applied sciences

MDPI

Article

Two-Stage Adaptive Large Neighbourhood Search for Team Formation and Worker Assignment Problems in Cellular Manufacturing Systems

Thanatat Pasupa [1,*] and Sadami Suzuki [2]

[1] Department of Industrial Engineering and Economics, Tokyo Institute of Technology, Tokyo 152-8550, Japan
[2] Graduate School of Social Sciences, Yokohama National University, Kanagawa 240-8501, Japan
* Correspondence: pasupa.t.aa@m.titech.ac.jp

Abstract: We present a novel mathematical programming model to address a team-oriented worker assignment problem, called the team formation and worker assignment problem (TFWAP). The model establishes a multi-skilled team with high group cohesion to meet cell operational requirements. To this end, we developed a two-stage decision methodology based on an adaptive large neighbourhood search (ALNS) method as a solution approach. The first stage was a team formation problem that maximised workers' skills. The second stage was a worker assignment problem that minimised the total inventory level and variations in the average cell worker's idle time. The performance of the two-stage ALNS method was assessed on ten cell formation benchmarks selected from the literature. The computational results show that the two-stage ALNS method could provide a solution equivalent to the exact method based on the heuristic-based brute force search (HBBFS) for small instances in the team formation stage. Moreover, the two-stage ALNS method outperformed the non-dominated sorting genetic algorithm-II (NSGA-II)-based single-stage decision methodology on all ten cell formation benchmarks in the worker assignment stage. Finally, the two-way analysis of variance (ANOVA) test highlighted the impact of the cell-cohesion requirement on performance when forming a team in a cell.

Keywords: adaptive large neighbourhood search; cellular manufacturing system; cross-trained workers; sociometry; team formation; worker assignment

check for updates

Citation: Pasupa, T.; Suzuki, S. Two-Stage Adaptive Large Neighbourhood Search for Team Formation and Worker Assignment Problems in Cellular Manufacturing Systems. *Appl. Sci.* **2022**, *12*, 8323. https://doi.org/10.3390/app12168323

Academic Editor: Paweł Sitek

Received: 28 June 2022
Accepted: 17 August 2022
Published: 19 August 2022

1. Introduction

In a cellular manufacturing system (CMS), a group of multi-skilled workers plays an important role because they create a capacity buffer in response to fluctuations in workforce demand and supply [1–3]. Workers are required to work as a team under the design of a dual resource-constrained manufacturing cell [1,4,5], a teamwork environment that creates a high degree of worker interaction. Under a team structure, cell team members are cross-trained and work together toward a common goal, which requires team dynamics and synergies in performing tasks [1]. Successful teams require high-quality taskwork, teamwork, and good working relationships. These team requirements are crucial when managing a cross-trained workforce in a CMS. Poor working relationships among cell team members can lead to extreme job dissatisfaction and an unpleasant working environment, which deteriorate cell productivity and quality of work [1,6]. Manufacturing firms would be unable to perceive the benefits of CMS if managers failed to form a cohesive cell team.

The dual resource-constrained manufacturing cell typically contains more tasks than workers. Cell team members are generally responsible for a wide range of tasks. This cell characteristic makes it essential to provide a proper assignment to workers. An assignment that mismatches workers' skills could decrease workers' morale and limit their productivity. Therefore, it is necessary to carefully select cell team members and assign workers to operate tasks with the desired qualifications. This perspective motivated us to propose an effective method that helps managers to form a cohesive, efficient work team in a CMS. The proposed method would benefit the workers and managers since both human and operation aspects of a CMS were considered.

Many researchers have considered the interpersonal relationships of team members in the team formation problem, mainly in the context of project management and concurrent engineering [7–14]. The existing literature often overlooks the workers' interpersonal relationships when addressing the worker assignment problem in a CMS. Furthermore, the impact of group cohesion on cell team formation remained unstudied. This study attempts to fill the literature gap by addressing the research question as follows: How does the group cohesion requirement affect the performance of cell team formation? To address this research question, we conducted an extensive-computational experiment based on a generated dataset to analyse the impact of group cohesion requirements on the performance of cell team formation. The experiment implemented the proposed mathematical programming model and solving approach for the team formation and worker assignment problem (TFWAP).

The major contributions of this study are twofold: theoretical contribution and managerial contribution. For theoretical contribution, we formulated a new mathematical programming model to address the TFWAP in a CMS, considering the interpersonal relationship of workers. We proposed a two-stage methodology as a solution approach based on the ALNS method (two-stage ALNS). The primary goal of this methodology is to generate feasible solutions within a reasonable computational time. The two-stage ALNS method reduced the computational complexity of the TFWAP by decomposing the problem into two dependent stages—that is, (1) team formation and (2) worker assignment. For managerial contribution, we provided insights by revealing the impact of group cohesion requirement on cell team formation through statistical analysis, with the results obtained from the two-stage ALNS method.

The remainder of this paper is organised as follows. Section 2 presents several relevant conclusions obtained from the study reported in this paper. Section 3 presents the problem description and mathematical model formulation. In Section 4, the proposed solution methodology is presented. Section 5 presents the numerical example. Section 6 presents the computational experiments. Finally, Section 7 presents the conclusions and discussion of future research direction.

2. Scientific Literature Review

In this section, we explore the existing literature related to the team-oriented workforce planning process, specifically for team formation and worker assignment in the context of CMS.

2.1. Team Formation

Several qualitative studies have identified a link between teamwork and interpersonal relationships among team members. Stevens and Campion [15] developed the framework of knowledge, skill, and ability (KSA) requirements for teamwork. Their framework reported that healthy interpersonal relationships were present in an effective team. Salas et al. [16] defined teamwork as a dynamic process encompassing team members' thoughts, feelings, and behaviours while interacting toward a common goal. Hoegl and Gemuenden [17] developed a comprehensive concept of the quality of interactions in a team—that is, teamwork quality (TWQ). They argued that high TWQ could not be achieved without an adequate level of group cohesion. Beal et al. [18] conducted a meta-analysis of the relationship between cohesion and group performance. Their analysis implemented three cohesion components, originally introduced by Mullen and Copper [19], namely, interpersonal attraction, task commitment, and group pride. They suggested that the method of sociometry developed by Moreno [20] could be used to indicate the interpersonal attraction of a group.

Numerous studies have implemented sociometry in quantitative methods to represent the interpersonal relationships of team members. Gutiérrez et al. [8] proposed a mathematical model to maximise positive social relationships among workers in each team project. Ballesteros-Pérez et al. [9] defined the degree of team cohesion to maximise positive group interactions and minimise negative group interactions. Chen [10] proposed a mathematical model and solving approach based on sociometry. The author applied the proposed method to optimise relationships of students based on three actual datasets. Campêlo et al. [11] introduced the sociotechnical team formation problem by considering team member skills and interpersonal relationships among team members. The existing literature shows that other methods can also represent the interpersonal relationships of team members. Zakarian and Kusiak [12] applied the analytical hierarchy process (AHP) to prioritise team members based on customer requirements, engineering characteristics of products, and team member preferences. Chen and Lin [7] assessed the working relationship of team members using the Myers–Briggs type indicator (MBTI) with AHP. Zhang and Zhang [13] implemented MBTI to represent the interpersonal relationships of team members in the context of project management. Fathian et al. [14] proposed a mathematical model to maximise interpersonal relationships based on the relationship probability and members' reliability issues.

Rahmanniyay et al. [21] argued that the competency of workers is one of the most critical factors in team formation problems. They proposed a multi-objective multi-stage stochastic programming model to optimise the competencies of team members and staffing costs under uncertainty. Zhang and Zhang [13] addressed the capabilities of team members using implemented fuzzy AHP based on fuzzy linguistic preference relations. Feng et al. [22] proposed criteria for member selection of the cross-functional team, including individual performance and collaborative performance. Despite the effort by researchers to develop the mathematical model for team formation problems, very few quantitative studies have developed models considering both teamwork and worker competencies in a manufacturing system. A pioneering study by Askin and Huang [23] developed a model for the team formation problem in a CMS, considering team synergy and worker job fitness. They implemented the Kolbe conative index (KCI) to measure team synergy and suggested that the KCI and MBTI were suitable for a quantitative model considering teamwork. Fitzpatrick and Askin [24] proposed a heuristic method to solve the team formation problem in a CMS based on the KCI and team skill requirements. Yilmaz et al. [25] implemented a lean principle-based fuzzy methodology to emphasise the importance of cross-functional worker teams by showing that a team-based approach could significantly reduce the lead time and the operational costs in new product development projects.

2.2. Worker Assignment

Workers' competency issues in worker assignment problems have also attracted considerable attention from researchers. Norman et al. [26] investigated the impact of worker and technical skills on the performance of worker assignments in a CMS. Wirojanagud et al. [27] developed a model to determine the amount of hiring, firing, and cross-training for workers at each general cognitive ability level. Süer and Tummaluri [28] proposed a three-phase method to solve the multi-period worker assignment problem in a CMS with worker skill-based operational time. McDonald et al. [29] developed a model for worker assignment in a CMS regarding worker cross-training levels. Aryanezhad et al. [30] developed a dynamic cell formation and worker assignment model that considered worker-task skill requirements. Yilmaz et al. [31] presented an integer programming optimisation model for the batch scheduling and worker assignment problems in a multi-hybrid CMS considering the skilled workforce. The learning-forgetting effect and knowledge transfer of workers have also been extensively studied by researchers [28,32–35].

A CMS typically includes two types of systems—that is, the conventional cellular manufacturing (CCM) system and Japanese cellular manufacturing (JCM) system [5]. Many studies have proposed a solution approach for worker assignment problems in CMS. Liu et al. [36] developed a three-stage heuristic to solve the task worker training plan and the worker-cell assignment. Subsequently, Ying and Tsai [37] proposed a two-phase heuristic algorithm to solve the problems highlighted by Liu et al. [36], their heuristic outperforming that of Liu et al. [36]. Yu et al. [38] and Lian et al. [39] presented a bi-objective worker assignment model for intra- and inter-cell workload balance. Wu et al. [40] and Chu et al. [34] developed a bi-objective worker assignment model to minimise the total cost, including the cost incurred by workload imbalance. Niakan et al. [41] addressed the bi-objective multiperiod cell formation and worker assignment problem considering worker safety issues. Kuo and Liu [42] proposed a two-phase methodology for the worker assignment problem, where worker transfer between cells was allowed. The results showed that the cell-transfer policy could reduce the total number of workers. Feng et al. [43] proposed a heuristic to solve cell formation and worker assignment problems. The author(s) showed that over-assignment could reduce the number of workers hired and improve their utilisation rates.

We summarised the studies considering workforce management related to team formation problems and worker assignment problems in Table 1. Our literature review identified a research gap by showing that quantitative research on team formation and worker assignment problems in a CMS considering teamwork issues remained sparse. Addressing the research gap and the research problem, the following sections present the problem and model formulation, as well as the solving methodologies.

Table 1. Summary of previous studies.

