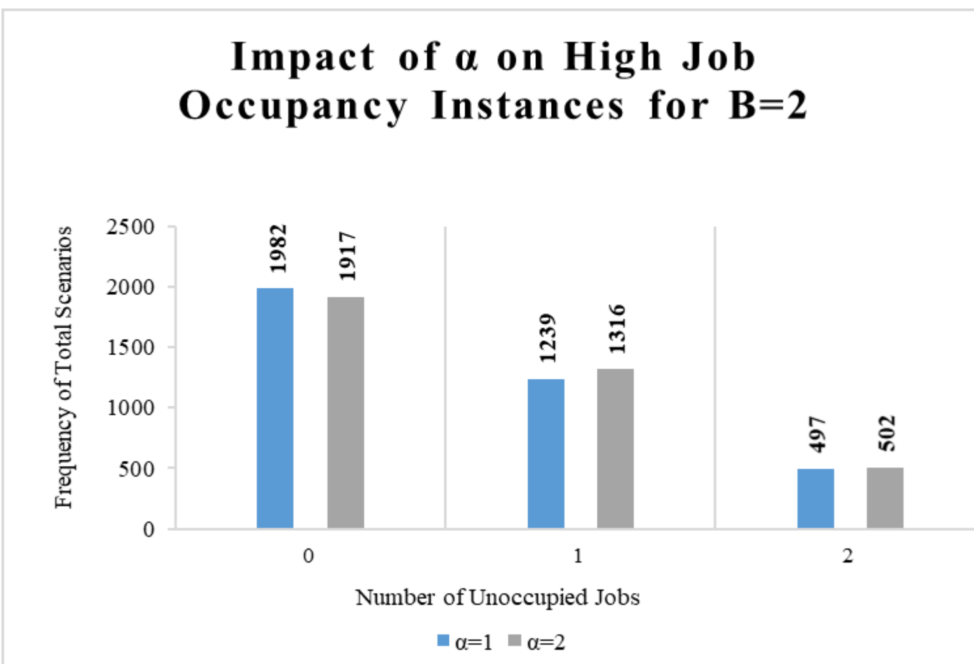

Figure 12: Impact of α on High Job Occupancy Instances

Furthermore, for higher values of α, all 4 instances of having 7 unoccupied jobs shift to having 6 unoccupied jobs instead in which the suggested cross-training output tends more towards protecting bad scenarios characterized by their high number of unoccupied jobs. Similarly, the number of instances of observing 5 unoccupied jobs in total decreases with the increase of α. This case is observed when B=3, the number of instances of having 5 unoccupied jobs is 2 at α=4 and decreases to become 1 at α=6. This reduction comes slightly on the expense of the number of scenarios of lower number of unoccupied jobs (Figure 13).

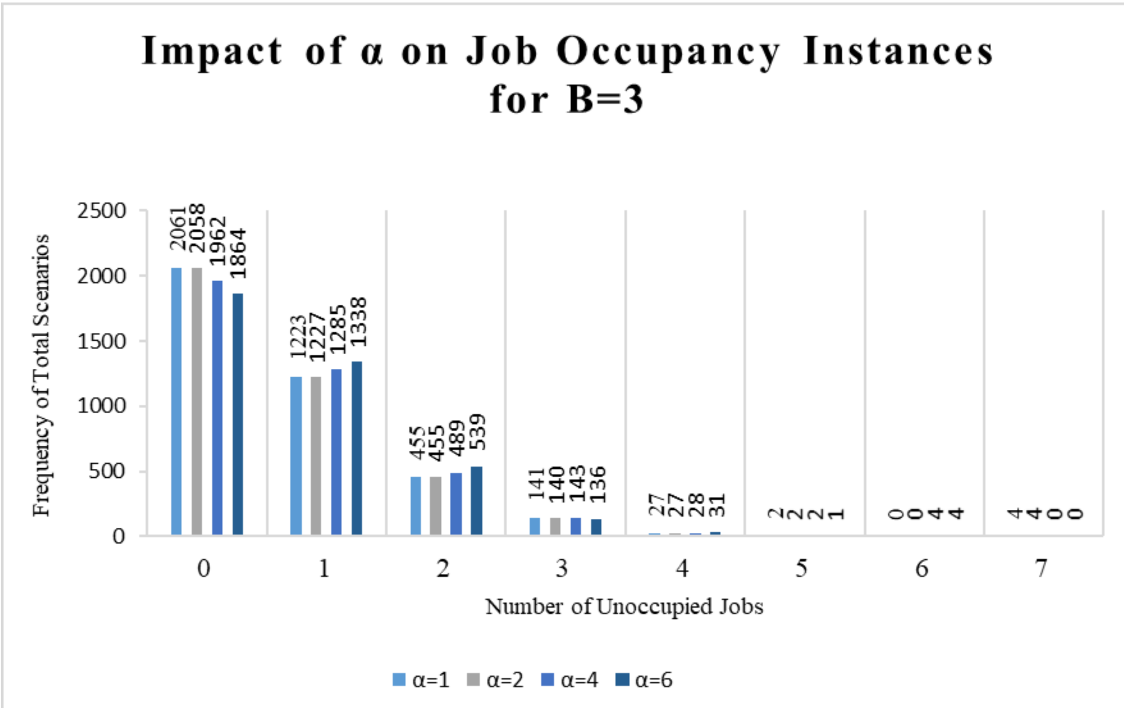

Figure 13: Impact of alpha on Job Occupancy Instances

The above results support the claim that the higher the value the α, the more weight is exerted on the scarce occurrences of high unoccupancy, and thus, the more the cross-training suggestion is capable of aiding these occurrences. As α increases, the more is the solution in favor of protecting scenarios characterized by their extreme shortage in available skills.

