A water-flow like algorithm for solving U-shaped assembly line balancing problems

Amir Nourmohammadi1*, Masood Fathi1, Mostafa Zandieh2, Morteza Ghabakhloo3,4
1Department of Production & Automation Engineering, University of Skövde, P.O. Box 408, SE-541 28, Skövde, Sweden
2Department of Industrial Management, Management and Accounting Faculty, SBU, G.C., Tehran, Iran
3Department of Industrial Engineering, Minab Higher Education Center, University of Hormozgan, Bandar Abbas, Iran
4Modern Technology Development and Implementation Research Center, University of Hormozgan, Bandar Abbas, Iran
*Corresponding author: Amir Nourmohammadi (e-mail: nourmohammadi.amir@gmail.com; amir.nourmohammadi@his.se).

ABSTRACT The problem of assigning assembly tasks to the stations arranged along a material handling device is known as assembly line balancing. This paper aims to address the U-shaped assembly line balancing problem (UALBP) which arises when a U-shaped assembly line has to be configured. It is widely known that this problem is NP-hard. Accordingly, different meta-heuristics based on a single solution (such as Simulated Annealing) or a population of solutions (such as Genetic Algorithms) have been proposed in the literature. Meanwhile, it has been argued that either of these meta-heuristics with a fixed number of solutions cannot maintain efficient search progress and thus can lead to premature convergence. Thus, this study aims at adopting a novel meta-heuristic algorithm with dynamic population sizes, namely Water Flow-like Algorithm (WFA), inspired by the behaviour of water flows in nature, to address the UALBP. The line efficiency and variation of workload are considered as the primary and the secondary objective, to be optimized, respectively. To verify the efficiency and robustness of the proposed WFA, a real case study taken from an automobile manufacturer as well as a set of standard problems are solved and the results compared with the existing approaches in the literature. The computational results demonstrate the superiority of the WFA, particularly in addressing medium to large-sized problems.

INDEX TERMS U-shaped, assembly line balancing, water-flow-like algorithm.

I. INTRODUCTION Manufacturing industry is nowadays witnessing the fourth industrial revolution that is commonly referred to as Industry 4.0. Since the mass production of individualized goods is one the key design principles of Industry 4.0, maximizing the assembly line (AL) performance is a primary objective for contemporary manufacturers. The AL is a well-known manufacturing process frequently used in different industries such as automobile and electronics manufacture, in which one/more product(s) is/are assembled through a sequence of stations arranged along a material handling device or system. The problem of assigning assembly tasks to the stations when one or more objectives are to be optimized considering some constraints is called the assembly line balancing problem (ALBP) [1]. The ALBP basically divides into the simple assembly line balancing (SALBP) and the generalized assembly line balancing problem (GALBP) [2].
The GALBP deals with more practical considerations arising in the real-world ALs such as U-shaped [3], two-sided [4] and stochastic task times [5], [6], which are not considered in the SALBP.

Within the Industry 4.0 era, applying lean manufacturing principles such as just-in-time (JIT) to assure a higher productivity and a lower production cost is a strategic necessity. U-shaped ALs can be considered as a consequence of Lean Manufacturing implementation, as they can facilitate the assignment of tasks to the stations (operators and / or machines) by providing more flexibility in dealing with tasks on both sides of the ALs. In addition, by allowing the interaction between tasks/stations in U-shaped lines, the line efficiency, workload equalization between stations (operators), space utilization, operators’ communication, job-enrichment, and work-in-process (WIP) reduction can also be further enhanced [7]. Considering these advantages, many ALs are being arranged in a U-shape rather than straight lines [8]. However, there are only a small number of studies reported in the literature dealing with the U-shaped specifically [9].

The problem of partitioning the tasks among the stations so that a single-model of a product can be assembled on a U-shaped AL, is called the U-shaped ALBP (UALBP). There are mainly two types of UALBPs, namely: (1) UALBP-1, and (2) UALBP-2 [7]. The first one, attempts to minimize the number of workstations (m), given a cycle time (CT) which is the maximum available processing time at each station. The second type, targets minimizing CT for the pre-determined m. The majority of studies in the UALBP literature have focused on the UALBP-1, as it has been found the most important decision problem arise when U-shaped ALs have to be designed [10]. Other objectives have also been considered while dealing with the main objective such as variations of workload [11], [12] and Smoothness Index [13].