Reference	Problem	Context	Objective	Human Issue	Solution Method
Askin and Huang [23]	TFWAP	CCM	TNC, WPT, and TS	WPT	GHBS
Norman et al. [26]	WAP	CCM	TP	WTHS	MIP
Süer and Tummaluri [28]	WAP	CCM	TNW	LF and WC	THPH
McDonald et al. [29]	WAP	CCM	TC	WCT	BIP
Liu et al. [36]	WAP	JCM	TC	-	THPH
Niakan et al. [41]	WAP	CCM	TC and TPW	WS	Hybrid NSGA-II and MOSA
Ying and Tsai [37]	WAP	JCM	TC	-	TWPH
Kuo and Liu [42]	WAP	CCM	CT and TNW	-	TPIP
Feng et al. [43]	WAP	CCM	TC	-	Hybrid CPSO and LP
Yilmaz et al. [31]	WAP	CCM	ACT	WC	GA, SA, and ABC
Lian et al. [39]	WAP	JCM	WB	-	NSGA-II
Yu et al. [38]	WAP	JCM	WB	-	ECBEA
Wu et al. [40]	WAP	CCM	TC	-	STABCA
Chu et al. [34]	WAP	CCM	TC	LF	AMDSA
Zakarian and Kusiak [12]	TFP	CE	PW	IR	AHP and QFD
Chen and Lin [7]	TFP	CE	TWC and IR	IR	AHP
Zhang and Zhang [13]	TFP	PM	PDTC and IR	IR	AHP and MOPSO
Fathian et al. [14]	TFP	PM	IR	IR	GAMS
Rahmanniyay et al. [21]	TFP	PM	SC and CS	WC	SCDA
Campêlo et al. [11]	TFP	PM	IR	IR	SA
Gutiérrez et al. [8]	TFP	PM	PE	IR	CP, LS, and VNS
Ballesteros-Pérez et al. [9]	TFP	PM	IR	IR	CA
Chen [10]	TFP	SG	IR	IR	GA
Feng et al. [22]	TFP	PM	IP and OCP	WC	NSGA-II
This study	TFWAP	CCM	CS, TIV, and WITV	IR, WCT, FI, and WC	Two-stage ALNS

Note: TFWAP: team formation and worker assignment problem; WAP: worker assignment problem; TFP: team formation problem; CE: concurrent engineering; CCM: conventional cellular manufacturing; JCM: Japanese cellular manufacturing; TNC: training cost; WPT: worker personality traits; TS: team synergy; GHBS: greedy heuristic-based beam search; TP: total profitability; WTHS: worker technical and human skills; MIP: mixed integer programming; TNW: total number of workers; LF: learning–forgetting effect; THPH: three-phase heuristic; TC: total cost; BIP: binary integer programming; NSGA-II: non-dominated sorting genetic algorithm; TPW: total production waste; WS: worker safety; MOSA: multi-objective simulated annealing; TWPH: two-phase heuristic; CT: cycle time; TPIP: two-phase integer programming; CPSO: combinatorial particle swarm optimisation; LP: linear programming; WB: workload balance; ECBEA: epsilon constraint-based exact algorithm; STABCA: superior tracking artificial bee colony algorithm; AMDSA: adaptive memetic differential search algorithm; TIV: total inventory level; WITV: workers idle time variation; IR: interpersonal relationship; FI: fairness issue; WCT: worker cross-training level; WC: worker competency; GA: genetic algorithm; SA: simulated annealing; ABC: artificial bee colony; PW: priority weight; PM: project management; SG: student grouping; PTDC: product development task capabilities; TWC: teamwork capability; SC: staffing cost; CS: competency score; PE: project efficiency; AHP: analytical hierarchy process; QFD: quality function development; MOPSO: multi-objective particle swamp optimization; GAMS: general algebraic modelling system; SCDA: scenario cluster decomposition algorithm; CP: constraint programming; LS: local search; VNS: variable neighbourhood search; CA: computer application; IP: individual performance; OCP: organisational collaborative performance.

3. Theory: Problem Statement and Mathematical Formulation

In the considered problem, we concentrated on grouping cross-trained workers and assigning parts to workers in each manufacturing cell simultaneously. Figure 1 illustrates the overview of the problem. We considered the TFWAP under the CCM system in a single-period operation. Our main focuses are: (1) to form a high TWQ team of skilled workers in each cell to satisfy the part production demands and (2) to assign parts to workers considering minimisation of inventory level and variation of workers' idle time in each cell. We represented the fairness issue by the minimisation of workers' idle time variation. The manufacturing cell consists of part families determined by the predefined cell formation, that is, types of tasks and parts are fixed in each cell. Workers are heterogeneous in terms of skill level and sociometric value. Operations in the cell are manual operations requiring workers' skills to perform the task. Some workers may be skilled in performing different tasks. A skilled worker with task skill coefficient "1" can execute the part at the standard part processing time, whereas an unskilled worker with task skill coefficient less than "1" requires more time than the standard part processing time. Thus, the part processing times of each worker differ depending on their task proficiency. The sociometric values indicate the workers' interpersonal relationships, rated from "1" to "5" as very bad, bad, neutral, good, and very good, respectively. The assumptions of the model were as follows:

1. The part production demand is known and deterministic.
2. The available working time of all workers is equal and deterministic.
3. The amount of time that each assigned worker spends on each part is equal to one hour.
4. Worker walking time between different workstations is neglected due to a compact cell layout.
5. The setup time when workers change task is neglected.
6. The standard part processing time for all parts is known and deterministic.
7. Sociometric matrices of workers are assumed to be known in advance.
8. The skill levels of workers are static, with learning and forgetting effects being neglected.
9. Workers' processing time for each cycle is static, with fatigue effects being neglected.
10. The tools and equipment for all tasks are duplicated; thus, workers can operate parts together in the same task.
11. The number of workers in a cell is less than or equal to the number of tasks in the cell.
12. Working overtime and cell transfer are prohibited.
13. Minimum cell-cohesion levels are required in each cell.
14. All part production demands have to be satisfied.

This paper presents two mathematical models—namely, Model A and Model B. Model A is a single-stage model that considers only the worker-part assignment problem. Model B is a two-stage model consisting of the two models B1 and B2. Model B1 is the team formation problem that intends to form a skilled workforce with a good relationship among team members in each cell. Model B2 is the worker assignment problem of each cell. The idea behind Model B1 is to assure Model B2 with skilled workers in satisfying part production demands. We developed Model B with the intention to clarify that the TFWAP is more suitable for the two-stage model. Owing to the model assumptions, both models are nonlinear using penalty functions.

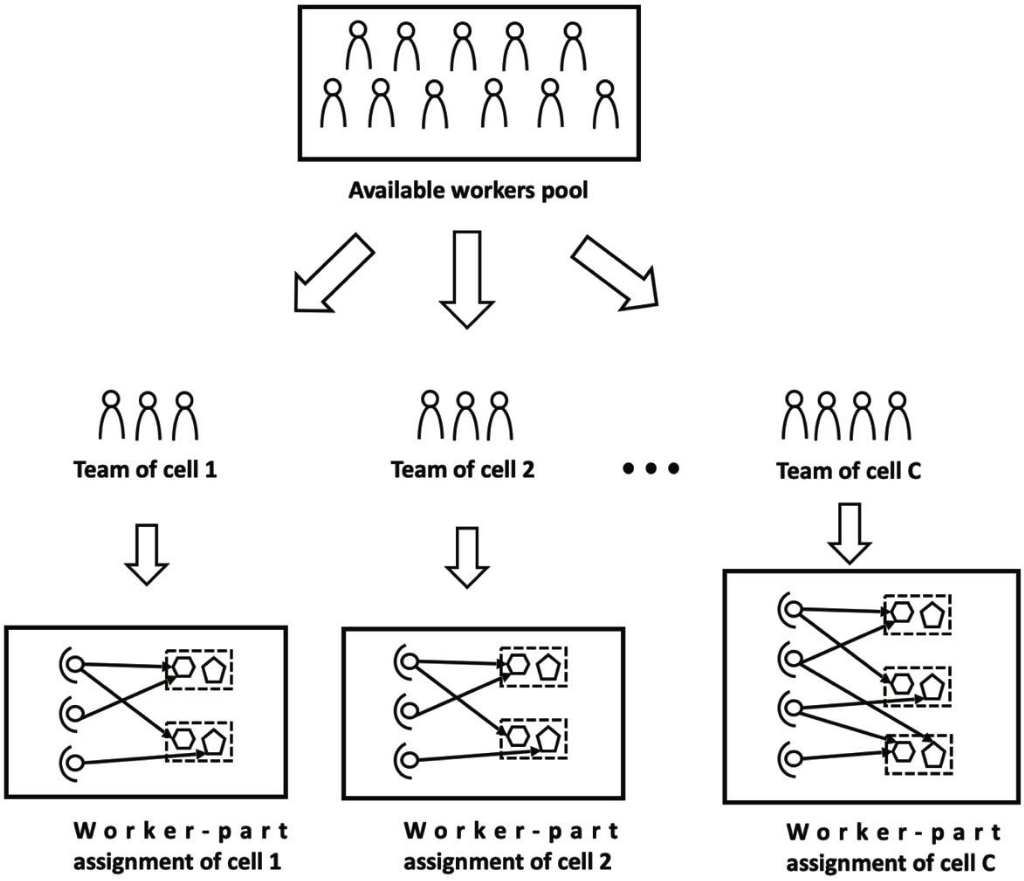

Figure 1. Illustration of TFWAP.

3.1. Notations for Model A and Model B

The notations for the addressed TFWAP are presented as follows:

Indices	
c	Index of cells ($c = 1, 2, \ldots, C$)
w	Index of workers ($\mathrm{w} = 1, 2, \ldots, W$)
p	Index of parts ($p = 1, 2, \ldots, P$)
t	Index of tasks ($\mathrm{t} = 1, 2, \ldots, T$)
k	Index of penalty functions ($\mathrm{k} = 1, 2, \ldots, K$)
Input parameters	
ST_p	Standard processing time of part p (seconds)
D_p	Demand of part p
A	Worker available working hours
α_{wt}	Skill coefficient of worker w at task t
r_{uv}	The sociometric value of worker u and worker v; $u, v \in W; u \neq v$
L	Cell-cohesion requirement
n	Minimum sociometric score
o	Maximum sociometric score
j	Penalty coefficient
q_{pt}	If part p belongs to task t, 1; otherwise, 0
h_{pc}	If part p belongs to cell c, 1; otherwise, 0
g_{tc}	If task t belongs to cell c, 1; otherwise, 0
s_{wt}	If worker w is competent at task t, 1; otherwise, 0
Variables	
I_{wc}	Idle time of worker w in cell c
COH_c	Total cohesion of cell c

3.2. Model A Formulation

In addition to the notation summarised in Section 3.1, Model A can be formulated using two additional variables, as follows:

Variable	
y_{wc}	If worker w works in cell c, 1; otherwise, 0
Decision variable	
x_{wp}	If worker w operates part p for one hour, 1; otherwise, 0