Furthermore, the balancing function of α is reflected in the designed case of providing 8 cross-training opportunities to the assembly line at (w_1, w_2) = (1,0). When α=1, the opportunities are not distributed evenly among the sections: Sections 1, 2, 4, 6, and 7 are each given 1 cross-training opportunity, whereas Section 3 is given 3 opportunities in an attempt to enhance the number of unoccupied jobs on average. Nonetheless, when α=6, 7 opportunities are distributed evenly such that each section is given one opportunity with section 1 receiving the surplus opportunity. In this case, it is clear that higher degrees of α treat sections more fairly tending to decrease the discrepancy of job occupation between them rather than optimizing particular sections on the expense of another.

5.2.2.3 Impact of w_1 and w_2

To analyze the role of w_1 and w_2, the expected value of the unoccupied jobs and non-priority assignment is compared for the three tested combinations: $(w_1, w_2) = (1,0)$, $(w_1, w_2) = (0.5,0.5$, and $(w_1, w_2) = (0.1,0.9)$. The expected value is calculated as a function of α.

Tables 78 summarizes the obtained results of objective value of overall unoccupied jobs.

Table 78: Expected Value of Overall Assignment

	W_1=1; W_2=0	**W_1=0.5; W_2=0.5**	**W_1=0.1; W_2=0.9**
α=1, B=2	5.1395	5.6297	5.6637
α=1, B=3	4.8336	5.5188	5.5528
α=1, B=4	4.6852	5.4168	5.4508
α=2, B=2	9.5063	10.5116	10.5116
α=2, B=3	8.9177	10.2272	10.2272
α=2, B=4	8.5403	9.9732	9.9732
α=4, B=2	66.2147	70.6064	70.6064
α=4, B=3	61.3488	67.0554	67.0555
α=4, B=4	57.2505	63.9893	63.9893
α=6, B=2	906.4723	934.0054	934.0053
α=6, B=3	830.4168	879.8784	879.8784
α=6, B=4	763.6404	826.9231	826.9231

The results above show that for all at hand combinations of α and B, the expected value of overall unoccupied jobs is lower for the case of $(w_1,w_2)=(1,0)$ than for the case of $(w_1,w_2)=(0.5,0.5)$; additionally, this expected value is lower for the case of $(w_1,w_2)=(0.5,0.5)$ than that of $(w_1,w_2)=(0.1,0.9)$. This is in compliance with the intended role of w1 and w2. For the case when emphasis is only on maximizing overall assignment on the expense of priority assignment ((w_1,

w_2) = (1,0)), overall unoccupied occupancy is minimal. Nonetheless, this overall unoccupancy increased for when overall assignment and priority assignment are treated equally and when priority assignment is enhanced on expense of overall assignment. Consequently, as w_1, being the weight controlling the minimization goal of overall unoccupancy, decreases then the expected value of overall unoccupancy increases.

Another table (Table 79) summarizes the values for the expected value of non-priority assignment.

Table 79: Expected Value of Non-priority Assignment

	W_1=1; W_2=0	**W_1=0.5; W_2=0.5**	**W_1=0.1; W_2=0.9**
α=1, B=2	11.4598	10.4204	10.3864
α=1, B=3	11.9535	10.4204	10.3864
α=1, B=4	12.2236	10.4204	10.3864
α=2, B=2	30.2379	24.8587	24.8587
α=2, B=3	31.5617	24.8587	24.8587
α=2, B=4	33.7639	24.8587	24.8587
α=4, B=2	452.6154	258.8694	258.8694
α=4, B=3	464.3113	258.8694	258.8694
α=4, B=4	501.8980	258.8694	258.8694
α=6, B=2	13647.6780	4901.1735	4901.1735
α=6, B=3	15045.7084	4901.1735	4901.1735
α=6, B=4	15191.7442	4901.1735	4901.1735

The table shows that the expected values of non-priority assignment results are more for the case of (w_1, w_2) = (1,0) than that of (w_1, w_2) = (0.5,0.5) for all possible combinations of α and β. Equivalently, this expected value at (w_1, w_2) = (0.5,0.5) is greater than or equal the corresponding value at (w_1, w_2) = (0.1,0.9). This could be referred to the fact that in the first case, priority assignment is neglected (w2=0) whereas in the second and third cases both assignment rules are taken into consideration such that the third cases more urges priority assignment. Hence, as the weight corresponding to the minimization of non-priority assignment increase, the expected value of this term decreases.

Besides, there's a depicted observation of constant expected value of non-priority assignment instances regardless of the number of provided-cross-training opportunities at a given α for $w_2>0$; this is justified by the fact that cross-training an additional worker to a job does not impact priority assignment as there's no pre-determinacy of a trainee to be prioritized for that specific newly acquired job.

Chapter 6 Conclusion

In this paper, we have presented a two-stage stochastic programming model for cross-training that copes with workers' undetermined absence events and adheres to the budget allocated for such an optimization to take place. Since different degrees of absenteeism is incurred throughout an operational period of time, this model allows developing unique cross-training plans in which each targets serving a different level of shortage in skills. The case study of the assembly line at hand has shed light on the improvement of job occupancy as budget availability increases. It has also showed the enhancement of job occupancy during riskier days as the risk-adjusting coefficient increases. Furthermore, it has minimized the discrepancy between the 7 sections. Not only that, the cross-training model has allowed for the engagement of different assignment rules in the case study besides overall assignment through the integration of an assignment model in its second stage.