To address the UALBPs, different solution methods have been suggested in the literature, including exacts [14], [15] and heuristics [16], [17]. Moreover, since any UALBP falls within the NP-hard problems when the problem sizes grow [18], different meta-heuristics have been suggested to address large-scale problems. The most widely applied algorithms in the UALBP are Genetic Algorithms [11], [19], Ant Colony Optimization [20], Simulated Annealing (SA) [21], Particle Swarm Optimization (PSO) [4], and Grouping Evolution Strategy [12]. For more information and recent literature reviews about the application of soft computing approaches for UALBP, refer to [7], [22], [23].

All of the above algorithms, which are either single-solution or population-based, maintain a fixed population over their search process. However, one of the most important features of meta-heuristics affecting their search efficiency and effectiveness while avoiding local traps and premature convergence is having a dynamic population size [24]. In this regard, the water flow-like algorithm (WFA), inspired by the behavior of water flows traversing from higher to lower altitudes, was introduced by Yang and Wang [24]. WFA is designed by imitating the water flow behaviours/operations, i.e., splitting and moving, merging, evaporation and precipitation, as a search mechanism while maintaining a dynamic population size. WFA was first applied as an Evolutionary Algorithm (EA) to address the bin packing problems and the results showed that it was superior to GA, PSO and Electromagnetism algorithms [24]. More recently WFA has been applied to other optimization problems, namely cell formation [25], logistics [26], travelling salesman problems [27], and scheduling [28]–[30] out performing other EAs including GA [24]–[26] and SA [25] in its limited applications.

Following the first successful application of WFA in addressing the bin packing problem which can be regarded as a reduced form of SALBP by eliminating the precedence relationships [17] and considering its superiority over other meta-heuristics in its limited applications, this paper aims at adopting the WFA to address the UALBP-1. To the authors’ knowledge, this is the first application of WFA in the ALBP literature. The objectives considered for addressing the UALBP are: primary, maximizing the line efficiency (which is equivalent to minimizing m when CT is fixed) and secondly, minimizing the variation of workload so that the workloads are equally distributed amongst the stations. To illustrate the applicability of the proposed WFA to solve real-world problems, an ALBP taken from an automobile manufacturing company is presented. Moreover, to further validate the performance of the proposed approach, a set of standard test problems are solved and the results are compared with the existing approaches in the literature. The results show the outstanding efficiency of WFA when compared to the existing methods, particularly in addressing medium to large-scale problems.

The remainder of the paper is organised as follows. In Section II, the definition and formulation of the problem and the application study addressed are summarised. Section III concentrates on the development of the proposed WFA to solve the UALBP under consideration. In Section IV, the computational study is performed including the parameter tuning and the computational results of WFA when compared to other algorithms. Finally, conclusions and future research directions are suggested in Section V.

II. PROBLEM DESCRIPTION AND CASE STUDY

In this section, the UALBP-1 problem considered is described in detail. Moreover, the case study taken from the engine assembly line is explained.

A. PROBLEM DESCRIPTION

The UALBP-1 considered aimed at partitioning a set of assembly tasks, represented by \( j = 1, 2, \ldots, n \), amongst a number of stations, indicated by \( k = 1, 2, \ldots, m \), that are arranged along a U-shaped line where stations are linked with a material handling system. The primary optimisation criterion for UALBP-1 is to minimize the number of stations...
(m) while the the variation of workload (VW) is considered as the secondary optimisation criterion. Each task \( j \) that has a processing time of \( t_j \) and can be assigned to a station if its immediate predecessor and successors have been performed in the previous/current station(s). The tasks relationships is usually determined by a precedence network as shown in FIGURE 1. In this figure, each task is represented by a node and the number above it represents the task time. The arrows between nodes define the precedence relationships (Pr).

![FIGURE 1. Illustration of a precedence network for an ALBP.](image)

The \( S_k \) is a set that contains all the tasks assigned to station \( k \) and the station time is calculated as \( t(S_k) = \sum_{j \in S_k} t_j \). There is a maxmimum time available for each station known as \( CT \). Given \( CT \), a line balancing solution is valid if and only if: (1) the sum of task times assigned to each station is less than or equal to \( CT \), and (2) the precedence relationships amongst tasks are not violated. For example, a feasible line balancing solution of the network presented in FIGURE 1 is depicted in FIGURE 2. Given a \( CT \) (i.e., 10), this figure shows a UALBP solution with five stations and stations workload of \( S_1=[1,5,2] \), \( S_2=[4,6] \), \( S_3=[3,11] \), \( S_4=[10,9] \) and \( S_5=[7,8] \).

