ORIGINAL RESEARCH

A hybrid method to solve reliability-cost-oriented bi-objective machine configuration problem for a flow shop system

Cheng-Ta Yeh[1] · Louis Cheng-Lu Yeng[2] · Yi-Kuei Lin[2,3,4,5] · Yu-Lun Chao[2]

Abstract

Machine configuration is a crucial strategic decision in designing a flow shop system (FSS) and directly affects its performance. This involves selecting device suppliers and determining the number of machines to be configured. This study addresses a bi-objective optimization problem for an FSS that considers repair actions and aims to determine the most suitable machine configuration that balances the production reliability and purchase cost. A nondominated sorting genetic algorithm II (NSGA-II) is used to determine all the Pareto solutions. The technique for order preference by similarity to an ideal solution is then used to identify a compromise alternative. It is necessary to assess the production reliability of any machine configuration identified by the NSGA-II. The FSS under the machine configuration is modeled as a multistate flow shop network, and Absorbing Markov Chain and Recursive Sum of Disjoint Products are integrated into the NSGA-II for reliability evaluation. The experimental results of solar cell manufacturing demonstrate the applicability of the proposed hybrid method and validate the efficiency of the NSGA-II compared with an improved strength Pareto evolutionary algorithm.

Keywords Multistate flow shop network · Production reliability · Bi-objective · Absorptive Markov Chain · Nondominated sorting genetic algorithm II (NSGA-II) · Technique for order preference by similarity to an ideal solution

✉ Yi-Kuei Lin
  yklin@nycu.edu.tw

  Cheng-Ta Yeh
  145712@mail.fju.edu.tw

[1] Department of Information Management, Fu Jen Catholic University, New Taipei City 242, Taiwan

[2] Department of Industrial Engineering and Management, National Yang Ming Chiao Tung University, Hsinchu 300, Taiwan

[3] Department of Business Administration, Asia University, Taichung 413, Taiwan

[4] Department of Industrial Engineering and Management, Chaoyang University of Technology, Taichung 413, Taiwan

[5] Department of Mathematics, Computer Science and Engineering, Graphic Era Deemed to Be University, Dehradun, India

 Springer

1 Introduction

A flow shop system (FSS) is a manufacturing system that usually includes multiple production lines to share the order demand for a single product (Brammer et al., 2022; Dios et al., 2018; Dolgui et al., 2021). Several workstations implement different processes in a production line, and each workstation comprises several identical machines (Yu et al., 2018). Because each machine may operate to provide a capacity or fail with a probability, the production capacity of each workstation has multiple states that follow a binomial probability distribution. Hence, the FSS can be modeled as a multistate flow network (Lin & Chang, 2012; Lin & Chen, 2022; Nguyen, 2022), namely, the multistate flow shop network (MFSN). In some industries, such as the printed wire-board and automobile manufacturing industries, defective work-in-process (WIP)/productions may be produced in specific workstations because of yield rates; they are then mostly repaired from the indicated workstations as much as possible to reduce the waste of being scraped (Hadjinicola, 2010; Sarker et al., 2008). Accordingly, Lin and Chang (2015) proposed the production reliability, which is defined as the probability that d units of order demand can be manufactured by an MFSN with repair actions.

The design of an FSS involves the manufacturing system configuration. The concept of manufacturing system configuration is multifaceted and encompasses various aspects, such as the physical arrangement of machines, machine selection, and task assignment (Saxena & Jain, 2012). In particular, machine configuration is a significant consideration in manufacturing system design, and traditionally, it involves decisions regarding appropriate machine types/suppliers and configuration of the number of machines (Chan et al., 2005; Chehade et al., 2012). Bukchin and Tzur (2000) developed an exact branch-and-bound algorithm and heuristic procedure to address the problem of machine selection and task assignment in a flexible assembly line to minimize the total machine cost. Bukchin and Rubinovitz (2003) extended their study to address the machine configurations in the FSSs, including the parallel workstation configurations. They proposed an integer linear programming model and branch-and-bound algorithm to minimize the number of workstations and total cost. Li et al. (2011) introduced the hierarchical compositional properties of components assembled in repetitive patterns during the automotive Li-ion battery pack manufacturing. They developed a recursive algorithm to generate processing sequence planning and machine configurations, minimize machine investment costs, and potentially increase the system throughput. Hossain and Sarker (2016) considered a multistage manufacturing system with an inspection station at the end of the production line to make decisions regarding defective products. They formulated a fractional mixed-integer nonlinear programming problem to minimize the unit cost of production by determining the optimal number and locations of offline rework stations. Oesterle et al. (2019) addressed the machine configuration problem by considering product design alternatives and assembly-line balancing. They proposed a detailed mathematical cost model to quantify the complex and interconnected consequences of product design, manufacturing technology, and process choices in a single-cost metric. Several metaheuristic algorithms, including the evolutionary algorithms, ant colony optimization, and artificial bee colonies, have been compared to solve this problem. Niroomand (2021) discussed the machine configuration problem while considering assembly line balancing to minimize the costs of station setup and machine purchase. This study aims to determine the optimal machine configuration decision and employs a combination of an artificial electric field algorithm and simulated annealing to solve this problem. Overall, studies related to the machine configurations assume that the machine states are deterministic. In addition, no study has simultaneously considered the decisions regarding machine types/suppliers and number of configured machines.