The objective functions and penalty functions of Model A can be expressed as follows:

$$\min Z_1 = \sum_{p=1}^{p} \left(\sum_{w=1}^{W} \frac{3600 \cdot q_{pt} \cdot x_{wp} \cdot \alpha_{wt}}{ST_p} - D_p \right) + \sum_{k=1}^{5} pen_k \tag{1}$$

$$\min Z_2 = \frac{1}{C} \sum_{c=1}^{C} \mathrm{IVR}_c + \sum_{k=1}^{5} pen_k \tag{2}$$

$$pen_1 = \sum_{p=1}^{P} \max\left(j \cdot \left(D_p - \sum_{w=1}^{W} \frac{3600 \cdot q_{pt} \cdot x_{wp} \cdot \alpha_{wt}}{ST_p} \right), 0 \right) \tag{3}$$

$$pen_2 = \max\left(\sum_{w=1}^{W} \left(j \cdot \left(\sum_{p=1}^{P} x_{wp} - A \right) \right), 0 \right) \tag{4}$$

$$pen_3 = j \cdot \sum_{w=1}^{W} \left(\sum_{c=1}^{C} y_{wc} - 1 \right) \tag{5}$$

$$pen_4 = j \cdot \sum_{c=1}^{C} \max(L - normCOH_c, 0) \tag{6}$$

$$pen_5 = j \cdot \sum_{c=1}^{C} \max\left(\left(\sum_{w=1}^{W} y_{wc} - \sum_{t=1}^{T} g_{tc} \right), 0 \right) \tag{7}$$

subject to,

$$IVR_c = \frac{1}{\overline{I}_c} \sqrt{\frac{\sum_{w=1}^{W} \left(I_{wc} - \overline{I_c} \right)^2}{\sum_{w=1}^{W} y_{wc}}} \quad \forall c \tag{8}$$

$$\overline{I}_c = \frac{1}{\sum_{w=1}^{W} y_{wc}} \sum_{w=1}^{W} I_{wc} \quad \forall c \tag{9}$$

$$I_{wc} = A - \sum_{P=1}^{P} x_{wp} \cdot h_{pc} \cdot s_{wt} \cdot q_{pt} \quad \forall c, t, w \tag{10}$$

$$y_{wc} = x_{wp} \cdot h_{pc} \quad \forall w, c, p \tag{11}$$

$$COH_c = \sum_{u,v \epsilon W} (r_{uv} \cdot y_{uc} \cdot y_{vc}) \quad \forall c, p \tag{12}$$

$$maxCOH_c = o \cdot \sum_{u,v \epsilon W} (y_{uc} \cdot y_{vc}) \quad \forall c, p \tag{13}$$

$$minCOH_c = n \cdot \sum_{u,v \epsilon W} (y_{uc} \cdot y_{vc}) \quad \forall c, p \tag{14}$$

$$normCOH_c = \frac{COH_c - minCOH_c}{maxCOH_c - minCOH_c} \quad \forall c, p \tag{15}$$

$$x_{wp}, y_{wc} \in 0, 1 \tag{16}$$

$$I_{wc}, COH_c,\ maxCOH_c,\ minCOH_c\ \in INT^+ \tag{17}$$

$$IVR_c, \overline{I}_c, normCOH_c\ \in\ R^+ \tag{18}$$

Equations (1) and (2) represent the two objective functions: the total inventory level and the average cell worker's idle time variation of all cells. Equations (1) and (2) can be calculated by considering the summation of the penalty functions of Equations (3)–(7). Equation (1) can be calculated using the sum of the parts produced by all workers for each part and its demand. Equation (2) can be calculated using the cell worker's idle time variation of cell c from Equation (8). Equation (9) shows the average worker idle time in cell c. Equation (10) calculates the idle time of worker w in cell c. Equation (3) represents the unsatisfied part of the demand penalty function. If the solution cannot satisfy the demand at part p, then the penalty value of part p becomes positive. Otherwise, the penalty value for part p is zero. Equation (4) represents the working overtime penalty. The summation of the worker's working hours must not exceed the available working hours, A. If the working hours of worker w exceeds A, the penalty value of worker w can be calculated. Equation (5) addresses the cell-transfer penalty by considering Equation (11), which ensures that worker w works in cell c if worker w operates part p which belongs to cell c. Equation (6) represents the summation of the cohesion penalties using Equations (12)–(15). Equation (12) represents the total cohesion value of cell c and calculates the possible maximum total cohesion score of cell c. Equation (13) calculates the possible minimum cohesion score of cell c. Equation (15) calculates the normalised cohesion score of cell c. Equation (7) penalises the solution when the total number of workers in cell c exceeds the number of total tasks in cell c. Equation (16) defines the binary variables. Equation (17) defines the positive integer variables. Finally, Equation (18) defines the positive real-number variables.

3.3. Model B1 Formulation: Team Formation

Model B1 aims to form a team of workers in each cell to ensure that part production demands can be satisfied in the worker assignment stage. Based on this idea, we proposed a new objective function for part-skill scores. The part-skill score objective consists of the summation of the total worker capabilities and the weight of each part production demands in each cell. Campbell [44] motivated us to consider team capabilities to ensure that there are enough skilled workers in the cell. As forming an effective team in the cell is the main concern, two additional constraints are introduced in this model. With two additional notations to those summarised in Section 3.1, we can formulate Model B1 as follows:

Input Parameter	
dem_p	If demand of part p is a positive integer, 1; otherwise, 0
Decision variable	
z_{wc}	If worker w is assigned to cell c, 1; otherwise, 0

The objective functions and penalty functions of Model B1 can be expressed as follows:

$$\max Z_1 = \sum_{c=1}^{C}\left(\sum_{p=1}^{P}\left(\frac{h_{pc}\cdot D_p}{\sum_{p=1}^{P}(h_{pc}\cdot D_p)}\cdot\sum_{w=1}^{W}(z_{wc}\cdot s_{wt}\cdot q_{pt}\cdot g_{tc})\right)\right)-\sum_{k=1}^{5}pen_k \quad (19)$$

$$pen_1 = \max\left(\sum_{p=1}^{P}\left(j\cdot\left(1-\sum_{w=1}^{W}(z_{wc}\cdot s_{wt}\cdot q_{pt}\cdot g_{tc}\cdot dem_p)\right)\right),0\right) \quad (20)$$

$$pen_2 = j\cdot\sum_{c=1}^{C}\max(L-normCOH_c,0) \quad (21)$$

$$pen_3 = j\cdot\max\left(\sum_{c=1}^{C}\left(\sum_{w=1}^{W}z_{wc}-\sum_{t=1}^{T}g_{tc}\right),0\right) \quad (22)$$

$$pen_4 = j\cdot\sum_{w=1}^{W}\left(\sum_{c=1}^{C}z_{wc}-1\right) \quad (23)$$

$$pen_5 = j\cdot\max\left(\sum_{w=1}^{W}\left(1-\sum_{c=1}^{C}z_{wc}\right),0\right) \quad (24)$$

subject to,

$$COH_c = \sum_{u,v\epsilon W}(r_{uv}\cdot z_{uc}\cdot z_{vc}) \quad \forall c \quad (25)$$

$$maxCOH_c = o\cdot\sum_{u,v\epsilon W}(z_{uc}\cdot z_{vc}) \quad \forall c \quad (26)$$

$$minCOH_c = n \cdot \sum_{u,v \epsilon W} (z_{uc} \cdot z_{vc}) \quad \forall c \tag{27}$$

$$normCOH_c = \frac{COH_c - minCOH_c}{maxCOH_c - minCOH_c} \quad \forall c \tag{28}$$

$$z_{wc} \in 0,1 \tag{29}$$

$$maxCOH_c,\ minCOH_c,\ COH_c \in INT^+ \tag{30}$$

$$normCOH_c \in\ R^+ \tag{31}$$

Equation (19) shows the part-skill score objective function calculated using the summation of the product of part p demand weight and the total number of worker skills at part p of each cell. Equations (20)–(24) are the penalty functions of Model B1, which are considered in Equation (19). Equation (20) calculates the additional penalty when there is no skilled worker in cell c to operate the task of the demanded part p. Equation (21) represents the penalty of cell c cohesion by taking Equations (25)–(28) into account, similar to Equation (6) of Model A. Equation (22) penalises the solution when the number of workers in cell c exceeds the number of total tasks in cell c. Equation (23) represents the cell-transfer penalty. Equation (24) represents the additional penalty for unassigned workers. Equation (29) represents the binary decision variable in Model B1. Equation (30) represents the positive integer variable in Model B1. Finally, Equation (31) represents the positive real-number variable in Model B1.

3.4. Model B2 Formulation: Worker Assignment

Model B2 has two objectives—that is, minimising the total inventory level and minimising the cell workers' idle time variation in each cell. In addition to the notations summarised in Section 3.1, two additional notations are introduced in model B2. The formulation of Model B2 can be expressed as follows:

Input Parameter	
a_{wc}	If worker w belongs to cell c, 1; otherwise, 0
Decision variable	
x_{wp}	If worker w is assigned to part p, 1; otherwise, 0

The objective functions and penalty functions of Model B2 can be expressed as follows:

$$\min Z_2 = \sum_{p=1}^{P} \left(\sum_{w=1}^{W} \frac{3600 \cdot q_{pt} \cdot x_{wp} \cdot \alpha_{wt}}{ST_p} - D_p \right) + \sum_{k=1}^{3} pen_k \tag{32}$$

$$pen_1 = \sum_{p=1}^{P} \max\left(j \cdot \left(D_p - \sum_{w=1}^{W} \frac{3600 \cdot q_{pt} \cdot x_{wp} \cdot \alpha_{wt}}{ST_p} \right), 0 \right) \tag{33}$$

$$pen_2 = \max(j \cdot (IVR_c - \epsilon), 0) \tag{34}$$

$$pen_3 = \max\left(\sum_{w=1}^{W} \left(j \cdot \left(\sum_{p=1}^{P} x_{wp} - A \right) \right), 0 \right) \tag{35}$$

subject to,

$$IVR_c = \frac{1}{\overline{I}_c} \sqrt{\frac{\sum_{w=1}^{W} \left(I_{wc} - \overline{I_c} \right)^2}{\sum_{w=1}^{W} a_{wc}}} \quad \forall c \tag{36}$$

$$\overline{I}_c = \frac{1}{\sum_{w=1}^{W} a_{wc}} \sum_{w=1}^{W} I_{wc} \quad \forall c \tag{37}$$

$$I_{wc} = A - \sum_{P=1}^{P} x_{wp} \cdot h_{pc} \cdot s_{wt} \cdot q_{pt} \quad \forall c, t, w \tag{38}$$

$$x_{wp} \in 0,1 \tag{39}$$

$$I_{wc} \in INT^{+} \tag{40}$$

$$IVR_c, \bar{I}_c \in R^{+} \tag{41}$$

Equation (32) represents the objective function of Model B2, which is the total inventory level. Equation (33) represents the unsatisfactory demand penalty function. Equation (34) represents the epsilon constraint to minimise the cell workers' idle time variation of cell c. Equation (35) represents the overtime working penalty of worker w. Equations (36)–(38) can be used to calculate Equation (34). Equation (39) represents the binary decision variable. Equation (40) represents the positive integer variable. Finally, Equation (41) represents the positive real-number variables.

4. Methodology: Problem-Solving Methodology

The worker assignment and team formation problems are combinatorial problems that generally cannot be solved in polynomial time [8,39]. Pisinger and Ropke [45] suggested that ALNS methods should be implemented as a standard framework for solving large-sized optimisation problems. ALNS has been successfully implemented with the vehicle routing problem (VRP) [46–50]. Several researchers have developed ALNS algorithms to solve other problems [51–56]. However, an ALNS-based solution has not yet been proposed for the workforce planning problem. Solving TFWAP presents a difficulty in handling both team formation and worker assignment decisions simultaneously, even for small-sized problem instances. Therefore, we proposed a two-stage ALNS framework based on a two-stage decision methodology. The other two solution approaches—namely, the heuristic-based brute force search (HBBFS) and non-dominated sorting genetic algorithm-II (NSGA-II) methods—were developed as comparison benchmarks to verify the effectiveness of the two-stage ALNS method. Details of the HBBFS and NSGA-II methods are provided in the Supplementary Materials.

4.1. Two-Stage Adaptive Large Neighbourhood Search (Two-Stage ALNS)

The two-stage ALNS method applies modified ALNS techniques in each stage, the solution for Stage1 being the input to Stage2. In Stage2, the search starts from the first cell to the last cell. Finally, the heuristic calculates the total inventory level and average value of the cell worker's idle time variation. The overall mechanism of the two-stage ALNS method is shown in Figure 2.