![FIGURE 2. Feasible task assignments for a UALBP with CT = 10.](image)

In order to better understand the objectives and constraints of the UALBP-1, a mathematical model of the problem is presented in equations (1) to (6). The decision variables used in the model are defined in Nomenclature.

The UALBP-1 considered can be modeled as follows:

\[
\text{Objectives} \quad \begin{align*}
\min LE &= \frac{\sum_{i=1}^{m} t_i}{m \times CT} \\
\min VW &= \frac{\sqrt{\sum_{k=1}^{m} (U_k - A)^2}}{m}
\end{align*}
\]

Equation (1) calculates the objective function, namely (1) \( LE \) and (2) \( VW \), where in \( VW \) \( A = \sum_{k=1}^{m} U_k / m \) is the average utilization of stations and \( U_k = t(S_k) / \max_{k=1}^{m} t(S_k) \) is the utilization of \( k \)-th station. The \( VW \) ranges between [0,1] in which a lower value indicates a line with smoother workloads between stations/workers. Equation (2) ensures that each task is assigned to only one station either in a forward or reverse direction. Equation (3) ensures that the sum of task times assigned to each station does not violate the given \( CT \). Equations (4) and (5) ensure that during the forward and backward assignments, respectively, the precedence relationships (Pr) among tasks are not violated. Finally, Equation (6) defines the decision variables domain which are binary.

### B. CASE STUDY EXPLANATION

The case studied is a section of an assembly line found in an automobile manufacturer. This problem includes 29 tasks, each with a known processing time in seconds. The data for the case study is summarised in Table I. This data has been extracted by experts in the company. Due to confidentiality requirements, the descriptions of tasks are not provided. The decision makers (DM) have attempted to identify the best U-shaped configuration of assembly line considering three scenarios of \( CT \) (40, 45 and 50 seconds respectively) so that line efficiency is maximized primarily and the variation of workload amongst stations is minimised as a secondary measure.

![TABLE I. THE DATA FOR THE CASE STUDY](image)
III. THE PROPOSED WATER-FLOW LIKE ALGORITHM

In general, WFA which is inspired by water flow behaviour in nature whilst passing different terrains, employs four main operations: (1) splitting and moving, (2) merging, (3) evaporation, and (4) precipitation. By considering water flows as the solutions and the terrains they travel through as the search space, the WFA has been introduced by Yang and Wang [24] as a search algorithm for optimization problems. The main feature of WFA that distinguishes it from other meta-heuristics is that, depending on the water flows and the landscape characteristics, the number of solutions/agents can dynamically change throughout the search mechanisms. Following its successful application in addressing the bin packing problem that can be regarded as a simple form of SALBP without precedence constraints with results that out perform other algorithms (e.g., GA, PSO, and SA), we apply WFA to address the UALBP-1. The following procedures have been identified in adapting the WFA to solve this problem.

A. INITIALIZATION OF SOLUTIONS AND PARAMETERS

At the initial step, it is assumed that only one flow exists (NF=1), with the initial mass \( W_0 \), velocity \( V_0 \), and randomly generated position. The position of each flow in the terrains has to be shown by a solution representation in WFA. Here, by assuming a vector with length \( n \) (number of tasks in UALBP), the solution representation includes a permutation vector of integers between \([1,n]\) as the task priorities \( (\rho_i) \). Thus, a random permutation vector of \([1,n]\) is generated as the initial solution of the WFA.

B. ENCODING, DECODING AND FITNESS FUNCTION EVALUATION

To ensure the feasibility of WFA solutions, there is a need for encoding and decoding processes. Moreover, to guide the water flow as the solution agents, towards the lowers terrains in the search space, there is a need for a fitness function evaluation. Considering the UALBP-1, the related processes are described as follows.