Although cross-training models have been recognized throughout literature, this paper presents the first cross-training model that allows for adjustment according to level of absenteeism and that is in synchronization with daily operation rules.

Appendix

8.1 MATLAB Code

```
clc;
clear all;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%% parameters %%%%%%%%%%%%%%%%%%
% setting parameters for golden search method
eps=10^-4;
tol1=1;
iter=0;
lambda_L=0;
lambda_u=1000;
%golden ratio
G=0.5*(1+sqrt(5));
d=(G-1)*(lambda_u-lambda_L);
lambda_1=lambda_L+d;
lambda_2=lambda_u-d;
% setting parameters for cross-training model
alpha=4;
w1=1;
w2=0;
ncrosstrain=2;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%% read data %%%%%%%%%%%%%%%%%%
% read real attendance data % different sheet names depending on (w1,w2)
[num,txt,raw]=xlsread('sec1','att');
att1=num;
% determine number of available samples
nsim1=size(att1,2);
% read skill matrix
[num1,txt1,raw1]=xlsread('sec1.xlsx', 'skill');
skmatrix1=num1;
% determine number of jobs and number of employees
[nworker1,njob1] = size(skmatrix1);
% read preference matrix
[num2,txt2,raw2]=xlsread('sec1.xlsx', 'prior')
```

```
pref_matrix1=num2;
% determine number of initial scenarios(before removing those satisfying
% full occupancy and priority assignment)
[num3,~,~]=xlsread('sec1.xlsx', 'att_initial');
n1 = size(num3,2);
clear num; clear txt; clear raw;
clear num1; clear txt1; clear raw1;
clear num2; clear txt2; clear raw2; clear num3
% 2nd Section
[num,txt,raw]=xlsread('sec2.xlsx','att');
att2=num;
nsim2=size(att2,2);
[num1,txt1,raw1]=xlsread('sec2.xlsx', 'skill');
skmatrix2=num1;
[nworker2,njob2] = size(skmatrix2);
[num2,txt2,raw2]=xlsread('sec2.xlsx', 'prior');
pref_matrix2=num2;
[num3,~,~]=xlsread('sec2.xlsx', 'att_initial');
n2 = size(num3,2);
clear num; clear txt; clear raw;
clear num1; clear txt1; clear raw1;
clear num2; clear txt2; clear raw2; clear num3
% 3rd Section
[num,txt,raw]=xlsread('sec3.xlsx','att');
att3=num;
nsim3=size(att3,2);
[num1,txt1,raw1]=xlsread('sec3.xlsx', 'skill');
skmatrix3=num1;
[nworker3,njob3] = size(skmatrix3);
[num2,txt2,raw2]=xlsread('sec3.xlsx', 'prior');
pref_matrix3=num2;
[num3,~,~]=xlsread('sec3.xlsx', 'att_initial');
n3 = size(num3,2);
clear num; clear txt; clear raw;
clear num1; clear txt1; clear raw1;
clear num2; clear txt2; clear raw2; clear num3
% 4th Section
[num,txt,raw]=xlsread('sec4.xlsx','att');
att4=num;
nsim4=size(att4,2);
[num1,txt1,raw1]=xlsread('sec4.xlsx', 'skill');
skmatrix4=num1;
[nworker4,njob4] = size(skmatrix4);
[num2,txt2,raw2]=xlsread('sec4.xlsx', 'prior');
pref_matrix4=num2;
[num3,~,~]=xlsread('sec4.xlsx', 'att_initial');
n4=size(num3,2);
clear num; clear txt; clear raw;
clear num1; clear txt1; clear raw1;
clear num2; clear txt2; clear raw2; clear num3
% 5th Section
[num,txt,raw]=xlsread('sec5.xlsx','att');
att5=num;
nsim5=size(att5,2);
[num1,txt1,raw1]=xlsread('sec5.xlsx', 'skill');
skmatrix5=num1;
[nworker5,njob5] = size(skmatrix5);
```

```
[num2,txt2,raw2]=xlsread('sec5.xlsx', 'prior');
pref_matrix5=num2;
[num3,~,~]=xlsread('sec5.xlsx', 'att_initial');
n5=size(num3,2);
clear num; clear txt; clear raw;
clear num1; clear txt1; clear raw1;
clear num2; clear txt2; clear raw2; clear num3
% 6th Section
[num,txt,raw]=xlsread('sec6.xlsx','att');
att6=num;
nsim6=size(att6,2);
[num1,txt1,raw1]=xlsread('sec6.xlsx', 'skill');
skmatrix6=num1;
[nworker6,njob6] = size(skmatrix6);
[num2,txt2,raw2]=xlsread('sec6.xlsx', 'prior');
pref_matrix6=num2;
[num3,~,~]=xlsread('sec6.xlsx', 'att_initial');
n6=size(num3,2);
clear num; clear txt; clear raw;
clear num1; clear txt1; clear raw1;
clear num2; clear txt2; clear raw2; clear num3
% 7th Section
[num,txt,raw]=xlsread('sec7.xlsx','att');
att7=num;
nsim7=size(att7,2);
[num1,txt1,raw1]=xlsread('sec7.xlsx', 'skill');
skmatrix7=num1;