We made two modifications to the ALNS method. First, problem-specific operators were used. We modified the destroy and repair operators for both stages. Second, we proposed a new adaptive mechanism to tune the degree of destruction when the solution showed no improvement for a predetermined number of iterations. Based on Algorithm 1, the modified ALNS method can be described as follows: The initial solution $(s_{initial})$ can be obtained from the initial heuristic. To begin with, all the operator scores are set as the initial score (ω_0). The roulette-wheel procedure chooses one destroy operator and one repair operator based on its past performance. Operators are awarded scores depending on their current performance. Once operators are selected, they are successively applied in the current iteration to generate a new solution known as the repaired solution $(s_{repaired})$. The global solution (s_{best}) can only be updated if the heuristic obtains a better solution. With the mechanism of embedded simulated annealing, $s_{repaired}$ without improvement may be accepted as a new current solution $(s_{current})$ to escape the local optima.

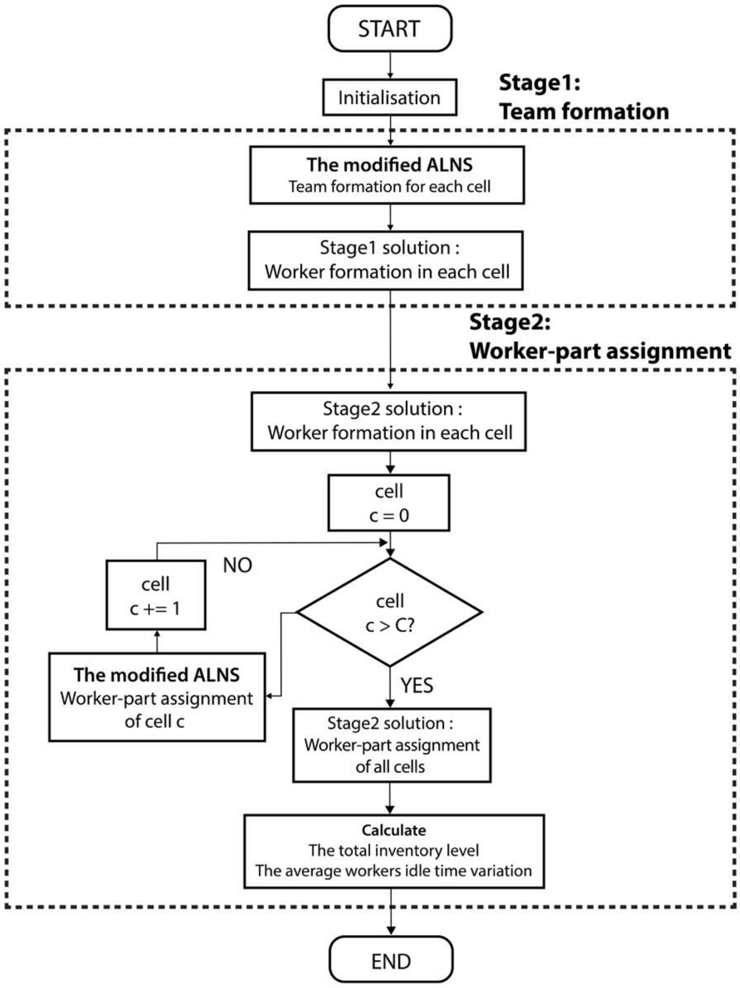

Figure 2. Flowchart of the proposed two-stage ALNS.

Algorithm 1 The modified ALNS

Input: Problem instances, the modified ALNS parameters, and initial solution $(s_{initial})$
Set $s_{current} \leftarrow s_{initial}$, $s_{best} \leftarrow s_{initial}$, $\omega \leftarrow \omega_0$, $temp \leftarrow temp_0$, $and\ i \leftarrow 0$
while $(i < I_{max})$ **do**
Select destroy operator and repair operator
Remove the partial solution from $s_{current}$
Reinsert the partial solution to $s_{current}$
  if $(s_{current} < s_{repaired})$ **then**
    Set $s_{best} \leftarrow s_{repaired}$ and $s_{current} \leftarrow s_{repaired}$
  else
    if $(s_{current} < s_{repaired})$ **then**
      Set $s_{current} \leftarrow s_{repaired}$
    else
      if $s_{repaired}$ is accepted by SA **then**
        Set $s_{current} \leftarrow s_{repaired}$
      end if
    end if
  end if
  Update ω of all operators and $temp$
  if $(i < i_{checkimp})$ **then**
    if $(s_{best}$ of $i < s_{best}$ of $i_{checkimp})$ **then**
      Randomly choose des
    end if
  end if
  Set $i \leftarrow i + 1$
end while
return s_{best}

4.1.1. Initial Heuristic of the Proposed Two-Stage ALNS

The initial heuristic of Stage1 and Stage2 search for the solution one cell at a time. The Stage1 initial heuristic works as follows: To begin with, the heuristic creates two empty worker lists for unassigned workers (UW) and total assigned workers (AW). In each cell, the heuristic generates the list $(SKILL_c)$ representing the total part-skill of each available worker. Next, the heuristic chooses the first worker of cell c by selecting the one who possesses the maximum number of skills in cell c from $SKILL_c$. Consequently, the next worker is chosen based on their interpersonal relationship with the previously selected worker. The heuristic assigns workers to a cell until the total number of workers reaches the total number of tasks in the cell. The overall framework of the Stage1 initial heuristic is presented in Algorithm 2.

The input of the Stage2 initial heuristic is the team formation solution of Stage1. The Stage2 initial heuristic works as follows: Workers are sorted by the total part skills of cell c in ascending order, denoted as wkr_c. At the same time, parts of cell c are sorted by their demand in descending order, denoted as $Pdem_c$. Workers in wkr_c are assigned to parts in $Pdem_c$. If worker w does not possess the skill of part p, then the heuristic moves to the next part. If the remaining working time of worker $(wrkt_w)$ is zero, the heuristic will skip worker w. After assigning worker w to part p, the heuristic updates the cumulative production of part $\left(prod_p\right)$ and $wrkt_w$. The heuristic also updates the number of total demand-satisfied parts $(Psat_c)$ when $prod_p$ exceeds or equalises the part production demand (dem_p). The heuristic prioritises the least skilled worker for the following reasons: prioritising the assignment of workers by the number of worker-part skills in descending order cannot satisfy the part production demands as the heuristic intends to satisfy all the part production demands of the current cell before moving to the next cell. The mechanism of the Stage2 initial heuristic is presented in Algorithm 3.

Algorithm 2 Stage1 initial heuristic

```
Input : Problem instances, the modified ALNS parameters, and initial solution s_initial
Initialise UW and AW
Set number of workers in the cell N_c ← 0 , UW ← 0, and AW ← 0
for each c ∈ C of s_initial1 do
  generate SKILL_c
  while (i < I_max) do
    if (N_c = 0 and UW ≠ 1) then
      Select w ∈ UW with respect to max(SKILL_c)
    else
      if (N_c ≥ 1 and UW ≠ 1) then
        Select w ∈ UW with respect to max(r_{w,w−1})
      else
        if (UW = 1) then
          Select w ∈ UW
        end if
      end if
    end if
    Set N_c ← N_c + 1
    Append w to AW
    Remove w from UW
  end while
end for
return s_initial1
```

Algorithm 3 Stage2 initial heuristic

```
Input : Problem instances and solution of worker formation in each cell S_1
Initialise Pdem_c, wkr_c, and Pdp_c of s_initial2
Set Psat_c ← 0
for each c ∈ C of s_initial2 do
  while (Psat_c < Pdp_c) do
    for each w ∈ wkr_c do
      for each p ∈ Pdem_c do
        if (skill_wp = 0) then
          continue
        else
          if (dem_p > prod_p and wkrt_w > 0) then
            Assign worker w to part p
            Update dem_p, prod_p and wkrt_w
          else
            if (dem_p ≤ prod_p) then
              Set Psat_c ← Psat_c + 1
              continue
            else
              if (wrkt_w = 0) then
                break
              end if
            end if
          end if
        end if
      end for
    end for
  end while
end for
return s_initial2
```

4.1.2. Adaptive Search Engine

The adaptive mechanism is the main characteristic of the modified ALNS method—that is, the modified ALNS method can explore and exploit a new search space for a new solution owing to its adaptive ability. Here, we present two adaptive search engines as follows:

1. Score operator: The score operator selects the destroy and repair operators using a roulette wheel based on their past performance, which can be expressed as follows:

$$\frac{\omega_j}{\sum_{i=1}^{k} \omega_i} \tag{42}$$

   where

   ω_j = Score of the selected operator.
   $\sum_{i=1}^{k} \omega_i$ = Score of all operators.
2. Degree of the destruction operator: The operator intensifies the search for a new space. Based on Algorithm 1, the modified ALNS method checks the solution improvement at every predetermined, fixed number of iterations.

4.1.3. Destroy Operator

Six destroy operators were customised in the proposed two-stage ALNS method. The destroy operator removes cells/parts from the current solution until a predefined number of removed cells/parts is achieved. The number of removed cells per part can be expressed as follows:

$$\rho = des \cdot R \tag{43}$$

where

ρ = Number of removed cells/parts.
des = Degree of destruction.
R = Number of total cells/parts.

At every predetermined iteration $i_{checkimp}$, des is tuned when the current solution shows no improvement. The destroy operators of Stage1 and Stage2 are as follows:

1. Random cell removal D_{1-1}: The operator randomly removes the cell until the predefined number of removed cells is reached.
2. Worst cell removal D_{1-2}: The operator removes the cell that contributes the least to the Model B1 objective value one at a time until the predefined number of removed cells is reached.
3. Shaw cell removal D_{1-3}: This removal operation was originally proposed by Shaw [57]. The operator randomly selects the cell and calculates its relatedness Φ_{ij} with the other cells. Next, the operator randomly removes one cell from the pair with the highest relatedness score. The cell relatedness can be determined using Equation (44). If there is no relatedness between all pairs, the operator randomly removes the cell from the current solution.
$$\Phi_{ij} = A_i^T \cdot A_j \tag{44}$$
where
Φ_{ij} = Relatedness value of cell/part i and cell/part j.
A_i^T = Transposed array of the first selected cell/part i.
A_j = Cell/part array j.
4. Random part removal D_{2-1}: The operator randomly removes parts until the predefined number of removed parts is met.
5. Worst-part removal D_{2-2}: The operator removes the parts that contribute the least to the Model B2 objective value one at a time until the predefined number of removed parts is met.
6. Shaw part removal D_{2-3}: This operator works similar to D_{1-3} operator.

4.1.4. Repair Operator

Seven repair operators were customised for the proposed two-stage ALNS method. The main characteristic of the repair operator is cell-transfer prevention. Before removing each cell from the current solution, the repair operator prevents cell-transfer situations by omitting the already assigned workers in each removed cell.