Because machine states are stochastic (Lin & Chang, 2012; Lin & Chen, 2022; Nguyen, 2022), designing a reliable FSS considering the machine configuration is a critical challenge. Thus, it is necessary to maximize the production reliability of the FSS. Yeh et al. (2023) utilized a genetic algorithm to solve a reliability-oriented machine configuration problem for a manufacturing system. However, assigning machines to workstations incurs high purchasing costs (Bajestani et al., 2009). Thus, the tradeoff between reliability maximization and cost minimization must be considered. Although Lin et al. (2019) discussed an issue related to the reliability and cost of a bi-objective machine configuration problem, they assumed that the manufacturing system had no repair actions. Accordingly, our study addresses the machine configuration problem of production reliability and total purchase cost optimization for an FSS with repair actions.

This problem involves the evaluation of the production reliability of an MFSN and determination of the optimal machine configuration. When searching for a machine configuration, it is necessary to evaluate production reliability. Lin et al. (1995) stated that the multistate flow network reliability evaluation problem is NP-hard. The production reliability evaluation problem is a typical multistate flow network reliability evaluation problem; therefore, it is NP-hard (Lin & Chang, 2015). Zhang and Bard (2006) and Jahromi and Tavakkoli-Moghaddam (2012) indicated that the machine configuration optimization problem for manufacturing systems is NP-hard. Therefore, the problem to be addressed must be NP-hard. Lin et al. (2017) and Yeh et al. (2017) evaluated production reliability through the analysis of minimal-consumed capacity patterns for demand d (d-MCCPs), which are derived from the raw material/WIP/product flow into each workstation by integrating a path decomposition method of Lin and Chang (2015). Subsequently, the recursive sum of disjoint products (RSDP) calculates the union probability of the d-MCCPs to obtain the production reliability. The path decomposition method separates a production line into normal and repair paths to determine the flow processed by each workstation. If z repair actions exist, the path decomposition method generates 2^z paths, including one normal path and $2^z - 1$ repair paths for a production line (Lin & Chang, 2015). The computation of the flow traveling through each workstation is time-consuming, with more workstations and repair actions in each production line. Instead, by referring to Bowling et al. (2004) and Pillai and Chandrasekharan (2008), we adopt the Absorptive Markov Chain (AMC) to determine the flow transition status between a pair of workstations to obtain the input flow of each workstation in the MFSN without enumerating all the regular and repair paths.

However, because the addressed machine configuration problem is NP-hard, it must be solved in a reasonable time using metaheuristic algorithms, such as the genetic algorithms, particle swarm optimization, and simulated annealing (Juan et al., 2015). Our problem is also bi-objective. For different types of bi-objective optimization problems, the Nondominated Sorting Genetic Algorithm II (NSGA-II) has usually presented better efficiency than that of the several well-known multi-objective metaheuristic algorithms, such as the multi-objective evolutionary algorithm based on the decomposition and improved strength Pareto evolutionary algorithm (SPEA2) (Lin et al., 2019; Liu et al., 2020; Silva et al., 2022; Yeh, 2020). Ma et al. (2023) highlighted the advantages of the NSGA-II: (1) it maintains the spread of the solutions, and (2) converges to the exact nondominated solutions well. Although Deb and Jain (2014) extended the NSGA-II to propose NSGA-III, it is more suitable for handling multi-objective (having four or more objectives) problems. Thus, the NSGA-II is used to search for the Pareto solutions to the addressed bi-objective problem. However, the output from the NSGA-II is a nondominated/Pareto set, and system administrators always need to select an alternative from the set (Lin & Yeh, 2012). Our study uses the technique for order preference by similarity to an ideal solution (TOPSIS) (Hwang et al., 1993) to determine a

compromise alternative. TOPSIS is widely used in multi-criteria decision-making because of its well-founded logical structure, simultaneous consideration of ideal and non-ideal solutions, and ease of calculation. Unlike other methods, TOPSIS provides a comprehensive view of alternatives by assigning individual values to each, thereby enabling a better understanding of the differences between the alternatives and varying criteria (Durak et al., 2022).