[nworker7,njob7] = size(skmatrix7);
[num2,txt2,raw2]=xlsread('sec7.xlsx', 'prior');
pref_matrix7=num2;
[num3,~,~]=xlsread('sec7.xlsx', 'att_initial');
n7=size(num3,2);
clear num; clear txt; clear raw;
clear num1; clear txt1; clear raw1;
clear num2; clear txt2; clear raw2; clear num3
% settings for solver
ops = sdpsettings('solver','cplex');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%% decision variables %%%%%%%%%%%%%%%%%%
% each x contains the updated skill matrix and assignment scenarios per
%section
x1 = binvar(nworker1,njob1*(nsim1+1),'full');
x2 = binvar(nworker2,njob2*(nsim2+1),'full');
x3 = binvar(nworker3,njob3*(nsim3+1),'full');
x4 = binvar(nworker4,njob4*(nsim4+1),'full');
x5 = binvar(nworker5,njob5*(nsim5+1),'full');
x6 = binvar(nworker6,njob6*(nsim6+1),'full');
x7 = binvar(nworker7,njob7*(nsim7+1),'full');
% intermediate variables for the plus function
% z for overall assignment
% zp for priority assignment
z1 = sdpvar(1,njob1*nsim1);
z2 = sdpvar(1,njob2*nsim2);
z3 = sdpvar(1,njob3*nsim3);
z4 = sdpvar(1,njob4*nsim4);
z5 = sdpvar(1,njob5*nsim5);
z6 = sdpvar(1,njob6*nsim6);
```

```
z7 = sdpvar(1,njob7*nsim7);
zp1 = sdpvar(1,njob1*nsim1);
zp2 = sdpvar(1,njob2*nsim2);
zp3 = sdpvar(1,njob3*nsim3);
zp4 = sdpvar(1,njob4*nsim4);
zp5 = sdpvar(1,njob5*nsim5);
zp6 = sdpvar(1,njob6*nsim6);
zp7 = sdpvar(1,njob7*nsim7);
z=[z1,z2,z3,z4,z5,z6,z7];
zp=[zp1,zp2,zp3,zp4,zp5,zp6,zp7];
nz=njob1*nsim1+njob2*nsim2+njob3*nsim3+njob4*nsim4+njob5*nsim5+njob6*nsim6+nj
ob7*nsim7;
s1=1; s2=s1+njob1*nsim1; s3=s2+njob2*nsim2; s4=s3+njob3*nsim3;
s5=s4+njob4*nsim4; s6=s5+njob5*nsim5; s7=s6+njob6*nsim6;
e1=njob1*nsim1; e2=e1+njob2*nsim2; e3=e2+njob3*nsim3; e4=e3+njob4*nsim4;
e5=e4+njob5*nsim5; e6=e5+njob6*nsim6; e7=e6+njob7*nsim7;
% call coefficients of plus function relative to alpha
coeff1 = coeff_alpha(njob1, alpha);
coeff2 = coeff_alpha(njob2, alpha);
coeff3 = coeff_alpha(njob3, alpha);
coeff4 = coeff_alpha(njob4, alpha);
coeff5 = coeff_alpha(njob5, alpha);
coeff6 = coeff_alpha(njob6, alpha);
coeff7 = coeff_alpha(njob7, alpha);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%% set constraints %%%%%%%%%%%%%%%%%
constr=[];
constr0=[];constr01=[];constr02=[];constr03=[];constr04=[];constr05=[];constr
06=[];constr07=[];
constr00=[];
constr1=[];constr2=[];constr3=[];constr4=[];constr5=[];constr6=[];constr7=[];
% linearization
for s=1:nsim1
    for j=0:(njob1-1)
        constr01=[constr01;z1(1,njob1*(s-1)+1+j)>=coeff1(1,j+1)*(njob1-
sum(sum(x1(1:nworker1,njob1*s+1:njob1*(s+1))))-j)];
        constr01=[constr01;zp1(1,njob1*(s-1)+1+j)>=coeff1(1,j+1)*(njob1-
sum(sum(pref_matrix1.*x1(1:nworker1,njob1*s+1:njob1*(s+1))))-j)];
    end
end
for s=1:nsim2
    for j=0:(njob2-1)
        constr02=[constr02;z2(1,njob2*(s-1)+1+j)>=coeff2(1,j+1)*(njob2-
sum(sum(x2(1:nworker2,njob2*s+1:njob2*(s+1))))-j)];
        constr02=[constr02;zp2(1,njob2*(s-1)+1+j)>=coeff2(1,j+1)*(njob2-
sum(sum(pref_matrix2.*x2(1:nworker2,njob2*s+1:njob2*(s+1))))-j)];
    end
end
for s=1:nsim3
    for j=0:(njob3-1)
        constr03=[constr03;z3(njob3*(s-1)+1+j)>=coeff3(1,j+1)*(njob3-
sum(sum(x3(1:nworker3,njob3*s+1:njob3*(s+1))))-j)];
        constr03=[constr03;zp3(1,njob3*(s-1)+1+j)>=coeff3(1,j+1)*(njob3-
sum(sum(pref_matrix3.*x3(1:nworker3,njob3*s+1:njob3*(s+1))))-j)];
    end
end
for s=1:nsim4
```

```
    for j=0:(njob4-1)
        constr04=[constr04;z4(1,njob4*(s-1)+1+j)>=coeff4(1,j+1)*(njob4-
sum(sum(x4(1:nworker4,njob4*s+1:njob4*(s+1))))-j)];
        constr04=[constr04;zp4(1,njob4*(s-1)+1+j)>=coeff4(1,j+1)*(njob4-
sum(sum(pref_matrix4.*x4(1:nworker4,njob4*s+1:njob4*(s+1))))-j)];
    end
end
for s=1:nsim5
    for j=0:(njob5-1)
        constr05=[constr05;z5(1,njob5*(s-1)+1+j)>=coeff5(1,j+1)*(njob5-
sum(sum(x5(1:nworker5,njob5*s+1:njob5*(s+1))))-j)];
        constr05=[constr05;zp5(1,njob5*(s-1)+1+j)>=coeff5(1,j+1)*(njob5-