We assume a graph \( \theta = (\eta, \lambda) \) for the precedence network of a \( n \)-task UALBP, in which \( \eta \) is the set of nodes and \( \lambda \) is the set of edges representing the tasks and their precedence relationships, respectively. Given \( \theta = (\eta, \lambda) \), the encoding process attempts to order the tasks according to the precedence network to find a sequence of tasks known as the Task Sequence (TS) for each vector of task priorities \( \rho \). Additionally, a decoding process is required to map the TS to a feasible UALBP solution by assigning the tasks in the TS to the stations so that the time of each station does not exceed the given \( CT \) and the precedence relationships among tasks are satisfied. The applied encoding and decoding processes are shown in FIGURE 3.

After performing the encoding and decoding processes, the fitness function of each water flow (solution) has to be calculated. For UALBP-1, the line efficiency \( (LE) \) and the variation of workload \( (VW) \) are calculated using Equation (1).

Algorithm 1: [TS] = encoding process \( (\rho, \theta = (\eta, \lambda)) \)
1. input: task priority vector \( (\rho) \), precedence network \( (\theta) \)
2. set \( \eta' = \emptyset \) \( (\eta' \subseteq \eta) \)
3. set \( \eta'' = \eta \)
4. set \( TS = \emptyset \)
5. repeat
6. for task \( j = 1 \) to \( n \) do
7. if \( j \in \eta'' \) and all its precessors and successors are included \( TS \)
8. set \( \eta'' = \eta'' \cup \{j\} \)
9. end
10. end
11. choose task \( i \) in \( \eta'' \) with the maximum priority according to \( \rho \)
12. insert the above task into the next vacant position in \( TS \) and remove it from \( \eta'' \) and \( \eta'' \)
13. until \( TS \) is complete
14. Output: TS

Algorithm 2: [feasible UALBP solution]=decoding process \( (TS, CT, t_j) \)
1. input: TS, CT, tasks’ time \( (t_j) \)
2. \( m = 1 \); (the first station is opened)
3. repeat
4. for \( j = 1 \) to \( n \) do
5. pick the task in \( j \)th position of TS \( (TS_j) \)
6. if \( TS_j \neq \emptyset \) and \( t(S_j) < CT \) and all predecessors and successors of task \( TS_j \) are already assigned \( t(S_j) \) is the time of task \( TS_j \)
7. assign the relating task to station \( k \) and put \( TS_j = 0 \)
8. \( t(S_k) = t(S_k) + t(TS_j) \)
9. end
10. end
11. \( m = m + 1 \)
12. until all elements of TS are zero
13. Output: Feasible UALBP solution

FIGURE 3. The encoding and decoding processes for WFA.

Since this study intends to maximize the \( LE \) and to minimize the \( VW \) simultaneously as per the considered objectives, the fitness function of each flow \( (FF) \) is calculated by applying the minimum deviation method [31] using equation (7):

\[
FF = \alpha \times \frac{\text{Max}(LE) - \text{LE}}{\text{Max}(LE) - \text{Min}(LE)} + \beta \times \frac{\text{VW} - \text{Min}(VW)}{\text{Max}(VW) - \text{Min}(VW)}
\]

where \( \text{Min}(LE) \) and \( \text{Min}(VW) \) are the minimum of \( LE \) and \( VW \) found so far, respectively and \( \text{Max}(LE) \) and \( \text{Max}(VW) \) are the maximum of \( LE \) and \( VW \) obtained to this point, respectively. As the \( LE \) is considered to be more important than the \( VW \) for the considered UALBP-1, it is assumed that \( \alpha \gg \beta \), so that it is guaranteed that the secondary objective \( (VW) \) is optimized in a hierarchy order and only after the primary objective \( (LE) \) is fully satisfied. Thus, the coefficients \( \alpha \) and \( \beta \) are chosen in a meaningful manner to give a very strong priority to the first objective. In such circumstances, after some pilot studies the relative importance of \( LE \) and \( VW \) were chosen as \( \alpha =0.9 \) and \( \beta=0.1 \), respectively.

C. SPLITTING AND MOVING OPERATIONS

While passing the rugged landscapes, the flows are split into several sub-flows and then start to move towards lower terrains in their neighborhood. These mechanisms are simulated in WFA by performing the splitting and moving operations.