To address the problem of FSS with repair actions, this study proposes a novel method that combines the AMC, RSDP, NSGA-II, and TOPSIS. The subsequent sections present the network model and optimization formulation (Sect. 2), production reliability evaluation based on AMC and RSDP (Sect. 3), NSGA-II procedure (Sect. 4), and TOPSIS approach for selecting a compromise alternative (Sect. 5). To demonstrate the effectiveness of our hybrid method, we apply it to a practical case of solar cell manufacturing (Sect. 6) and compare the efficiencies of NSGA-II and SPEA2. Finally, we summarize our conclusions in Sect. 7.

2 Bi-objective machine configuration optimization modeling for FSS

The FSS can be described as a network $(\mathbf{N}, \mathbf{A})$ with w production lines, $L_1, L_2, \ldots, L_m$ to produce the same product, and n workstation in each production line, in which $\mathbf{N} = \{n_{j,i} | j = 1, 2, \ldots, w, i = 1, 2, \ldots, m\}$ represents a set of m workstations/nodes and $\mathbf{A}$ represents the set of arcs/transport devices. The workstation $n_{j,i}$ signifies the process i in production line L_j. Let I_j and O_j denote the raw material input and product output of L_j for $j = 1, 2, \ldots, m$, respectively. In each production line L_j, there are z repair workstations, $n_{j,\beta_1}, n_{j,\beta_2}, \ldots, n_{j,\beta_z}$, with $\beta_e \in \{1, 2, \ldots, m\}$. If defect product flow is detected at the repair workstation n_{j,β_e}, it will be repaired from a pointed workstation n_{j,α_e} to the workstation n_{j,β_e} for $e = 1, 2, \ldots, z$, where n_{j,α_e} is the initial node of the repair node n_{j,β_e} with $\alpha_e \in \{1, 2, \ldots, \beta_e\}$. Figure 1 illustrates a general FSS network with two production lines to process raw material inputs of I_1 and I_2 and then generate the product outputs of O_1 and O_2. Each production line has two black nodes denoting the repair workstations.

For each process i, q_i device suppliers supply the machines. Each machine has four features: purchase cost $c_{u,i}$, machine reliability $r_{u,i}$, production capacity $h_{u,i}$, and yield rate $p_{u,i}$ for $u = 1, 2, \ldots, q_i$ and $i = 1, 2, \ldots, m$. Let $X = (x_{1,1}, x_{2,1}, \ldots, x_{1,m}, x_{2,1}, x_{2,2}, \ldots, x_{2,m}, \ldots, x_{w,1}, x_{w,2}, \ldots, x_{w,m})$ be a supplier selection with $x_{j,i} \in \{1, 2, \ldots, q_i\}$ signifying the index of the selected supplier for the workstation $n_{j,i}$. Then, let $Y = (y_{1,1}, y_{2,1}, \ldots, y_{1,m}, y_{2,1}, y_{2,2}, \ldots, y_{2,m}, \ldots, y_{w,1}, y_{w,2}, \ldots, y_{w,m})$ denote a machine amount pattern with $y_{j,i} \in \{1, 2, \ldots, M_i^{x_{j,i}}\}$ representing the number of machines provided by the supplier $x_{j,i}$, where $M_i^{x_{j,i}}$ expresses the available quantity of machines supplied by the supplier $x_{j,i}$. The pair (X, Y) is represented as a machine configuration. We denote $C(X, Y)$ as the total purchase cost of
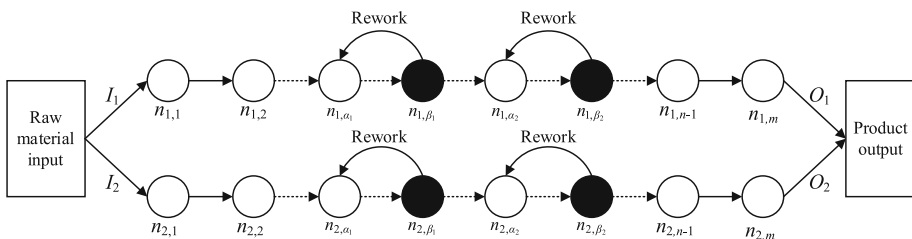

Fig. 1 A general FSS network with two repair workstations

machine configuration (X, Y) and formulate it as:

$$C(X, Y) = \sum_{j=1}^{w} \sum_{i=1}^{m} c_{x_{j,i}} y_{j,i}, \tag{1}$$

where $c_{x_{j,i}} y_{j,i}$ indicates the purchase cost of $n_{j,i}$.