sum(sum(pref_matrix5.*x5(1:nworker5,njob5*s+1:njob5*(s+1))))-j)];
    end
end
for s=1:nsim6
    for j=0:(njob6-1)
        constr06=[constr06;z6(1,njob6*(s-1)+1+j)>=coeff6(1,j+1)*(njob6-
sum(sum(x6(1:nworker6,njob6*s+1:njob6*(s+1))))-j)];
        constr06=[constr06;zp6(1,njob6*(s-1)+1+j)>=coeff6(1,j+1)*(njob6-
sum(sum(pref_matrix6.*x6(1:nworker6,njob6*s+1:njob6*(s+1))))-j)];
    end
end
for s=1:nsim7
    for j=0:(njob7-1)
        constr07=[constr07;z7(1,njob7*(s-1)+1+j)>=coeff7(1,j+1)*(njob7-
sum(sum(x7(1:nworker7,njob7*s+1:njob7*(s+1))))-j)];
        constr07=[constr07;zp7(1,njob7*(s-1)+1+j)>=coeff7(1,j+1)*(njob7-
sum(sum(pref_matrix7.*x7(1:nworker7,njob7*s+1:njob7*(s+1))))-j)];
    end
end
for k=1:nz
    constr0=[constr0;z(1,k)>=0];
end
for k=1:nz
    constr0=[constr0;zp(1,k)>=0];
end
% team leader constraint
% can be a team leader only if knows major jobs
% not necessarily a tag relief
for i = 1:nworker1
    constr00 = [constr00;x1(i,njob1)-sum(x1(i,1:(njob1-2)))/(njob1-2) <=0];
end
% tag relief constraint
% can be tag relief only if knows major jobs
for i = 1:nworker1
    constr00 = [constr00;x1(i,(njob1-1))-sum(x1(i,1:(njob1-2)))/(njob1-2)
<=0];
end
for i = 1:nworker2
    constr00 = [constr00;x2(i,njob2)-sum(x2(i,1:(njob2-2)))/(njob2-2) <=0];
end
for i = 1:nworker2
    constr00 = [constr00;x2(i,(njob2-1))-sum(x2(i,1:(njob2-2)))/(njob2-2)
<=0];
end
for i = 1:nworker3
```

```
    constr00 = [constr00;x3(i,njob3)-sum(x3(i,1:(njob3-2)))/(njob3-2) <=0];
end
for i = 1:nworker3
    constr00 = [constr00;x3(i,(njob3-1))-sum(x3(i,1:(njob3-2)))/(njob3-2)
<=0];
end
for i = 1:nworker4
    constr00 = [constr00;x4(i,njob4)-sum(x4(i,1:(njob4-2)))/(njob4-2) <=0];
end
for i = 1:nworker4
    constr00 = [constr00;x4(i,(njob4-1))-sum(x4(i,1:(njob4-2)))/(njob4-2)
<=0];
end
for i = 1:nworker5
    constr00 = [constr00;x5(i,njob5)-sum(x5(i,1:(njob5-2)))/(njob5-2) <=0];
end
for i = 1:nworker5
    constr00 = [constr00;x5(i,(njob5-1))-sum(x5(i,1:(njob5-2)))/(njob5-2)
<=0];
end
for i = 1:nworker6
    constr00 = [constr00;x6(i,njob6)-sum(x6(i,1:(njob6-2)))/(njob6-2) <=0];
end
for i = 1:nworker6
    constr00 = [constr00;x6(i,(njob6-1))-sum(x6(i,1:(njob6-2)))/(njob6-2)
<=0];
end
for i = 1:nworker7
    constr00 = [constr00;x7(i,njob7)-sum(x7(i,1:(njob7-2)))/(njob7-2) <=0];
end
for i = 1:nworker7
    constr00 = [constr00;x7(i,(njob7-1))-sum(x7(i,1:(njob7-2)))/(njob7-2)
<=0];
end
% Adding knowledge or maintaining previous knowledge is allowed
for i = 1:nworker1
    for j = 1:njob1
        constr00 = [constr00;x1(i,j) >= skmatrix1(i,j)];
    end
end
for i = 1:nworker2
    for j = 1:njob2
        constr00 = [constr00;x2(i,j) >= skmatrix2(i,j)];
    end
end
for i = 1:nworker3
    for j = 1:njob3
        constr00 = [constr00;x3(i,j) >= skmatrix3(i,j)];
    end
end
for i = 1:nworker4
    for j = 1:njob4
        constr00 = [constr00;x4(i,j) >= skmatrix4(i,j)];
    end
end
for i = 1:nworker5
    for j = 1:njob5
```

```
        constr00 = [constr00;x5(i,j) >= skmatrix5(i,j)];
    end
end
for i = 1:nworker6
    for j = 1:njob6
        constr00 = [constr00;x6(i,j) >= skmatrix6(i,j)];
    end
end
for i = 1:nworker7
    for j = 1:njob7
        constr00 = [constr00;x7(i,j) >= skmatrix7(i,j)];
    end
end
% assignment constraints
for s = 1:nsim1
    for i = 1:nworker1
        for j = 1:njob1
            constr1 = [constr1;x1(i,j+s*njob1) <= x1(i,j)];%to be assigned
need to be trained
            constr1 = [constr1;sum(x1(1:nworker1,j+s*njob1)) <= 1];%one to
one assignment
        end
        constr1 = [constr1;sum(x1(i,1+s*njob1:(1+s)*njob1)) <= att1(i,s)];%to
be assigned need to show up
    end
end
for s = 1:nsim2
    for i = 1:nworker2
        for j = 1:njob2
            constr2 = [constr2;x2(i,j+s*njob2) <= x2(i,j)];%to be assigned
need to be trained
            constr2 = [constr2;sum(x2(1:nworker2,j+s*njob2)) <= 1];%one to