1. Random cell insertion R_{1-1}: The operator randomly selects the removed cells one at a time. After selecting the cell, it randomly selects the available workers for the removed cell until the number of workers in the cell reaches the total number of tasks. The procedure is repeated until all removed cells are reinserted into the current solution.
2. Greedy-B cell insertion R_{1-2}: The operator sequentially selects the removed cell to repair and reinsert into the current solution, one at a time, based on the ascending order of the number of total tasks. In each removed cell, the operator selects the available workers as follows: The operator generates B worker combinations, where the total number of workers in each combination is equal to T_c. The operator chooses the combination of workers that contributes most to the Model B1 objective value.
3. Regret-B cell insertion R_{1-3}: The operator works similarly to repair operator R_{1-2} by generating B combinations of workers in each removed cell. In addition to the mechanism of R_{1-2}, the operator applies look-ahead information by calculating the regret value reg_c of all B worker combinations as follows:
$$reg_c = \sum_{combi=1}^{B} \left(f_{1-combi} - f^*_{1-combi} \right) \tag{45}$$
where
$f^*_{1-combi}$ = Model B1 objective value when placing the removed cell back to the current solution with the best worker combination.
$f_{1-combi}$ = Model B1 objective value when placing the removed cell back to the current solution with the worker combination, starting from the second-best combination to the last combination.

4. Workers idle time variation based on part insertion R_{2-1}: The operator aims to minimise the cell workers' idle time variation in each cell by assigning workers to parts in a distributional manner. Initially, the heuristic generates a list of sorted removed parts with respect to their demands, in descending order. For each removed part, the skilled worker with the highest remaining working time is prioritised for the assignment. The heuristic stops assigning workers to the removed part after the demand is satisfied. The overall framework of R_{2-1} heuristic can be found in the Supplementary Materials.
5. Pseudorandom part insertion R_{2-2}: At the beginning of the repair operation, the operator creates a list of skilled workers, whose remaining working time is still available for part assignment, denoted as Wlist. The operator randomly assigns workers from the Wlist to the part until part production demand is satisfied.
6. Pseudorandom greedy-B part insertion R_{2-3}: The operator reinserts the removed part into the current solution one at a time. In each repair procedure, the operator creates the list Wlist for worker assignments for each part. The operator then generates B worker assignment combinations for each removed part. Workers in each combination are chosen from Wlist. Finally, the operator chooses the combination of workers that contributes most to the Model B2 objective value.
7. Pseudorandom regret-B part insertion R_{2-4}: This operator works similarly to operator R_{2-3}. In addition to the procedure of R_{2-3}, the operator selects the worker assignment combination based on regret value (reg_p), which is calculated as follows:

$$reg_p = \sum_{combi=1}^{B} \left(f_{2-combi} - f^*_{2-combi} \right) \tag{46}$$

where

$f^*_{2-combi}$ = Model B2 objective value when placing the removed part back to the current solution with the best worker assignment combination.
$f_{2-combi}$ = Model B2 objective value when placing the removed part back to the current solution with the worker assignment combination, starting from the second-best combination to the last combination.

4.1.5. Simulated Annealing (SA)

We embedded the simulated annealing (SA) algorithm at the top level of the modified ALNS algorithm to avoid the solution being trapped at a local optima. SA accepts an $s_{repaired}$ value that is worse than $s_{current}$ with the acceptance probability defined by Equation (47) for Stage1 and Equation (48) for Stage2. The calculation of annealing temperature (temp) can be expressed as shown in Equation (49):

$$temp = temp_0 \cdot \gamma \tag{47}$$

$$prob_1 = e^{\frac{F_{repaired} - F_{current}}{temp}} \tag{48}$$

$$prob_2 = e^{\frac{F_{current} - F_{repaired}}{temp}} \tag{49}$$

where

$temp$ = Annealing temperature; $temp > 0$.
$temp_0$ = Initial annealing temperature.
γ = Cooling rate; $0 < \gamma < 1$.
$prob_1$ = Probability acceptance of Stage1.
$prob_2$ = Probability acceptance of Stage2.
$F_{current}$ = Objective value of the current solution.
$F_{repaired}$ = Objective value of the repaired solution.

5. Numerical Example

We provided a numerical example to further illustrate the application of the proposed formulations in Section 3 and verify our proposed method. The example is based on a CCM system consisting of two manufacturing cells, four tasks, and four partial cross-trained workers. Each cell is dedicated to a family of parts. The input data were categorised into two categories: (1) part production data and (2) workforce data. The part production data include the details of cell formation, standard part processing time, and part production demand as shown in Table 2. Workforce data are the information of workers, including task proficiency and sociometric value, as displayed in Tables 3 and 4.

Table 2. Part information represented in the cell formation structure for the example problem.

	Part Information (D_p, ST_p)				
Tasks	**Part A**	**Part B**	**Part C**	**Part D**	**Part E**
1	(40,50)	(20,50)	-	-	-
2	(0,50)	(30,40)	-	-	-
3	-	-	(40,40)	(0,40)	(20,50)
4	-	-	(0,40)	(40,40)	-

Table 3. Workforce skill coefficient for the example problem.

	Worker Skill Coefficient (α_{wt})			
	Tasks			
Worker w	**1**	**2**	**3**	**4**
1	0.77	1.25	0.77	-
2	1.00	-	1.00	1.00
3	1.11	-	1.43	0.91
4	1.11	1.43	1.25	-

Table 4. Workforce sociometric value for the example problem.

	Sociometric Value (r_{uv})			
	Worker v			
Worker u	**1**	**2**	**3**	**4**
1	-	5	1	5
2	-	-	2	1
3	-	-	-	5
4	-	-	-	-

Initially, we pre-processed the part production data from the given cell formation to each part indices. Task 1 contained parts {1,2}, task 2 contained parts {3,4}, task 3 contained parts {5,6,7}, and task 4 contained parts {8,9}. Cell 1 contained parts {1,2,3,4} and cell 2 contained parts {5,6,7,8,9}. Once the part information of the proposed Model A and Model B are pre-processed, the single-stage NSGA-II and two-stage ALNS are implemented to solve each model. First, we defined the solution representation in each algorithm. We implemented the binary value 1 or 0 to denote two decisions: (1) whether a worker is chosen for the team in the cell and (2) whether a worker is chosen to execute the part.

Due to the hard constraints presented in TFWAP, the final solution obtained from the single-stage methodology will likely deliver an infeasible solution. As shown in Tables 5–7, it is evident that the single-stage methodology is not well-suited for handling the constraints of the TFWAP. The solution is infeasible and penalised by two static penalties of Equations (5) and (7), including the cell transfer of worker 1 and worker 4 and the number of total workers in cell 2 exceeding the total number of tasks. These two penalties belong to the constraints of the team formation problem, as shown in Model B1 formulation. Unlike the two-stage ALNS, the single stage NSGA-II did not implement the decomposition strategy that reduces the search space of the TFWAP by treating the team formation as a subproblem. Without the decomposition strategy, the algorithm will explore a larger search space and hardly obtain a feasible solution. Therefore, there is a high possibility that a set of team formation constraints cannot be satisfied in the final solution. Table 7 displays a solution obtained from the single-stage NSGA-II as a two-dimensional chromosome of the numerical example. The calculation of the solution in Table 7 can be found in the Supplementary Materials.

Table 5. Solution of worker-part assignment obtained by NSGA-II for the illustrative example.

	Cell 1					Cell 2			
Worker w	**Part 1**	**Part 2**	**Part 3**	**Part 4**	**Part 5**	**Part 6**	**Part 7**	**Part 8**	**Part 9**
1	0	1	0	1	1	0	0	0	0
2	0	0	0	0	1	0	1	0	1
3	0	0	0	0	0	0	0	0	0
4	1	0	0	0	0	0	1	0	0

Table 6. Penalty values obtained by using NSGA-II for the illustrative example.

Penalty	**Penalty Value**
1. Demand unfulfillment	0
2. Working overtime	0
3. Worker cell transfer	$2 \cdot 1,000,000$
4. Cell-cohesion requirement	0
5. Workers > Tasks in cell	$1 \cdot 1,000,000$
Total penalty value	3,000,000

Table 7. Solution obtained by using NSGA-II for the illustrative example.

Parameter	**Description**	**Value**
Z_1	The total inventory level	3,000,467
Z_2	The average cell worker's idle time variation of all cells	3,000,000.134
$normCOH_{avg}$	The normalised average cohesion of all cells	0.833

To handle the constraints of the TFWAP, we proposed the two-stage ALNS that decomposed the TFWAP into two subproblems and embedded the problem-oriented constraint handling heuristics. Tables 8–12 display the obtained feasible solution using the two-stage ALNS. At the team formation stage, each cell contained a highly cohesive work team where cell 1 contained workers {1,2} and cell 2 contained workers {3,4}. The two-stage ALNS guaranteed that each cell contains workers with the skills to execute parts. At the worker-part assignment stage, only worker 1 was assigned to operate parts {1,2,4} in cell 1. Worker 3 was assigned to operate part {3} and worker 4 was assigned to operate parts {5,7}. Table 12 shows the solution of the example obtained by the two-stage ALNS method. The calculation of the solution in Table 12 is shown in the online Supplementary Materials.

Table 8. Solution of team formation stage obtained by using two-stage ALNS for the illustrative example.

	Cells	
Worker w	**Cell 1**	**Cell 2**
1	1	0
2	1	0
3	0	1
4	0	1

Table 9. Solution of worker-part assignment in Cell 1 obtained by using two-stage ALNS for the illustrative example.

Worker w	Part 1	Part 2	Part 3	Part 4
1	0	0	0	0
2	1	1	0	1

Table 10. Solution of worker-part assignment in Cell 2 obtained by using two-stage ALNS for the illustrative example.

Worker w	Part 5	Part 6	Part 7	Part 8	Part 9
3	0	0	0	0	1
4	1	1	0	1	0

Table 11. Penalty values obtained by using two-stage ALNS for the illustrative example.

Penalty	Penalty Value
1. No skilled worker in the cell	0
2. Cell-cohesion requirement	0
3. Workers > Tasks in cell	0
4. Worker cell transfer	0
5. No skilled workers in the cell	0
6. Demand unfulfillment	0
7. Epsilon constraint	0
8. Working overtime	0
Total penalty value	0

Table 12. Solution obtained by using two-stage ALNS for the illustrative example.

Parameter	Description	Value
Z_1	The total part-skill score	3.265
Z_2	The total inventory level	315
IVR_{avg}	The average cell worker's idle time variation of all cells	0.182
$normCOH_{avg}$	The normalised average cohesion of all cells	1.000

6. Results: Computational Experiment

We assessed the quality of the solution and validated the effectiveness of the proposed two-stage ALNS method via a series of experiments. We first conducted two performance comparison experiments by comparing the solution of the two-stage ALNS with the HBBFS and the NSGA-II methods on the same input dataset. Finally, we applied the two-stage ALNS method to study the impact of cell-cohesion requirements on the ability to form a team of high skilled, cross-trained workers. All algorithms were coded using the Python programming language version 3.7.12 (Guido van Rossum: Amsterdam, Netherlands) and run on a PC with an Intel Xeon CPU (2.20 GHz) with 13 GB of RAM.

6.1. Problem Instance and Parameter Settings

We generated the input dataset for the TFWAP using ten cell formation settings selected as problem benchmarks from the literature, as shown in Table 13. The problem size varies based on the total number of parts, tasks, cells, and workers. The part production and cross-trained workforce data in each problem benchmark were heterogeneous and randomly generated based on the pattern in Table 14. We created four conditions of cell operational requirements for the performance comparison experiment. Each condition included the combination of cross-training level and cell-cohesion score requirements. The cross-training level represents the percentage of total tasks that each worker in a team can execute, which are 60% and 80%. The cell-cohesion score requirement required a team to have at least 0.3 and 0.6 of the normalised cell-cohesion score. To deliver insights regarding team formation, we constructed the three-level factorial design, which had two factors—the cross-training level and the cell-cohesion score requirement, each at three levels, as shown in Table 14. The algorithm-related input parameters can be found in Table 15. More details about the problem benchmarks and the input dataset can be found in a public GitHub repository.