Associated with the machine configuration (X, Y), each workstation with several identical machines has multiple capacity states. Therefore, FSS network (N, A) is viewed as an MFSN. Let $S = (s_{1,1}, x_{2,1}, \ldots, s_{1,m}, s_{2,1}, s_{2,2}, \ldots, s_{2,m}, \ldots, s_{w,1}, s_{w,2}, \ldots, s_{w,m})$ be the current capacity vector of (N, A), where $s_{j,i}$ is the production capacity of workstation $n_{j,i}$ and may be 0, $h_{x_{j,i,i}}$, 2 $h_{x_{j,i,i}}, \ldots$, or $y_{j,i} h_{x_{j,i,i}}$. The following equation defines the probability of production capacity $s_{j,i}$.

$$\Pr(s_{j,i} = \tau h_{x_{j,i,i}}) = \binom{y_{j,i}}{\tau} (r_{x_{j,i,i}})^{\tau} (1 - r_{x_{j,i,i}})^{y_{j,i}-\tau}, \tag{2}$$

where $\tau \in \{0, 1, 2, \ldots, y_{j,i}\}$ indicates the number of regular machines in $n_{j,i}$. Let S denote the set of all the capacity states of (N, A) successfully manufacturing order demands d associated with machine configuration (X, Y). The production reliability of configuration (X, Y) is defined as:

$$R(d, (X, Y)) = \sum_{S \in S_{(X,Y),d}} \Pr(S), \tag{3}$$

where $\Pr(S) = \Pr(s_{1,1}) \times \Pr(s_{1,2}) \times \ldots \times \Pr(s_{w,m})$, and $\Pr(s_{j,i})$ is calculated using Eq. (2).

According to the network model described earlier, the following mathematical model is built to describe the bi-objective problem:

$$Maximize \ R(d, (X, Y)) = \sum_{S \in S_{(X,Y),d}} \Pr(S), \tag{4}$$

$$Minimize \ C(X, Y) = \sum_{j=1}^{w} \sum_{i=1}^{m} c_{x_{j,i,i}} y_{j,i} \tag{5}$$

Subject to

$$\sum_{j:x_{j,i}=u} y_{j,i} \leq M_i^{x_{j,i}} \quad \text{for } u = 1, 2, \ldots, q_i \text{ and } i = 1, 2, \ldots, m, \tag{6}$$

$$x_{j,i} \in \{1, 2, \ldots, q_i\} \text{ for } j = 1, 2, \ldots, w \text{ and } i = 1, 2, \ldots, m, \text{ and} \tag{7}$$

$$y_{j,i} \in \left\{1, 2, \ldots, M_i^{x_{j,i}}\right\} \text{ for } j = 1, 2, \ldots, w \text{ and } i = 1, 2, \ldots, m. \tag{8}$$

Equations (4) and (5) are the objective functions of production reliability and purchase cost. Constraint (6) expresses that the number of machines supplied by each supplier cannot exceed the available quantity. Constraints (7) and (8) limit the domains of decision variables $x_{j,i}$ and $y_{j,i}$, respectively.

Assumptions

(1) Each transport device is perfectly reliable.
(2) The defective product flow from the repair workstation n_{j,β_e} must be repaired from the pointed workstation n_{j,α_e} and can only be repaired once. This implies that such defective flow is repaired until a usable state is reached. If the defective flow after reworking is still defective, it is non-reparable and is scrapped (Lin & Chang, 2013).
(3) The machines configured to workstation $n_{j,i}$ must be from the same supplier to ensure consistency in product quality.
(4) The states of the different workstations under machine configuration (X, Y) are statistically independent.

Fig. 2 The framework of the hybrid method
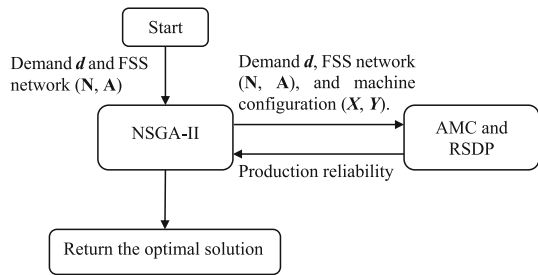
(5) Owing to the distributed nature of production lines, the movement of products within one production line cannot be transferred to other production lines.

Figure 2 illustrates the framework of the proposed hybrid method. The NSGA-II searches for the optimal machine configuration, as described in Sect. 4. The AMC and RSDP calculate the production reliability for each machine configuration generated by the NSGA-II, which is discussed in Sect. 3.

3 Production reliability evaluation using AMC and RSDP

Calculating the production reliability $R(d, (X, Y)) = \sum_{S \in S_{(X,Y),d}} \Pr(S)$ may be time-consuming or result in an out-of-memory error because it requires enumerating all $S \in S$. Lin and Chang (2015) and Yeh et al. (2017) hence recommended searching for all d-MCCPs and then expressing the production reliability as a union probability of all d-MCCPs, where each d-MCCPs is the minimum vector in S. This section proposes an AMC-based approach integrated with the RSDP to compute the production reliability for a machine configuration (X, Y). An AMC-based approach is used to enumerate all d-MCCPs. The union probability of all d-MCCPs is driven by the RSDP.