Assuming a flow \( f \) with mass \( W_f \) and velocity \( V_f \), the splitting operation for flow \( f \) is performed when the product of its mass and velocity \( (W_f \times V_f) \) exceeds a predefined threshold \( (T) \). As the splitting operation continues from the higher to the lower terrains, the current number of flows (NF),
can grow exponentially. To control this growth, an upper bound is defined on the number of sub-flows which can be derived from each flow (NS). As a result, the number of streams forked from flow \( f \) (\( n_{sf} \)) is calculated by equation (8).

\[
  n_{sf} = \min\{\max\{1, \text{int}\left(\frac{W_{v,v}}{T}\right)\}, NS\}
\]

(8)

where function int() calculates the closest integer number to a value. When the number of streams flowing from each flow is determined, the positions to which those streams are going to move have to be determined. This mechanism is called the moving operation in WFA and is performed by applying one/several neighbourhood search mechanism(s) that can lead to better neighbourhood solutions. In this paper, since WFA deals with the permutation of numbers as the representation of solutions, the 2-opt and 3-opt operators shown in FIGURE 4 (a) and (b), respectively, are applied as the local search mechanisms to find better neighbourhood solutions. In this regard, first the 2-opt operator is performed. If the 2-opt operator could not find a better solution, the 3-opt operator will be performed, subsequently. These operations are repeated until a maximum number of local search iterations are reached.

At the end of a splitting and moving operation, the original flow is discarded because it has been replaced by its sub-flows.

**D. MERGING OPERATION**

In contrast to the splitting and moving operations, wherein more flows are generated and move towards lower terrains, there is a possibility that they meet each other again in the lower terrains. When this happens they will merge. Inspired by this phenomenon, WFA uses a merging operation to tune its population size adaptively and avoid any redundant searches.

When flows \( f \) and \( h \) meet each other at the same location, which happens in WFA when solutions with the same fitness function are found, the flow \( h \) will be discarded and its mass and velocity will be merged with flow \( f \) using equations (11) and (12).

\[
  W'_f = W_f + W_h
\]

(11)

\[
  V'_f = \frac{V_f W_f + V_h W_h}{W_f + W_h}
\]

(12)

**E. EVAPORATION OPERATION**

Over the course of water flows on the terrains, some parts of them evaporate to atmosphere. This phenomenon helps water flows expand their search territory and avoid being trapped in the local valleys (local optima). Accordingly, WFA applies an evaporation mechanism on each flow existing in the current iteration to imitate this phenomenon. Thus, when the evaporation period is reached (i.e., every \( t \) iterations) a portion (\( 1/t \)) of each flow’s mass (e.g., \( W_f' \)) is removed using equation (13), resulting to a new mass for flow \( f \) (\( W_f'' \)).

\[
  W_f'' = (1 - \frac{1}{t})W_f'
\]

(13)

**F. PRECIPITATION OPERATION**

The evaporated water, then returns to the terrain through the rainfall mechanism to complete the water cycle in nature. This phenomenon is performed by WFA using two types of precipitation operations, namely, irregular and regular precipitations.

Irregular precipitation is applied when all flows in the current iteration, i.e., \( NF \) flows, are stagnant with zero velocity. To help flows escape from local traps, all flows are evaporated and through rainfall the same amount of flows will be precipitated so that the locations of new flows are stochastically deviated from their origin. Then new masses are generated which are stochastically deviated from their origins. Additionally, their velocities are initialized by setting them equal to \( V_0 \).

\[
  W'_f = \frac{W_f}{\sum_{i=1}^{NF} W_i} W_0
\]

(14)

To diversify the search domain, new solutions have to be generated which are stochastically deviated from their origins. Therefore, the inverse operator had been used by WFA as
shown in FIGURE 5 to obtain new solutions/flows deviated from the current solutions.

Regular precipitation is performed in accordance with the evaporation operation every $t$ iterations. Using this type of precipitation, the evaporated water is precipitated and results in new flows that are deviated from their origin in positions. Considering the total mass of evaporated water which amounts to $W_0 - \sum_{f=1}^{NF} W_f$, the mass of the new precipitated flow $h$ ($W_h'$), can be calculated by equation (15).

$$W_h' = \frac{W_h}{\sum_{f=1}^{NF} W_f} (W_0 - \sum_{f=1}^{NF} W_f)$$  \hspace{1cm} (15)

After adding the new flows generated by the regular precipitation to the existing flows, the number of current flows (solutions) will increase.