one assignment
        end
        constr2 = [constr2;sum(x2(i,1+s*njob2:(1+s)*njob2)) <= att2(i,s)];%to
be assigned need to show up
    end
end
for s = 1:nsim3
    for i = 1:nworker3
        for j = 1:njob3
            constr3 = [constr3;x3(i,j+s*njob3) <= x3(i,j)];%to be assigned
need to be trained
            constr3 = [constr3;sum(x3(1:nworker3,j+s*njob3)) <= 1];%one to
one assignment
        end
        constr3 = [constr3;sum(x3(i,1+s*njob3:(1+s)*njob3)) <= att3(i,s)];%to
be assigned need to show up
    end
end
for s = 1:nsim4
    for i = 1:nworker4
        for j = 1:njob4
            constr4 = [constr4;x4(i,j+s*njob4) <= x4(i,j)];%to be assigned
need to be trained
            constr4 = [constr4;sum(x4(1:nworker4,j+s*njob4)) <= 1];%one to
one assignment
```

```
        end
        constr4 = [constr4;sum(x4(i,1+s*njob4:(1+s)*njob4)) <= att4(i,s)];%to
be assigned need to show up
    end
end
for s = 1:nsim5
    for i = 1:nworker5
        for j = 1:njob5
            constr5 = [constr5;x5(i,j+s*njob5) <= x5(i,j)];%to be assigned
need to be trained
            constr5 = [constr5;sum(x5(1:nworker5,j+s*njob5)) <= 1];%one to
one assignment
        end
        constr5 = [constr5;sum(x5(i,1+s*njob5:(1+s)*njob5)) <= att5(i,s)];%to
be assigned need to show up
    end
end
for s = 1:nsim6
    for i = 1:nworker6
        for j = 1:njob6
            constr6 = [constr6;x6(i,j+s*njob6) <= x6(i,j)];%to be assigned
need to be trained
            constr6 = [constr6;sum(x6(1:nworker6,j+s*njob6)) <= 1];%one to
one assignment
        end
        constr6 = [constr6;sum(x6(i,1+s*njob6:(1+s)*njob6)) <= att6(i,s)];%to
be assigned need to show up
    end
end
for s = 1:nsim7
    for i = 1:nworker7
        for j = 1:njob7
            constr7 = [constr7;x7(i,j+s*njob7) <= x7(i,j)];%to be assigned
need to be trained
            constr7 = [constr7;sum(x7(1:nworker7,j+s*njob7)) <= 1];%one to
one assignment
        end
        constr7 = [constr7;sum(x7(i,1+s*njob7:(1+s)*njob7)) <= att7(i,s)];%to
be assigned need to show up
    end
end
constr=[constr0;constr01;constr02;constr03;constr04;constr05;constr06;constr0
7;constr00;constr1;constr2;constr3;constr4;constr5;constr6;constr7];
% explicit form of relaxed constraint g(x)
relaxed_const=(sum(sum(x1(1:nworker1,1:njob1)))-
sum(sum(skmatrix1(1:nworker1,1:njob1)))+sum(sum(x2(1:nworker2,1:njob2)))-
sum(sum(skmatrix2(1:nworker2,1:njob2)))+sum(sum(x3(1:nworker3,1:njob3)))-
sum(sum(skmatrix3(1:nworker3,1:njob3)))+sum(sum(x4(1:nworker4,1:njob4)))-
sum(sum(skmatrix4(1:nworker4,1:njob4)))+sum(sum(x5(1:nworker5,1:njob5)))-
sum(sum(skmatrix5(1:nworker5,1:njob5)))+sum(sum(x6(1:nworker6,1:njob6)))-
sum(sum(skmatrix6(1:nworker6,1:njob6)))+sum(sum(x7(1:nworker7,1:njob7)))-
sum(sum(skmatrix7(1:nworker7,1:njob7)))-ncrosstrain);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%% Golden search section method %%%%%%%%%%%%%%%%%%
% obj function
```

```
ob1=w1*(((sum(z(1,s1:e1)))/n1)+((sum(z(1,s2:e2)))/n2)+((sum(z(1,s3:e3)))/n3)+
((sum(z(1,s4:e4)))/n4)+((sum(z(1,s5:e5)))/n5)+((sum(z(1,s6:e6)))/n6)+((sum(z(
1,s7:e7)))/n7));
ob2=w2*(((sum(zp(1,s1:e1)))/n1)+((sum(zp(1,s2:e2)))/n2)+((sum(zp(1,s3:e3)))/n
3)+((sum(zp(1,s4:e4)))/n4)+((sum(zp(1,s5:e5)))/n5)+((sum(zp(1,s6:e6)))/n6)+((
sum(zp(1,s7:e7)))/n7));
ob3_1=lambda_1*relaxed_const;
obj1=ob1+ob2+ob3_1;
outputILP1 = optimize(constr,obj1,ops)
f1=value(obj1);
%to determine f2, define new obj function with lambda=lambda2 and solve
%using previous constraints and settings
ob3_2=lambda_2*relaxed_const;
obj2=ob1+ob2+ob3_2;
outputILP2 = optimize(constr,obj2,ops)
f2=value(obj2);
while (tol1>eps)
    if f1>f2
        clear ob3_1; clear obj1; clear outputILP1;
        lambda_max=lambda_1;
        fmax=f1;
        lambda_L=lambda_2;
        lambda_2=lambda_1;
        f2=f1;
        d=(G-1)*(lambda_u-lambda_L);
        lambda_1=lambda_L+d;