3.1 AMC-based approach to determine d-MCCPs

Let $\Gamma_{j,e} = \{n_{j,\alpha_e}, n_{j,\alpha_e+1}, \ldots, n_{j,\beta_e-1}, n_{j,\beta_e}\}$ be a set of sequential workstations for the repair process from n_{j,α_e} to n_{j,β_e}. An AMC transition matrix of production line L_j, denoted by $\mathbf{B}_j$, comprises four elements: the transient-to-transient status matrix $\mathbf{U}_j$, transient-to-absorbing status matrix $\mathbf{V}_j$, zero matrix $\mathbf{0}$, and identity matrix $\mathbf{I}$, which is represented as:

$$\mathbf{B}_j = \begin{bmatrix} \mathbf{U}_j & \mathbf{V}_j \\ \mathbf{0} & \mathbf{I} \end{bmatrix}, \quad \text{for } j = 1, 2, \ldots, w. \tag{9}$$

The AMC transition matrix owns $m + \sum_e^z |\Gamma_{j,e}| + 2$ statuses. The transient-to-transient status matrix $\mathbf{U}_j$ is $m + \sum_e^z |\Gamma_{j,e}|$-by-$m + \sum_e^z |\Gamma_{j,e}|$. Each element in $\mathbf{U}_j$ represents the transition probability of an ordered pair of transient statuses (workstations). There are m transient statuses in the normal process of the production line L_j. The transient-to-absorbing status matrix $\mathbf{V}_j$ is $m + \sum_e^z |\Gamma_{j,e}|$-by-2, where "2" means both absorbing statuses: the flow is scrapped, or the product is successfully produced. All transition probabilities in $\mathbf{U}_j$ and $\mathbf{V}_j$ are assigned according to the yield rates of machine configuration (X, Y). Each element in $\mathbf{V}_j$ represents the transition probability from a transient status to an absorbing status. The zero matrix $\mathbf{0}$ is 2-by-$m + \sum_e^z |\Gamma_{j,e}|$, and $\mathbf{I}$ is a 2-by-2 identity matrix. Based on the definition of

the AMC transition matrix, each production line consists of $m + \sum_{e}^{z} |\Gamma_{j,e}| +$ two statuses for $j = 1, 2, \ldots, w$.

The expected value matrix of production line L_j is denoted by $\mathbf{T}_j$ and is expressed as:

$$\mathbf{T}_j = \left[\mathbf{I} - \mathbf{U}_j\right]^{-1}, \quad \text{for } j = 1, 2, \ldots, w. \tag{10}$$

Each element in the first row and ηth column of $\mathbf{T}_j$, denoted by $t_{1,\eta}^{j}$, is the expected value of one unit of product flow arriving at the ηth transient status from the first transient status. Subsequently, the absorption probability matrix denoted by $\mathbf{E}_j$ is expressed as:

$$\mathbf{E}_j = \mathbf{T}_j \mathbf{V}_j, \quad \text{for } j = 1, 2, \ldots, w. \tag{11}$$

Elements $E_{1,1}^{j}$ and $E_{1,2}^{j}$ in the first row of $\mathbf{E}_j$ signify the probability of scrap flow and successful production of a finished product, respectively.

Let $\hat{\boldsymbol{I}} = (I_1, I_2, \ldots, I_w)$ be an input raw material vector and $\boldsymbol{F} = (f_{1,1}, f_{1,2}, \ldots, f_{1,m}, f_{2,1}, f_{2,2}, \ldots, f_{w,m})$ be a flow vector, where $f_{j,i}$ is the product flow processed by the workstation $n_{j,i}$. Because there are w production lines to share the demand d, the following constraint must be satisfied:

$$d = \sum_{j=1}^{w} d_j, \tag{12}$$

where d_j is the amount shared by the production line L_j. Let $\boldsymbol{d} = (d_1, d_2, \ldots, d_m)$ be the demand vector satisfying Constraint (12). It is necessary to determine the required input vector $\hat{\boldsymbol{I}}$ and flow vector $\boldsymbol{F}$ to obtain all d-MCCPs based on the yield rates. Considering probability $E_{1,2}^{j}$, the necessary raw material input I_j of production line L_j to manufacture d_j is computed as:

$$I_j = d_j / E_{1,2}^{j} \quad \text{for } j = 1, 2, \ldots, w. \tag{13}$$

Assuming the statuses of production line L_j as $1, 2, \ldots,$ and $m + \sum_{e}^{z} |\Gamma_{j,e}| + 2$. Let $\boldsymbol{\Phi}_{j,i}$ be a set of the statuses related to workstation $n_{j,i}$. Each product flow $f_{j,i}$ can be determined as:

$$f_{j,i} = I_j \sum_{\eta \in \boldsymbol{\Phi}_{j,i}} t_{1,\eta}^{j}, \quad \text{for } j = 1, 2, \ldots, w \text{ and } i = 1, 2, \ldots, m. \tag{14}$$