As a result of both irregular and regular precipitations, the number of flows can grow considerably and might give similar fitness functions. Thus, after each type of precipitation operation, a flow merging operation will be performed to avoid the excessive growth of flows (solutions) as well as possible redundant searches.

These operations are repeated until a stopping condition is met. The general procedure of the proposed WFA for UALBP-1 is shown in FIGURE 6.

### IV. COMPUTATIONAL STUDY

In this section, the computational study is performed where after tuning the WFA parameters, the computational results of the WFA on the real case study and the standard test problems are presented and discussed.

#### A. TUNING THE WFA PARAMETERS

The choice of values for the algorithm parameters can influence the quality of solutions, as such the Taguchi Method (TM) is applied to tune the parameters of the proposed WFA. The reason TM is applied, is that it uses the fractional rather than full factorial designs to run the experiments which can result in considerable savings in both time and cost of analysis [32]. In TM, parameters are categorised as: controllable ($C$) and uncontrollable ($U$), and it selects the best levels for $C$ so that the algorithm is robust in the presence of variation in $U$. TM applies two types of analysis: (1) analysis of variance, and (2) analysis of signal-to-noise ratio ($S/N$). The former is applied when the analysis is based on a single replication of experiments, while the latter is implemented when multiple replications of experiments exist [32]. In this study, the $S/N$ ratio is applied and TM aims to find the best level of parameters so that the $S/N$ ratios are maximized.

TM is applied to tune three main parameters of WFA i.e., $W_0$, $V_0$, and $t$. Different levels of WFA parameters are given in Table II.

<table>
<thead>
<tr>
<th>Level</th>
<th>$W_0$</th>
<th>$V_0$</th>
<th>$t$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
<td>5</td>
<td>20</td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>10</td>
<td>80</td>
</tr>
<tr>
<td>3</td>
<td>15</td>
<td>15</td>
<td>140</td>
</tr>
<tr>
<td>4</td>
<td>20</td>
<td>20</td>
<td>200</td>
</tr>
</tbody>
</table>

Considering the number of parameters and their levels, the $L_{16}(4^3)$ design is selected for the experiments according to the TM. For precise parameter tuning of WFA for UALBP-1, the case studied and the standard test problems existing in the literature [33] are categorized into small, medium, and large sized problems considering their number of tasks and arrows.

FIGURE 5. The inverse operation applied by WFA to deviated from a solution.

FIGURE 6. The flow diagram of the proposed WFA for UALBP-1.
Table III presents the problem characteristics including size, name, # of tasks, # of arrows, CTs, and # of CTs considered. TM is applied to tune the WFA for each problem size, separately. According to this table, different problems and the relating CTs have to be considered while tuning WFA parameters. Thus, for each experiment of TM, WFA is run for the total number of CTs in that size. For instance, for small size, since there are 11 CTs in total, WFA is performed 11 times. Since the mean and the variance of the responses vary consistently, the nominal is best $S/N$ ratio calculated by equation (16) is applied in this study [34].

$$\frac{\bar{y}}{S} = \text{Nominal is best} \left(10 \log \left( \frac{\bar{y}^2}{S^2} \right) \right)$$

where $\bar{y}$ and $S^2$ are the mean and the sample variance of the responses. The TM aims to maximize the $S/N$ ratio for each parameter against its levels. By applying TM on each problem size using Minitab, the $S/N$ ratio diagrams shown in FIGURE 7 are obtained. Based on the $S/N$ ratio diagrams, the optimum level of WFA parameters for each problem size are shown in Table IV.