Table 13. Problem benchmarks.

		Number of			
Problem ID	**Author**	**Parts**	**Tasks**	**Cells**	**Workers**
P1	Won and Kim [58]	9	4	2	4
P2	Won and Kim [58]	15	6	2	6
P3	Zolfaghari and Liang [59]	21	7	3	7
P4	Islam and Saker [60]	29	8	3	8
P5	Yang and Yang [61]	41	10	3	10
P6	Won and Kim [58]	28	11	4	11
P7	Moon and Chi [62]	53	12	4	12
P8	Seifoddini and Djassemi [63]	86	35	5	35
P9	Won and Kim [58]	124	26	6	26
P10	Seifoddini and Djassemi [63]	153	41	5	41

Table 14. Summary of the model input dataset.

Parameters	Description	Values
α_{wt}	Task skill coefficient	[0.00, 0.77,0.83,0.91,1.00,1.11,1.25,1.43]
r_{uv}	Sociometric value of a worker pair	Discrete Uniform [1,5]
ST_p	Standard part processing time	Discrete Uniform [3,5]· 10
D_p	Part production demand in batch unit	Discrete Uniform [0,5]
	Batch size of part production demand	10
A	Worker available working hours	7
ϵ	The epsilon value	0.50
j	The penalty coefficient value	1,000,000
o	Maximum sociometric score	5.00
n	Minimum sociometric score	1.00
CTL	Cross-training level	[0.60, 0.80,1.00]
L	Requirement of cell-cohesion score	[0.00, 0.30, 0.60]

Table 15. Parameter settings for the proposed solution methodologies.

Methodologies	Parameters	Description	Values	
			Stage1	**Stage2**
NSGA-II	POP	Number of populations	100	-
	$maxGen$	Number of generations	1000	-
	kpx	k point crossover	$0.25 \cdot P$	-
	mut	Mutation rate	$0.05 \cdot W \cdot P$	-
Two-stage ALNS	des	Degree of destruction	[0.30,0.50,0.60]	[0.30,0.50,0.60]
	$i_{checkimp}$	Number of iterations without improvement	1000	10
	ω_0	Initial score	10	10
	ω_1	New global solution score	10	10
	ω_2	Improved solution score	5	5
	ω_3	New solution score	1	1
	$temp_0$	Initial annealing temperature	10,000	100
	γ	Cooling rate	0.99	0.99
	i_{max}	ALNS max iterations	10,000	100
	B	Number of total combinations	10	10

6.2. Computational Results

6.2.1. Solution Quality of the Proposed Method

A detailed comparison of the results from the two-stage ALNS and HBBFS methods is shown in Table 16. All the instances can be regarded as small instances with a maximum of seven workers in the team formation stage. We generated five replications of each cell condition concerning the random sociometric data, resulting in 140 test instances. The solution obtained using the two-stage ALNS method is equivalent to the exact solution obtained using the HBBFS method for all 140 test instances. Based on Figure 3, the two-stage ALNS method requires more CPU time than the HBBFS method for problems P1–P5. However, the two-stage ALNS method tends to be very effective on problems P6 and P7, providing the same results as the HBBFS method, with less computational effort.

6.2.2. Effectiveness of the Two-Stage Decision Methodology

We further validated the effectiveness of the two-stage ALNS method using the NSGA-II-based single-stage methodology. We examined the total inventory level and average cell worker's idle time variation of the two methods. Table 17 shows the results of the 40 test instances of all problem benchmarks. The results confirm the two-stage decision methodology to be more effective than the single-stage methodology, with the two-stage ALNS providing a feasible solution for all 40 test instances. However, a feasible solution could not be found using the NSGA-II method. All solutions obtained using the NSGA-II method were penalised by penalty functions. Another significant result from Figure 4 shows that the NSGA-II method consumed more CPU time than the two-stage ALNS method in all problem benchmarks—the two-stage ALNS found a feasible solution within 20 min for even the largest problem.

6.2.3. Managerial Insight Regarding Team Formation

We investigated the impact of the cell operational requirements on part-skill scores, with results being obtained from the two-stage ALNS method for solving problems P8–P10. Each problem benchmark included the three-level factorial design. Each problem benchmark included the three-level factorial design. We performed five replications for each cell condition concerning the random sociometric matrix input, yielding 135 instances. On the obtained results, we conducted a two-way analysis of variance (ANOVA) [64] to test for the interaction effects of the cross-training level and the level of cell-cohesion requirement. Table 18 shows the ANOVA results for investigating part-skill scores during team formation.

Finally, Table 19 summarises multiple comparisons of the part-skill score for each level of the cell-cohesion requirement.

Table 16. Comparison of two-stage ALNS and HBBFS methods on 140 small-sized instances.

				Part-Skill Score		Number of Solutions When Compared to HBBFS		
Instance	**Problem ID**	***CTL***	***L***	**HBBFS**	**Two-Stage ALNS**	**Better**	**Worse**	**Equal**
1–5	P1	0.80	0.60	3.27	3.27	0	0	5
6–10		0.60	0.60	2.55	2.55	0	0	5
11–15		0.80	0.30	3.23	3.23	0	0	5
16–20		0.60	0.30	2.91	2.91	0	0	5
21–25	P2	0.80	0.60	4.68	4.68	0	0	5
26–30		0.60	0.60	3.30	3.30	0	0	5
31–35		0.80	0.30	5.21	5.21	0	0	5
36–40		0.60	0.30	4.26	4.26	0	0	5
41–45	P3	0.80	0.60	6.46	6.46	0	0	5
46–50		0.60	0.60	5.27	5.27	0	0	5
51–55		0.80	0.30	6.76	6.76	0	0	5
56–60		0.60	0.30	6.29	6.29	0	0	5
61–65	P4	0.80	0.60	7.15	7.15	0	0	5
66–70		0.60	0.60	4.86	4.86	0	0	5
71–75		0.80	0.30	7.81	7.81	0	0	5
76–80		0.60	0.30	5.84	5.84	0	0	5
81–85	P5	0.80	0.60	9.16	9.16	0	0	5
86–90		0.60	0.60	7.18	7.18	0	0	5
91–95		0.80	0.30	9.42	9.42	0	0	5
96–100		0.60	0.30	8.22	8.22	0	0	5
101–105	P6	0.80	0.60	9.83	9.83	0	0	5
106–110		0.60	0.60	8.41	8.41	0	0	5
111–115		0.80	0.30	9.88	9.88	0	0	5
116–120		0.60	0.30	9.22	9.22	0	0	5
121–125	P7	0.80	0.60	11.15	11.15	0	0	5
126–130		0.60	0.60	10.01	10.01	0	0	5
131–135		0.80	0.30	11.44	11.44	0	0	5
136–140		0.60	0.30	11.11	11.11	0	0	5

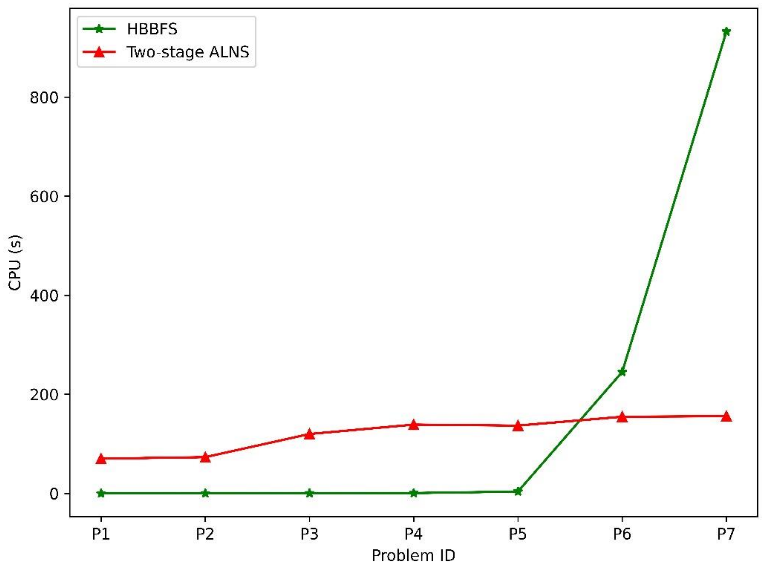

Figure 3. Comparison of the CPU time between the two-stage ALNS and HBBFS methods during the team formation stage.

Table 17. Results of instances obtained using the NSGA-II and two-stage ALNS methods.

				Total Inventory Level		Average Workers Idle Time Variation	
Instance	Problem ID	*CTL*	*L*	NSGA-II	Two-Stage ALNS	NSGA-II	Two-Stage ALNS
141	P1	0.80	0.60	*	315.00	*	0.18
142		0.60	0.60	*	263.00	*	0.18
143		0.80	0.30	*	291.00	*	0.09
144		0.60	0.30	*	325.00	*	0.09
145	P2	0.80	0.60	*	757.00	*	0.24
146		0.60	0.60	*	623.00	*	0.30
147		0.80	0.30	*	686.00	*	0.32
148		0.60	0.30	*	588.00	*	0.32
149	P3	0.80	0.60	*	774.00	*	0.13
150		0.60	0.60	*	801.00	*	0.18
151		0.80	0.30	*	717.00	*	0.18
152		0.60	0.30	*	760.00	*	0.25
153	P4	0.80	0.60	*	1463.00	*	0.28
154		0.60	0.60	*	1697.00	*	0.31
155		0.80	0.30	*	1402.00	*	0.24
156		0.60	0.30	*	1550.00	*	0.19
157	P5	0.80	0.60	*	2107.00	*	0.16
158		0.60	0.60	*	2390.00	*	0.30
159		0.80	0.30	*	2165.00	*	0.30
160		0.60	0.30	*	2279.00	*	0.23
161	P6	0.80	0.60	*	1315.00	*	0.30
162		0.60	0.60	*	1360.00	*	0.31
163		0.80	0.30	*	1369.00	*	0.27
164		0.60	0.30	*	1253.00	*	0.31
165	P7	0.80	0.60	*	3237.00	*	0.29
166		0.60	0.60	*	2545.00	*	0.34
167		0.80	0.30	*	2835.00	*	0.37
168		0.60	0.30	*	3021.00	*	0.34
169	P8	0.80	0.60	*	3448.00	*	0.29
170		0.60	0.60	*	3680.00	*	0.32
171		0.80	0.30	*	3533.00	*	0.32
172		0.60	0.30	*	3749.00	*	0.35
173	P9	0.80	0.60	*	5380.00	*	0.40
174		0.60	0.60	*	5958.00	*	0.31
175		0.80	0.30	*	5573.00	*	0.42
176		0.60	0.30	*	6004.00	*	0.34
177	P10	0.80	0.60	*	6319.00	*	0.40
178		0.60	0.60	*	6584.00	*	0.37
179		0.80	0.30	*	6478.00	*	0.34
180		0.60	0.30	*	6643.00	*	0.46

* Infeasible solution.

To visualise the impact of cell-cohesion requirement when forming a team of cross-trained workers in a cell, Figure 5 shows the interaction effects between the level of cross-training and cell-cohesion requirement on the part-skill score. The interaction effects are significant in all three problem benchmarks. However, the multiple comparison results in Table 19 show that the difference in part-skill scores between no cell-cohesion requirement and low cell-cohesion requirement is not statistically significant. The results suggest that there is no impact of low cell-cohesion requirement on the decision of team formation in terms of workers' capabilities. In contrast, the decision of team formation becomes more critical when requiring 0.6 of the normalised cell-cohesion score, which could diminish the total part-skill score of partially cross-trained workers in the cell. This is intuitive since the team member selection is quite limited in the condition when a high cell-cohesion is required. As a result, there is a high possibility that workers in the high cohesion cell might not be able to satisfy part production demands. Thus, the decision maker should be aware of the high cell-cohesion requirement when forming a partially cross-trained team in the cell.