        ob3_1 =lambda_1*relaxed_const;
        obj1=ob1+ob2+ob3_1;
        outputILP1 = optimize(constr,obj1,ops)
        f1=value(obj1);
        time1=outputILP1.solvertime;
        time2=outputILP1.yalmiptime;
    else
        clear ob3_2; clear obj2; clear outputILP2;
        lambda_max=lambda_2;
        fmax=f2;
        lambda_u=lambda_1;
        lambda_1=lambda_2;
        f1=f2;
        d=(G-1)*(lambda_u-lambda_L);
        lambda_2=lambda_u-d;
        ob3_2 =lambda_2*relaxed_const;
        obj2=ob1+ob2+ob3_2;
        outputILP2 = optimize(constr,obj2,ops)
        f2=value(obj2);
        time1=outputILP2.solvertime;
        time2=outputILP2.yalmiptime;
    end
    iter=iter+1;
    store_tt1(iter,1)=time1;
    store_tt2(iter,1)=time2;
    tol1=abs(lambda_u-lambda_L);
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%% Final %%%%%%%%%%%%%%%%%
ob3= lambda_max*relaxed_const;
obj=ob1+ob2+ob3;
```

```
outputILP = optimize(constr,obj,ops)
% to check duality gap
check1=value(relaxed_const);
check2=value(lambda_max*relaxed_const);
% output
% value of updated skill matrix and assignment matrices
solution_ILPx1= value(x1);
solution_ILPx2=value(x2);
solution_ILPx3=value(x3);
solution_ILPx4=value(x4);
solution_ILPx5=value(x5);
solution_ILPx6=value(x6);
solution_ILPx7=value(x7);
% value of intermediate variables
solution_ILPz1=value(z1);
solution_ILPz2=value(z2);
solution_ILPz3=value(z3);
solution_ILPz4=value(z4);
solution_ILPz5=value(z5);
solution_ILPz6=value(z6);
solution_ILPz7=value(z7);
solution_ILPz = value(z);
solution_ILPzp1=value(zp1);
solution_ILPzp2=value(zp2);
solution_ILPzp3=value(zp3);
solution_ILPzp4=value(zp4);
solution_ILPzp5=value(zp5);
solution_ILPzp6=value(zp6);
solution_ILPzp7=value(zp7);
solution_ILPzp= value(zp);
% to determine what sections are updated
diff1=value(sum(sum(x1(1:nworker1,1:njob1)))-
sum(sum(skmatrix1(1:nworker1,1:njob1))));
diff2=value(sum(sum(x2(1:nworker2,1:njob2)))-
sum(sum(skmatrix2(1:nworker2,1:njob2))));
diff3=value(sum(sum(x3(1:nworker3,1:njob3)))-
sum(sum(skmatrix3(1:nworker3,1:njob3))));
diff4=value(sum(sum(x4(1:nworker4,1:njob4)))-
sum(sum(skmatrix4(1:nworker4,1:njob4))));
diff5=value(sum(sum(x5(1:nworker5,1:njob5)))-
sum(sum(skmatrix5(1:nworker5,1:njob5))));
diff6=value(sum(sum(x6(1:nworker6,1:njob6)))-
sum(sum(skmatrix6(1:nworker6,1:njob6))));
diff7=value(sum(sum(x7(1:nworker7,1:njob7)))-
sum(sum(skmatrix7(1:nworker7,1:njob7))));
% to determine updated spots in each section
for i = 1:nworker1
        for j = 1:njob1
            diff_sk1(i,j)=solution_ILPx1(i,j)-skmatrix1(i,j);
        end
end
for i = 1:nworker2
        for j = 1:njob2
            diff_sk2(i,j)=solution_ILPx2(i,j)-skmatrix2(i,j);
        end
end
for i = 1:nworker3
```

```
        for j = 1:njob3
            diff_sk3(i,j)=solution_ILPx3(i,j)-skmatrix3(i,j);
        end
end
for i = 1:nworker4
        for j = 1:njob4
            diff_sk4(i,j)=solution_ILPx4(i,j)-skmatrix4(i,j);
        end
end
for i = 1:nworker5
        for j = 1:njob5
            diff_sk5(i,j)=solution_ILPx5(i,j)-skmatrix5(i,j);
        end
end
for i = 1:nworker6
        for j = 1:njob6
            diff_sk6(i,j)=solution_ILPx6(i,j)-skmatrix6(i,j);
        end
end
for i = 1:nworker7
        for j = 1:njob7
            diff_sk7(i,j)=solution_ILPx7(i,j)-skmatrix7(i,j);
        end
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%% function %%%%%%%%%%%%%%%%%%
% this function is designed to determine coefficients needed to linearize
% the objective function
%these coefficients are unique for each alpha, being the risk adjusting
%factor
function coeff = coeff_alpha(njob, alpha)
coeff=zeros(1 ,njob);
coeff(1,1)=1;
for i=2:njob
    v=repmat(i,1,njob);
    for j=1:(njob-1)
        v1(1,1)=v(1,1);
        v1(1,j+1)=v(1,j+1)-j;
    end
    coeff(1,i)=(i^alpha)-sum(coeff(1,1:njob).*v1(1,1:njob));
end
end
```