### TABLE III

**Classification of the considered UALBPs**

<table>
<thead>
<tr>
<th>Size</th>
<th>Problem</th>
<th># of tasks</th>
<th># of Arrows</th>
<th>CTs</th>
<th># of CTs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small</td>
<td>Jackson</td>
<td>11</td>
<td>15</td>
<td>13.21</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Mitchell</td>
<td>21</td>
<td>30</td>
<td>14.15</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Heskia</td>
<td>28</td>
<td>42</td>
<td>138.205</td>
<td>3</td>
</tr>
<tr>
<td></td>
<td>Case study</td>
<td>29</td>
<td>34</td>
<td>40.45</td>
<td>3</td>
</tr>
<tr>
<td>Medium</td>
<td>Sawyer</td>
<td>30</td>
<td>40</td>
<td>27.30,33.54,75</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Kilbridge</td>
<td>45</td>
<td>70</td>
<td>57.184</td>
<td>2</td>
</tr>
<tr>
<td></td>
<td>Tonge</td>
<td>70</td>
<td>100</td>
<td>176.364,410.468,527</td>
<td>5</td>
</tr>
<tr>
<td>Large</td>
<td>Arcus</td>
<td>83</td>
<td>115</td>
<td>5853,6842,7571,8412,8998,10816</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>Arcus2</td>
<td>111</td>
<td>178</td>
<td>5755,10027,10743,11378,17067</td>
<td>5</td>
</tr>
</tbody>
</table>

**B. COMPUTATIONAL RESULTS**

To demonstrate the efficiency and robustness of the proposed WFA in addressing the UALBP-1, we compare its results with other algorithms existing in the UALBP literature, namely Multi-Objective Hybrid Improvement Heuristic (MOHIH), Multi-Objective Genetic Algorithm (MOGA) and Grouping Evolution Strategy (GES) proposed by [13], [11] and [12], respectively. For this comparison, the real case considered and different test problems in the literature, which have been classified in Table III into three problem sizes, are solved by WFA for 10 times and the best results are compared with the best results reported by MOHIH, MOGA and GES. For this study, WFA was coded and executed using MATLAB on a PC with 2.0 GHz processor and 4 GB of RAM. The stopping criterion for WFA was set to 100 iterations with the same fitness function as it was considered in [11]. The minimum and the maximum computational times spent by WFA to solve the considered problems are ranged between 0.1s and 170s, respectively. In Table V, the first four columns represent the problem characteristics including size, name, CT, and the optimal $M$ (number of station) in the relating literature. For the rest of the columns, the performance of the algorithms are compared in terms of the optimal/best-found values of the optimization criteria i.e., the number of stations ($M$), the line efficiency ($LE\%$), and the variation of workload ($VW$).
According to results shown in Table V, one can observe that when considering the number of stations as well as LE, all four algorithms namely MOHIH, MOGA, GES, and WFA, have solved the UALBP-1 test problems to an optimal state in terms of \( M \) and LE. The results can be justified by prioritizing the number of stations (LE) as the primary objective compared to VW as the secondary objective. For the real case study, it was found that the CT of 50 seconds resulted in the minimum number of stations i.e. 5 and the maximum line efficiency of 97.6 percent, accordingly. Thus, a CT of 50 seconds is selected as the best scenario for the case study.

In terms of VW, Table V shows that in all small-size problems, WFA has resulted to the same minimum VW obtained by either MOHIH, MOGA or GES. This can be justified by the smaller and simpler search space in the small problems in which the algorithms can reach to (near) optimal solutions without difficulty. For the case study, the VW obtained for CT of 50 has led to the minimum VW compared to other scenarios, i.e., 40 and 45 seconds, which is in line with the preferred scenario in terms of M or LE. However, when the problem size increases to medium-size, the WFA could find a lower VW in 4 out of 12 problems, while in the rest of the problems similar results were obtained by all algorithms. For the large-sized problems, putting aside the only problem