Table 18. Two-way ANOVA results on part-skill score.

	Problem ID: P8		Problem ID: P9		Problem ID: P10	
	Part-Skill Score		Part-Skill Score		Part-Skill Score	
Source	F-Value	p-Value	F-Value	p-Value	F-Value	p-Value
CTL	631.30	0.00 *	1588.19	0.00 *	4190.05	0.00 *
L	43.73	0.00 *	143.14	0.00 *	206.46	0.00 *
$CTL \times L$	12.14	0.00 *	59.51	0.00 *	54.59	0.00 *

* Significant at the 5% level.

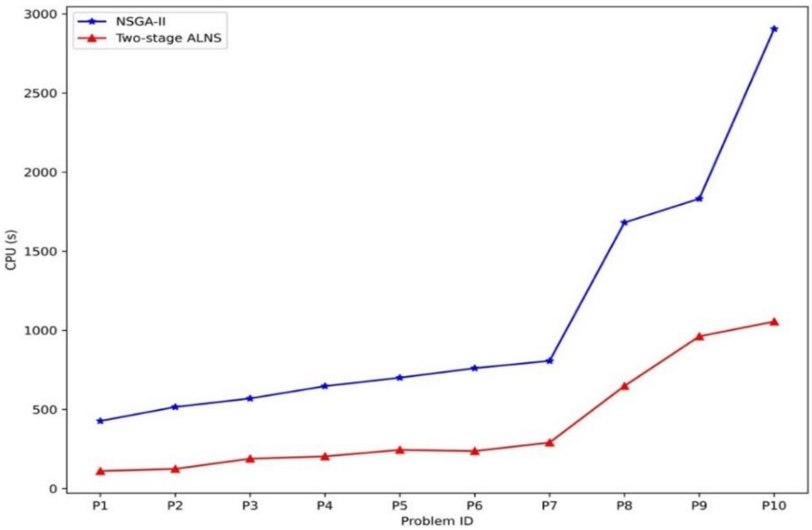

Figure 4. The two-stage ALNS method outperforms the NSGA-II method in terms of computational effort.

Table 19. Multiple comparison of part-skill score in each cell-cohesion requirement level.

		Problem ID: P8		Problem ID: P9		Problem ID: P10	
		Part-Skill Score		Part-Skill Score		Part-Skill Score	
Cell-Cohesion Requirement (L)		Mean Difference	p-Value	Mean Difference	p-Value	Mean Difference	p-Value
0.00	0.30	−0.30	0.99	0.75	0.71	−0.12	0.62
	0.60	2.08	0.00 *	1.43	0.00 *	2.26	0.00 *
0.30	0.00	0.30	0.99	−0.75	0.71	0.12	0.62
	0.60	2.11	0.00 *	−1.35	0.00 *	2.39	0.00 *
0.60	0.00	−2.08	0.00 *	−1.43	0.00 *	−2.26	0.00 *
	0.30	−2.11	0.00 *	−1.35	0.00 *	−2.39	0.00 *

* Significant at the 5% level.

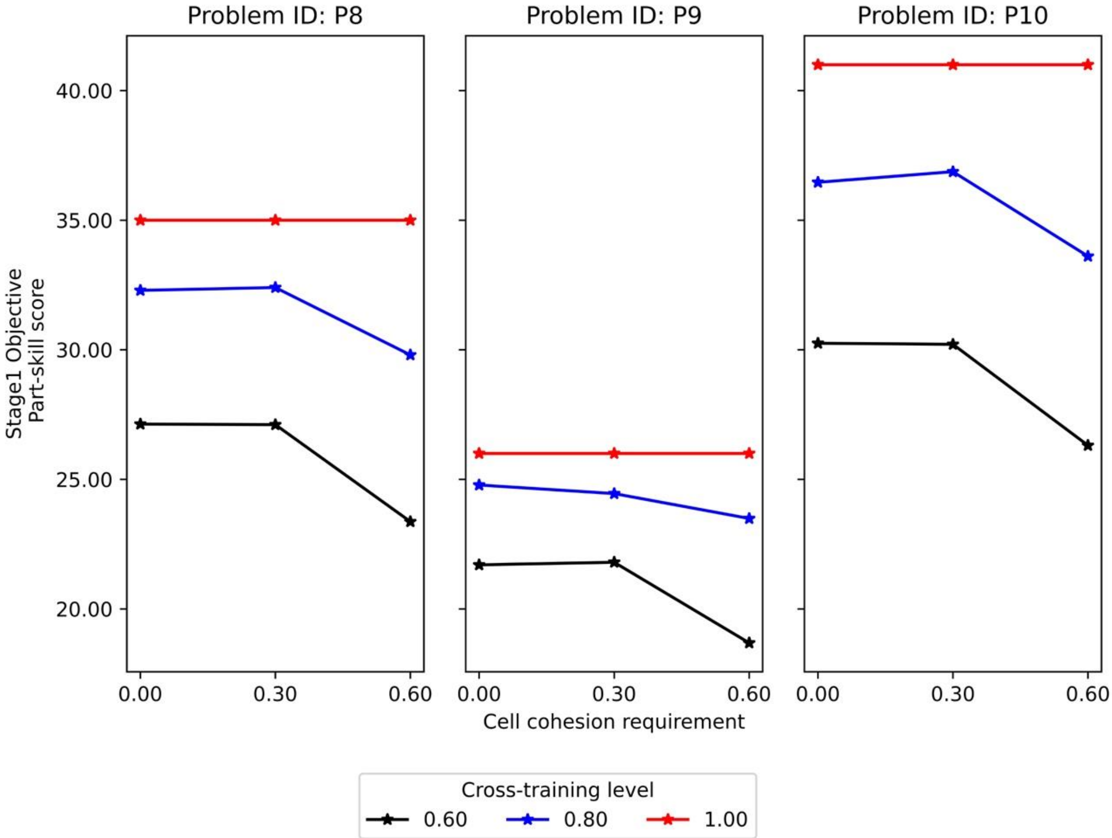

Figure 5. Interaction effects between cell-cohesion requirement and cross-training level on part-skill score.

7. Concluding Remarks

This study addressed the team formation and worker assignment problem through a mathematical programming model that considered interpersonal relationships among cross-trained workers in a CMS. A two-stage decision-based methodology was developed based on the ALNS framework—namely, two-stage ALNS. The solution quality of the two-stage ALNS method was tested on 140 single-stage small-sized test instances using the developed HBBFS algorithm. Of the 140 small-sized test instances, the solutions using the proposed method were equivalent to the exact solution at the team formation stage. Furthermore, the two-stage ALNS method was compared to the NSGA-II-based single-stage decision methodology with 40 test instances for all problem benchmarks. The results revealed that two-stage ALNS method outperformed the NSGA-II method, providing feasible solutions with less computational effort for all test instances. Finally, based on the results of the two-stage ALNS method, we investigated how the cross-training level and cell-cohesion requirement affected the decision when forming a team of workers in the cell. The results suggest that a higher level of cell-cohesion requirement could reduce the total number of workers' skills for partially cross-trained workers.

The scope of this study can be split into three possible directions. First, the problem can be extended to a multiperiod problem. For instance, future work could introduce the dynamism of the relationships among workers, which could change over time. Other metaheuristic-based solving methods could be developed to improve the solution quality and reduce the computational time of the multiperiod problem. Second, it is worthwhile to investigate the impact of team-based requirements on the performance of cell team formation using the KCI and MBTI. Third, the problem dimension of the TFWAP can be extended in several ways. In reality, workers would prefer to operate their familiar tasks. Managers could prioritise each part's production differently according to the customer's demands. These issues are essential for a practical TFWAP. We expect that the developed model could be adapted and applied in the context of team project management—as it shares some common characteristics with the TFWAP in a CMS.

Supplementary Materials: The following supporting information can be downloaded at: https://www.mdpi.com/article/10.3390/app12168323/s1, Online supplemental appendix.

Author Contributions: Conceptualization, T.P. and S.S.; methodology, T.P. and S.S.; software, T.P.; validation, T.P.; writing—original draft preparation, T.P.; writing—review and editing, T.P.; supervision, S.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The following data can be downloaded at: https://github.com/Carlzeriss/TFWAP (accessed on 8 Febuary 2022).

Acknowledgments: The authors would like to thank Paniti Achararit and Pavinee Rerkjirattikal for their insightful suggestions and comments on improving this manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Bidanda, B.; Ariyawongrat, P.; Needy, K.L.; Norman, B.A.; Tharmmaphornphilas, W. Human related issues in manufacturing cell design, implementation, and operation: A review and survey. *Comput. Ind. Eng.* **2005**, *48*, 507–523. [CrossRef]
2. Cesaní, V.I.; Steudel, H.J. A study of labor assignment flexibility in cellular manufacturing systems. *Comput. Ind. Eng.* **2005**, *48*, 571–591. [CrossRef]
3. Slomp, J.; Molleman, E. Cross-training policies and team performance. *Int. J. Prod. Res.* **2005**, *40*, 1193–1219. [CrossRef]
4. Sengupta, K.; Jacobs, F.R. Impact of work teams: A comparison study of assembly cells and assembly line for a variety of operating environments. *Int. J. Prod. Res.* **2004**, *42*, 4173–4193. [CrossRef]
5. Sakazume, Y. Is japanese cell manufacturing a new system?: A comparative study between japanese cell manufacturing and cellular manufacturing. *J. Jpn. Ind. Manag. Assoc.* **2005**, *55*, 341–349.
6. Min, H.; Shin, D. Simultaneous formation of machine and human cells in group technology: A multiple objective approach. *Int. J. Prod. Res.* **1993**, *31*, 2307–2318. [CrossRef]
7. Chen, S.; Lin, L. Modeling team member characteristics for the formation of a multifunctional team in concurrent engineering. *IEEE Trans. Eng. Manag.* **2004**, *51*, 111–124. [CrossRef]
8. Gutiérrez, J.H.; Astudillo, C.A.; Ballesteros-Pérez, P.; Mora-Melià, D.; Candia-Véjar, A. The multiple team formation problem using sociometry. *Comput. Oper. Res.* **2016**, *75*, 150–162. [CrossRef]
9. Ballesteros-Pérez, P.; González-Cruz, M.C.; Fernández-Diego, M. Human resource allocation management in multiple projects using sociometric techniques. *Int. J. Proj. Manag.* **2012**, *30*, 901–913. [CrossRef]
10. Chen, R. Grouping optimization based on social relationships. *Math. Prob. Eng.* **2012**, *2012*, 170563. [CrossRef]
11. Campêlo, M.; Figueiredo, T.; Silva, A. The sociotechnical teams formation problem: A mathematical optimization approach. *Ann. Oper. Res.* **2020**, *286*, 201–216. [CrossRef]
12. Zakarian, A.; Kusiak, A. Forming teams: An analytical approach. *IIE Trans.* **1999**, *31*, 85–97. [CrossRef]
13. Zhang, L.; Zhang, X. Multi-objective team formation optimization for new product development. *Comput. Ind. Eng.* **2013**, *64*, 804–811. [CrossRef]
14. Fathian, M.; Saei-Shahi, M.; Makui, A. A new optimization model for reliable team formation problem considering experts' collaboration network. *IIE Trans. Eng. Manag.* **2017**, *64*, 586–593. [CrossRef]
15. Stevens, M.J.; Campion, M.A. The knowledge, skill, and ability requirements for teamwork: Implications for human resource management. *J. Manag.* **1994**, *20*, 503–530. [CrossRef]
16. Salas, E.; Shuffler, M.L.; Thayer, L.A.; Wendy, B.L.; Elizabeth, L.H. Understanding and improving teamwork in organizations: A scientific based practical guide. *Hum. Res. Manag.* **2014**, *54*, 599–622. [CrossRef]
17. Hoegl, M.; Gemuenden, H.G. Teamwork quality and the success of innovative projects: A theoretical concept and empirical evidence. *Org. Sci.* **2001**, *12*, 435–449. [CrossRef]
18. Beal, D.J.; Cohen, R.R.; Burke, M.J.; McLendon, C.L. Cohesion and performance in groups: A meta-analytic clarification of construct relations. *J. Appl. Psychol.* **2003**, *88*, 989–1004. [CrossRef]
19. Mullen, B.; Copper, C. The relation between group cohesiveness and performance: An integration. *Psychol. Bull.* **1994**, *115*, 210–227. [CrossRef]
20. Moreno, J.L. Foundations of sociometry, an introduction. *Sociometry* **1941**, *4*, 15–35. [CrossRef]