Under the machine configuration $(\boldsymbol{X}, \boldsymbol{Y})$, flow vector $\boldsymbol{F}$ generated by Eq. (14) should fulfill the following maximal production capacity constraint:

$$f_{j,i} \leq y_{j,i} h_{x_{j,i},i}, \quad \text{for } j = 1, 2, \ldots, w \text{ and } i = 1, 2, \ldots, m. \tag{15}$$

Any $\boldsymbol{F}$ that does not satisfy Constraint (15) implies that the FSS network $(\mathbf{N}, \mathbf{A})$ cannot fulfill the corresponding demand vector $\boldsymbol{d} = (d_1, d_2, \ldots, d_m)$. For convenience, let $\mathbf{F} = \{\boldsymbol{F} | \boldsymbol{F}$ fulfilling Constraint (15)$\}$. Any production capacity vector $\boldsymbol{S}$ transformed from a flow vector $\boldsymbol{F} \in \mathbf{F}$ via Eq. (16) is regarded as a d-MCCP (Lin & Chang, 2015).

$$s_{j,i} = \omega h_{x_{j,i},i} \text{ if } (\omega - 1)h_{x_{j,i},i} < f_{j,i} \leq \omega h_{x_{j,i},i} \text{ with } \omega \in \{0, 1, 2, \ldots, y_{j,i}\}$$
$$\text{for } j = 1, 2, \ldots, w \text{ and } i = 1, 2, \ldots, m. \tag{16}$$

The following solution procedure based on the AMC methodology is used to determine all the d-MCCPs.

AMC-based approach

Input: (1) Order demand d.

(2) FSS network $(\mathbf{N}, \mathbf{A})$: (i) production lines $L_1, L_2, \ldots, L_m$, (ii) repair workstations n_{j,β_1}, $n_{j,\beta_2}, \ldots, n_{j,\beta_z}$, and (iii) initial workstations $n_{j,\alpha_1}, n_{j,\alpha_2}, \ldots, n_{j,\alpha_z}$.
(3) Machine configuration $(\mathbf{X}, \mathbf{Y})$ with each configured machine's production capacity $h_{x_{j,i},i}$ and yield rate $p_{x_{j,i},i}$.

Step 1.   Use Eq. (17) to compute the probability of each workstation state under the machine configuration $(\mathbf{X}, \mathbf{Y})$.

$$\Pr(s_{j,i} = \tau h_{t_{j,i},i}) = \binom{y_{j,i}}{\tau}(r_{x_{j,i},i})^{\tau}(1 - r_{x_{j,i},i})^{y_{j,i}-\tau} \quad \text{for } t = 0, 1, 2, \ldots, y_{j,i},$$

$$j = 1, 2, \ldots, w, \text{ and } i = 1, 2, \ldots, m. \tag{17}$$

Step 2.   Calculate the probability of each status for all the production lines.

    1.1.   Build the AMC transition matrix $\mathbf{B}_j$ for each production line.

$$\mathbf{B}_j = \begin{bmatrix} \mathbf{U}_j & \mathbf{V}_j \\ \mathbf{0} & \mathbf{I} \end{bmatrix}, \quad \text{for } j = 1, 2, \ldots, w. \tag{18}$$

    1.2.   Calculate the expected value matrix $\mathbf{T}_j$ for each production line.

$$\mathbf{T}_j = [\mathbf{I} - \mathbf{U}_j]^{-1}, \quad \text{for } j = 1, 2, \ldots, w. \tag{19}$$

    1.3.   Calculate the absorbing probability matrix $\mathbf{E}_j$ for each production line.

$$\mathbf{E}_j = \mathbf{T}_j \mathbf{V}_j, \quad \text{for } j = 1, 2, \ldots, w.$$

The element $E_{1,2}^j$ in $\mathbf{E}_j$ represents the probability of successfully producing a finished product using production line L_j.

Step 3.   Find all demand patterns $\boldsymbol{d} = (d_1, d_2, \ldots, d_w)$ that satisfy the following constraints.

$$d = \sum_{j=1}^{w} d_j. \tag{20}$$

Step 4.   Compute the required raw material input vector $\hat{\boldsymbol{I}}$ and product flow vector $\boldsymbol{F}$ for each feasible demand pattern $\boldsymbol{d}$ obtained in Step 3 using the following equations:

$$I_j = d_j / E_{1,2}^j \quad \text{for } j = 1, 2, \ldots, w. \tag{21}$$

$$f_{j,i} = I_j \sum_{\eta \in \Phi_{j,i}} t_{1,\eta}^j, \quad \text{for } j = 1, 2, \ldots, w \text{ and } i = 1, 2, \ldots, m. \tag{22}$$

Step 5.   Reserve the feasible flow vectors satisfying the following constraint.

$$f_{j,i} \leq y_{j,i} h_{x_{j,i},i}, \quad \text{for } j = 1, 2, \ldots, w \text{ and } i = 1, 2, \ldots, m. \tag{23}$$

Step 6.   Use Eq. (24) to transform $\boldsymbol{F} \in \mathbf{F}$ into a d-MCCPs

$$s_{j,i} = \omega h_{x_{j,i},i} \quad \text{if } (\omega - 1)h_{x_{j,i},i} < f_{j,i} \leq \omega h_{x_{j,i},i} \text{ with } \omega \in \{0, 1, 2, \ldots, y_{j,i}\}$$
$$\text{for } j = 1, 2, \ldots, w \text{ and } i = 1, 2, \ldots, m. \tag{24}$$

Output: All d-MCCPs.