References

Araz, O., & Fowler, J. (2008). Two stage stochastic integer programming model for workforce cross training. In *IIE Annual Conference and Expo 2008* (pp. 314-319)

Askin, R. G., & Huang, Y. (2001). Forming effective worker teams for cellular manufacturing. *International Journal of Production Research*, *39*(11), 2431–2451. https://doi.org/10.1080/00207540110040466

Billionnet, A. (1999). Integer programming to schedule a hierarchical workforce with variable demands. *European Journal of Operational Research*, *114*(1), 105–114. https://doi.org/10.1016/S0377-2217(98)00182-9

Blumenfeld, D. E., & Inman, R. R. (2009). Impact of Absenteeism on Assembly Line Quality and Throughput. *Production and Operations Management*, *18*(3), 333–343. https://doi.org/10.1111/j.1937-5956.2009.01019.x

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. https://doi.org/10.1080/07408170490496209

Bureau of Labor Statistics, U.S. Department of Labor, *The Economics Daily*, Work absences due to bad weather from 1994 to 2016 on the Internet at https://www.bls.gov/opub/ted/2017/work-absences-due-to-bad-weather-from-1994-to-2016.htm

Fisher, M. L. (2004). The Lagrangian Relaxation Method for Solving Integer Programming Problems. *Management Science*, *50*(12_supplement), 1861–1871. https://doi.org/10.1287/mnsc.1040.0263

Harrison, D. A., & Martocchio, J. J. (1998). Time for Absenteeism: A 20-Year Review of Origins, Offshoots, and Outcomes. *Journal of Management*, *24*(3), 305–350. https://doi.org/10.1177/014920639802400303

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. https://doi.org/10.1287/mnsc.50.1.83.27051

Hopp, W.J., & Spearman, M. L. (2011). *Factory Physics*. Waveland Press.Inman, R. R., Jordan, W. C., & Blumenfeld, D. E. (2004). Chained cross-training of assembly line workers. *International Journal of Production Research*, *42*(10), 1899–1910. https://doi.org/10.1080/00207540410001666224

Iravani, S. M. R., Kolfal, B., & Van Oyen, M. P. (2007). Call-Center Labor Cross-Training: It's a Small World After All. *Management Science*, *53*(7), 1102–1112. https://doi.org/10.1287/mnsc.1060.0621

Kilincli Taskiran, G., & Zhang, X. (2017). Mathematical models and solution approach for cross-training staff scheduling at call centers. *Computers & Operations Research*, *87*, 258–269. https://doi.org/10.1016/j.cor.2016.07.001

Kocakülâh, M. C., Kelley, A. G., Mitchell, K. M., & Ruggieri, M. P. (2011). Absenteeism Problems And Costs: Causes, Effects And Cures. *International Business & Economics Research Journal (IBER)*, *8*(5). https://doi.org/10.19030/iber.v8i5.3138

Mateo, R. (2007). The impact of absenteeism on the quality of assembly line production: The importance of a specialization requirement. *Human Factors and Ergonomics in Manufacturing*, *18*(1), 49–69. https://doi.org/10.1002/hfm.20095

Molleman, E., & Slomp, J. (1999). Functional flexibility and team performance. *International Journal of Production Research*, *37*(8), 1837–1858. https://doi.org/10.1080/002075499191021

Navarro, C., & Bass, C. (2006). The Cost of Employee Absenteeism. *Compensation & Benefits Review*, *38*(6), 26–30. https://doi.org/10.1177/0886368706295343

Prater, T., & Smith, K. (2011). Underlying Factors Contributing To Presenteeism And Absenteeism. *Journal of Business & Economics Research (JBER)*, *9*(6), 1. https://doi.org/10.19030/jber.v9i6.4374

Slomp, J., Bokhorst, J. A. C., & Molleman, E. (2005). Cross-training in a cellular manufacturing environment. *Computers & Industrial Engineering*, *48*(3), 609–624. https://doi.org/10.1016/j.cie.2003.03.004

Ulusam Seçkiner, S., Gökçen, H., & Kurt, M. (2007). An integer programming model for hierarchical workforce scheduling problem. *European Journal of Operational Research*, *183*(2), 694–699. https://doi.org/10.1016/j.ejor.2006.10.030

Weaver, R. (2006). Cost of Presenteeism Surpasses Absenteeism. Retrieved from https://ezinearticles.com/?Cost-of-Presenteeism-Surpasses-Absenteeism&id=397305

Wirojanagud, P., Gel, E. S., Fowler, J. W., & Cardy, R. (2007). Modelling inherent worker differences for workforce planning. *International Journal of Production Research*, *45*(3), 525–553. https://doi.org/10.1080/00207540600792242