<table>
<thead>
<tr>
<th>Size</th>
<th>Problem</th>
<th>CT</th>
<th>MOHIH M_LE (%)</th>
<th>MOHIH VW</th>
<th>MOGA M_LE (%)</th>
<th>MOGA VW</th>
<th>GES M_LE (%)</th>
<th>GES VW</th>
<th>Meta-heuristics WFA M_LE (%)</th>
<th>WFA VW</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small</td>
<td>Jackson</td>
<td>13</td>
<td>88.5 0.042</td>
<td></td>
<td>88.5</td>
<td></td>
<td>93.7 0.023</td>
<td></td>
<td>88.5 0.042</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Mitchell</td>
<td>18</td>
<td>87.5 0.023</td>
<td></td>
<td>87.5</td>
<td></td>
<td>93.7 0.023</td>
<td></td>
<td>93.7 0.023</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Haskey</td>
<td>21</td>
<td>5 100</td>
<td></td>
<td>5 100</td>
<td></td>
<td>92.7 0.005</td>
<td></td>
<td>92.7 0.005</td>
<td></td>
</tr>
<tr>
<td>Medium</td>
<td>Sawyer</td>
<td>27</td>
<td>92.3 0.023</td>
<td></td>
<td>92.3</td>
<td></td>
<td>99.9 0.001</td>
<td></td>
<td>99.9 0.001</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Kilbridge</td>
<td>57</td>
<td>96.8 0.007</td>
<td></td>
<td>96.8</td>
<td></td>
<td>96.8 0.007</td>
<td></td>
<td>96.8 0.007</td>
<td></td>
</tr>
<tr>
<td>Large</td>
<td>Arcus1</td>
<td>5853</td>
<td>92.4 0.012</td>
<td></td>
<td>92.4</td>
<td></td>
<td>93.8 0.004</td>
<td></td>
<td>93.8 0.004</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Arcus2</td>
<td>5853</td>
<td>92.4 0.012</td>
<td></td>
<td>92.4</td>
<td></td>
<td>93.8 0.004</td>
<td></td>
<td>93.8 0.004</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Real</td>
<td>45</td>
<td>97.9 0.004</td>
<td></td>
<td>97.9</td>
<td></td>
<td>97.9 0.004</td>
<td></td>
<td>97.9 0.004</td>
<td></td>
</tr>
</tbody>
</table>

*Not available since the problem is not solved by this algorithm in the related study.*
heuristics rely on a fixed number of solutions, there is no guarantee that either of these algorithms can search the solution space efficiently while avoiding the local traps and premature convergence. Thus, for the first time, a novel algorithm. i.e., WFA inspired from the behaviour of water flows traversing from higher to lower altitudes, is applied to solve the UALBP considering two objectives. The primary objective was to maximize the line efficiency (LE) and the secondary objective was to minimize the variation of GES in the literature over a set of known test domains. The computational results show that the maximum LE was obtained by the proposed WFA and it outperformed MOHH, MOGA and GES especially in the medium to large-sized test problem domain in terms of VW. Speculating on future research directions, other real-world practical situations, such as zoning constraints and/or ergonomic considerations, stochastic task times, etc., could also be studied in the context of UALBP. Additionally, the proposed WFA can be modified to address other types of ALBP such as two-sided and mixed-model assembly lines.

REFERENCES


[31] M. Fathi, A. Nourmohammadi, A. H. C. Ng, A. Syberfeldt, and


AMIR NOURMOHAMADI holds a PhD in Industrial and Systems Engineering from Tarbiat Modares University, Tehran, Iran. He has been a research assistant in the Department of Production and Automation Engineering in the University of Skövde where he engaged in research projects in collaboration with industry. His main research interests include the application of operations research and soft computing techniques in production and logistics.

MASOOD FATHI received his Ph.D. in industrial engineering from the University of Navarra, Spain. He is currently a Senior Lecturer in Department of Production and Automation Engineering in the University of Skövde, Sweden. His main research interests include the applications of operations research, and optimization techniques in production and logistics. He has been involved as a Leader or/and a Researcher in several national and international research projects in collaboration with industry.

MOSTAFA ZANDIEH accomplished his B.Sc. in Industrial Engineering at Amirkabir University of Technology, Tehran, Iran (1994–1998), and M.Sc. in Industrial Engineering at Sharif University of Technology, Tehran, Iran (1998–2000). He obtained his Ph.D. in Industrial Engineering from Amirkabir University of Technology, Tehran, Iran (2000–2006). Currently, he is an Associate Professor at Department of Industrial Management, Management and Accounting Faculty, SBU, G.C., Tehran, Iran. His research interests are Production Planning and Scheduling, Financial Engineering, Quality Engineering, Applied Operations Research, Simulation, and Artificial Intelligence techniques in the areas of manufacturing systems design.

Morteza Ghobakhloo is an assistant professor at the Department of Industrial Engineering, Minab Higher Education Center, University of Hormozgan. His research interests include lean and sustainable manufacturing, Industry 4.0, smart manufacturing, business value of information technology, and electronic supply chain management. His research have been published in many leading operations management and research journals such as IPR, IJAMT, IMDS, BITDM, JMTM, ITD, JSBED, JM2, IRMJ, JOEUC and JOCEC among others.