3.2 RSDP for production reliability computation

Let $\mathbf{\Omega} = \{\boldsymbol{S}_1, \boldsymbol{S}_1, \ldots, \boldsymbol{S}_\kappa\}$ be the set of all d-MCCPs for configuration $(\boldsymbol{X}, \boldsymbol{Y})$. The production reliability is reformulated using Eq. (25).

$$R(d, (X, Y)) = \Pr\left\{\bigcup_{i=1}^{\kappa} \{\boldsymbol{S}|\boldsymbol{S} \geq \boldsymbol{S}_i\}\right\}, \tag{25}$$

Such a union probability can be computed using the inclusion–exclusion principle (Hudson & Kapur, 1985; Lin et al., 1995), disjoint-event method (Hudson & Kapur, 1985; Yarlagadda & Hershey, 1991), state-space decomposition (Alexopoulos, 1995; Aven, 1985), and RSDP (Bai et al., 2015; Zuo et al., 2007). The RSDP is proposed based on the sum of disjoint products and has been validated as being more efficient than the other techniques for larger networks (Zuo et al., 2007). Bai et al. (2015) later proposed ordering heuristics to improve the efficiency. Hence, our study applies the improved RSDP to calculate the union probability by inputting the d-MCCPs generated by the AMC-based approach, machine reliability $r_{x_{j,i},i}$, and production capacity $h_{x_{j,i},i}$. Details of the RSDP procedure can be found in Bai et al. (2015).

4 NSGA-II to search for Pareto solutions

In the case of multiple conflicting objectives, a single solution may not exist with the optimal objectives. Therefore, a tradeoff solution is required. The NSGA-II is proposed by Deb et al. (2002) and is appropriate for bi-objective optimization (Chambari et al., 2021; Lin et al., 2019). The NSGA-II uses a nondominated sorting method and crowding distance to rank the chromosomes in the population, and then adopts the evolution process to obtain a Pareto set.

The following procedure illustrates the implementation of the NSGA-II to solve the problem, followed by subsections explaining its modules.

NSGA-II

Step 1. Generate an initial population, in which the representation of machine configuration $(\boldsymbol{X}, \boldsymbol{Y})$ is extended to express a chromosome (see Sect. 4.1).

Step 2. For each chromosome $(\boldsymbol{X}, \boldsymbol{Y})$, evaluate $\boldsymbol{R}(d, (\boldsymbol{X}, \boldsymbol{Y}))$ using the AMC-based approach and RSDP (see Sects. 3.1 and 3.2, respectively), and compute $C(\boldsymbol{X}, \boldsymbol{Y}) = \sum_{j=1}^{w} \sum_{i=1}^{m} c_{x_{j,i},i} y_{j,i}$.

Step 3. Rank the chromosomes in the population.

    3.1. The nondominated sorting method is used to rank the chromosomes in the population (see Sect. 4.2).

    3.2. The crowding distance of each chromosome is calculated to represent its density (see Sect. 4.3).

Step 4. Implement the evolution process (see Sect. 4.4).

    4.1. Parents selection.
    4.2. Parents crossover.
    4.3. Offspring mutation.
    4.4. Repeat Steps 4.1–4.3 to generate enough offspring.

Step 5. Update population (see Sect. 4.5).

Step 6. Go to Step 4 if the terminal condition is not met; otherwise, output the Pareto set.

4.1 Population initiation

The generation of a chromosome generates a pair consisting T and O. According to the setting of the population size (P_{size}), the following equations must be repeated P_{size} times to initialize the population:

$$x_{j,i} = \text{Rand}(1, q_i) \quad \text{for } j = 1, 2, \ldots, w \text{ and } i = 1, 2, \ldots, m, \text{ and} \quad (26)$$

$$y_{j,i} = \begin{cases} Rand(1, M_i^{x_{j,i}}) & \text{if } j = 1 \\ Rand\left(1, M_i^{x_{j,i}} - \sum_{a=1}^{j-1} y_{a,i}\right) & \text{if } j > 1 \end{cases} \quad \text{for } j = 1, 2, \ldots, w \text{ and } i = 1, 2, \ldots, m,$$

$$(27)$$

$Rand(\cdot)$ is a random function for selecting a random integer between the given intervals. Equation (26) randomly selects a supplier for each workstation. In addition, according to Eqs. (26), (27) then determines the number of configured machines limited to the remaining quantity of each selected supplier.