21. Rahmanniyay, F.; Yu, A.J.; Seif, J. A multi-objective multi-stage stochastic model for project team formation under uncertainty in time requirements. *Comput. Ind. Eng.* **2019**, *132*, 153–165. [CrossRef]
22. Feng, B.; Jiang, Z.; Fan, Z.; Fu, N. A method for member selection of cross-functional teams using the individual and collaborative performances. *Eur. J. Oper. Res.* **2010**, *203*, 652–661. [CrossRef]
23. Askin, R.G.; Huang, Y. Forming effective worker teams for cellular manufacturing. *Int. J. Prod. Res.* **2001**, *39*, 2431–2451. [CrossRef]
24. Fitzpatrick, E.L.; Askin, R.G. Forming effective worker teams with multi-functional skill requirements. *Comput. Ind. Eng.* **2005**, *48*, 593–608. [CrossRef]
25. Yilmaz, O.F.; Ozcelik, G.; Yeni, F.B. Lean holistic fuzzy methodology employing cross-functional worker teams for new product development projects: A real case study from high-tech industry. *Eur. J. Oper. Res.* **2020**, *282*, 989–1010. [CrossRef]
26. Norman, B.A.; Tharmmaphornphilas, W.; Needy, K.L.; Bidanda, B.; Warner, R.C. Worker assignment in cellular manufacturing considering technical and human skills. *Int. J. Prod. Res.* **2002**, *40*, 1479–1492. [CrossRef]
27. Wirojanagud, P.; Gel, E.S.; Fowler, J.W.; Cardy, R. Modelling inherent worker differences for workforce planning. *Int. J. Prod. Res.* **2007**, *45*, 525–553. [CrossRef]
28. Süer, G.A.; Tummaluri, R.R. Multi-period operator assignment considering skills, learning and forgetting in labour-intensive cells. *Int. J. Prod. Res.* **2008**, *46*, 469–493. [CrossRef]
29. McDonald, T.; Ellis, K.P.; Van Aken, E.M.; Patrick, K.C. Development and application of a worker assignment model to evaluate a lean manufacturing cell. *Int. J. Prod. Res.* **2009**, *47*, 2427–2447. [CrossRef]
30. Aryanezhad, M.B.; Deljoo, V.; Mirzapour Al-e-hashem, S.M.J. Dynamic cell formation and the worker assignment problem: A new model. *Int. J. Adv. Manuf. Technol.* **2009**, *41*, 329–342. [CrossRef]
31. Yilmaz, O.F.; Durmusoglu, M.B. A performance comparison and evaluation of metaheuristics for a batch scheduling problem in a multi-hybrid cell manufacturing system with skilled workforce assignment. *J. Ind. Manag. Opt.* **2018**, *14*, 1219–1249. [CrossRef]
32. Liu, C.; Wang, J.; Leung, J.Y.T. Worker assignment and production planning with learning and Forgetting in Manufacturing Cells by Hybrid Bacteria Foraging Algorithm. *Comput. Ind. Eng.* **2016**, *96*, 162–179. [CrossRef]
33. Jin, H.; Hewitt, M.; Thomas, B.W. Workforce grouping and assignment with learning-by-doing and knowledge transfers. *Int. J. Prod. Res.* **2018**, *5*, 4968–4982. [CrossRef]
34. Chu, X.; Gao, D.; Cheng, S.; Wu, L.; Chen, J.; Shi, Y.; Qin, Q. Worker assignment with learning-forgetting effect in cellular manufacturing system using adaptive memetic differential search algorithm. *Comput. Ind. Eng.* **2019**, *136*, 381–396. [CrossRef]
35. Méndez-Vázquez, Y.M.; Nembhard, D.A. Worker-cell assignment: The impact of organizational factors on performance in cellular manufacturing systems. *Comput. Ind. Eng.* **2019**, *127*, 1101–1111. [CrossRef]
36. Liu, C.; Yang, N.; Li, W.; Lian, J.; Evans, S.; Yin, Y. Training and assignment of multi-skilled workers for implementing seru production systems. *Int. J. Adv. Manuf. Technol.* **2013**, *69*, 937–959. [CrossRef]
37. Ying, K.; Tsai, Y. Minimising total cost for training and assigning multiskilled workers in seru production systems. *Int. J. Prod. Res.* **2017**, *55*, 2978–2989. [CrossRef]
38. Yu, Y.; Wang, J.; Ma, K.; Sun, W. Seru system balancing: Definition, formulation, and exact solution. *Comput. Ind. Eng.* **2018**, *122*, 318–325. [CrossRef]
39. Lian, J.; Liu, C.; Li, W.J.; Yin, Y. A multi-skilled worker assignment problem in seru production systems considering the worker heterogeneity. *Comput. Ind. Eng.* **2018**, *118*, 366–382. [CrossRef]
40. Wu, L.; Cai, F.; Li, L.; Chu, X. Cross-trained worker assignment problem in cellular manufacturing system using swarm intelligence metaheuristics. *Math. Prob. Eng.* **2018**, *2018*, 4302062. [CrossRef]
41. Niakan, F.; Baboli, A.; Moyaux, T.; Botta-Genoulaz, V. A bi-objective model in sustainable dynamic cell formation problem with skill-based worker assignment. *Comput. Ind. Eng.* **2016**, *38*, 46–62. [CrossRef]
42. Kuo, Y.; Liu, C. Operator assignment in a labor-intensive manufacturing cell considering inter-cell manpower transfer. *Comput. Ind. Eng.* **2017**, *110*, 83–91. [CrossRef]
43. Feng, H.; Da, W.; Xi, L.; Pan, E.; Xia, T. Solving the integrated cell formation and worker assignment problem using particle swarm optimization and linear programming. *Comput. Ind. Eng.* **2017**, *110*, 126–137. [CrossRef]
44. Campbell, G.M. Cross-utilization of workers whose capabilities differ. *Manag. Sci.* **1999**, *45*, 722–7323. [CrossRef]
45. Pisinger, D.; Ropke, S. A general heuristic for vehicle routing problems. *Comput. Oper. Res.* **2007**, *34*, 2403–2435. [CrossRef]
46. Ropke, S.; Pisinger, D. An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transp. Sci.* **2006**, *40*, 455–472. [CrossRef]
47. Lahyani, R.; Gouguenheim, A.L.; Coelho, L.C. A hybrid adaptive large neighbourhood search for multi-depot open vehicle routing problems. *Int. J. Prod. Res.* **2019**, *57*, 6963–6976. [CrossRef]
48. Hellsten, E.O.; Sacramento, D.; Pisinger, D. An adaptive large neighbourhood search heuristic for routing and scheduling feeder vessels in multi-terminal ports. *Eur. J. Oper. Res.* **2020**, *287*, 682–698. [CrossRef]
49. Chen, C.; Demir, E.; Huang, Y. An adaptive large neighborhood search heuristic for the vehicle routing problem with time windows and delivery robots. *Eur. J. Oper. Res.* **2021**, *294*, 1164–1180. [CrossRef]
50. Friedrich, C.; Elbert, R. Adaptive large neighborhood search for vehicle routing problems with transshipment facilities arising in city logistics. *Comput. Oper. Res.* **2022**, *137*, 105491. [CrossRef]
51. Belo-Filho, M.A.F.; Amorim, P.; Almada-Lobo, B. An adaptive large neighbourhood search for the operational integrated production and distribution problem of perishable products. *Int. J. Prod. Res.* **2015**, *53*, 6040–6058. [CrossRef]

52. He, K.; Tole, K.; Ni, F.; Yuan, Y.; Liao, L. Adaptive large neighborhood search for solving the circle bin packing problem. *Comput. Oper. Res.* **2021**, *127*, 105140. [CrossRef]
53. Avci, M.G.; Avci, M. An adaptive large neighborhood search approach for multiple traveling repairman problem with profits. *Comput. Oper. Res.* **2019**, *111*, 367–385. [CrossRef]
54. Jiang, D.; Li, X. Order fulfilment problem with time windows and synchronisation arising in the online retailing. *Int. J. Prod. Res.* **2021**, *59*, 1187–1215. [CrossRef]
55. Jarumaneeroj, P.; Sakulsom, N. An adaptive large neighborhood search for the multiple-day music rehearsal problems. *Comput. Ind. Eng.* **2021**, *157*, 107279. [CrossRef]
56. He, L.; Liu, X.; Laporte, G.; Chen, Y.; Chen, Y. An improved adaptive large neighborhood search algorithm for multiple agile satellites scheduling. *Comput. Oper. Res.* **2018**, *100*, 12–25. [CrossRef]
57. Shaw, P. Using constraint programming and local search methods to solve vehicle routing problems. In *Principles and Practice of Constraint Programming—CP98*; Maher, M., Puget, J.F., Eds.; Springer: Berlin/Heidelberg, Germany, 1998; Volume 1520, pp. 417–431.
58. Won, Y.; Kim, S. Multiple criteria clustering algorithm for solving the group technology problem with multiple process routings. *Comput. Ind. Eng.* **1997**, *32*, 207–220. [CrossRef]
59. Zolfaghari, S.; Liang, M. Comparative study of simulated annealing, genetic algorithms and tabu search for solving binary and comprehensive machine-grouping Problems. *Int. J. Prod. Res.* **2002**, *40*, 2141–2158. [CrossRef]
60. Islam, K.M.S.; Sarker, B.R. A similarity coefficient measure and machine-parts grouping in cellular manufacturing systems. *Int. J. Prod. Res.* **2000**, *38*, 699–720. [CrossRef]
61. Yang, M.; Yang, J. Machine-part cell formation in group technology using a modified ART1 method. *Eur. J. Oper. Res.* **2008**, *188*, 140–152. [CrossRef]
62. Moon, Y.B.; Chi, S.C. Generalized part family formation using neural network techniques. *J. Manuf. Syst.* **1992**, *11*, 149–159. [CrossRef]
63. Seifoddini, H.; Djassemi, M. The threshold value of a quality index for formation of cellular manufacturing systems. *Int. J. Prod. Res.* **1996**, *34*, 3401–3416. [CrossRef]
64. Montgomery, D.C. *Design and Analysis of Experiments*, 3rd ed.; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 2013; ISBN 9781118146927.