4.2 Nondominated sorting

The nondominated sorting method ranks all the chromosomes in a population based on non-domination. Let $\Lambda = \{1, 2, \ldots, P_{size}\}$ be a set of chromosome indices. For convenience, $R(\lambda)$ and $C(\lambda)$ represent the production reliability and total purchase cost of chromosome λ, respectively. The nondominated sorting method is described by the following pseudocode:

Input: All chromosomes in the population and offspring

For each chromosome $\lambda \in \Lambda$.

    $D_\lambda = \varnothing$ //store the chromosomes dominated by λ

    $N_\lambda = 0$ //the number of chromosomes which dominate λ

    $\Psi_\rho = \varnothing$ //store the rank-ρ chromosomes

    For chromosome $\theta \in \Lambda$ and $\lambda \neq \theta$

        If $R(\lambda) > R(\theta)$ and $C(\lambda) \leq C(\theta)$ //if λ dominates θ

            $D_\lambda = D_\lambda \cup \{\theta\}$

        End If

        If $R(\lambda) = R(\theta)$ and $C(\lambda) < C(\theta)$ //if λ dominates θ

            $D_\lambda = D_\lambda \cup \{\theta\}$

        End If

        If $R(\lambda) < R(\theta)$ and $C(\lambda) \geq C(\theta)$ // if θ dominates λ

            $N_\lambda = N_\lambda + 1$

        End If

        If $R(\lambda) = R(\theta)$ and $C(\lambda) > C(\theta)$ // if θ dominates λ

            $N_\lambda = N_\lambda + 1$

        End If

    End For

    If $N_\lambda = 0$ //store Pareto solution

        $Rank_\lambda = 1$ // set the rank of chromosome λ as level 1

        $\Psi_1 = \Psi_1 \cup \{\lambda\}$ //store the Pareto solution

    End If

End For

$\rho = 1$

While $\Psi_\rho \neq \varnothing$

    $\Xi = \varnothing$ //store the individuals of the next rank

    For each $\lambda \in \Psi_\rho$

        For each $\theta \in D_\lambda$

            $N_\theta = N_\theta - 1$

            If $N_\theta = 0$ //chromosome θ belongs to the next rank

                $Rank_\theta = \rho + 1$

                $\Xi = \Xi \cup \{\theta\}$

            End If

        End For

    End For

    $\rho = \rho + 1$

    $\Psi_\rho = \Xi$

End While

Output: ranks of all chromosomes

Nondominated sorting method

The four if–then rules within the second for-loop ensure that the chromosome with $R(\lambda) = 0$ has a worse rank than that of the feasible ones, and all the infeasible chromosomes are ranked according to the total purchase cost. After comparing all the chromosomes, the chromosomes dominated by fewer chromosomes are assigned better ranks. The output of the procedure is the rank of all the chromosomes.

4.3 Density evaluation

The crowding distance measures the density relationship between the chromosomes of same rank. Identically ranked chromosomes are sorted according to the bth objective for $b = 1$ and 2. Subsequently, the crowding distance of the λth chromosome, denoted by CD_λ is evaluated as:

$$CD_\lambda = \sum_{b=1}^{2} \Delta_{\lambda,b}, \tag{28}$$

where $\Delta_{\lambda,b}$ is the distance of the λth chromosome with respect to bth objective, and is calculated as:

$$\Delta_{\lambda,b} = \begin{cases} \frac{K_{\lambda+1,b} - K_{\lambda-1,b}}{K_b^{\max} - K_b^{\min}} & \text{if } K_{\lambda,b} \neq K_b^{\max} \text{ or } K_{\lambda,b} \neq K_b^{\min}; \\ \infty & \text{otherwise.} \end{cases} \tag{29}$$

where $K_{\lambda,b}$ is the bth objective value of the λth chromosome, and $K_{\lambda+1,b}$ and $K_{\lambda\text{-}1,b}$ represent the neighbors' bth objective values of the λth chromosome. The maximal and minimal values of the bth objective in the chromosomes are denoted by $K_b^{\max}$ and $K_b^{\min}$, respectively.

4.4 Evolution process

For selection, the NSGA-II randomly selects two chromosomes, λ and θ. Subsequently, the following crowded comparison rule is employed to select the best chromosome according to the rank and crowded distance attributes:

Crowded comparison rule Chromosome λ performs better than Chromosome θ if ($Rank_\lambda = Rank_\theta$ and $CD_\lambda > CD_\theta$) or ($Rank_\lambda < Rank_\theta$).

Rank$_\lambda$ indicates the rank of the λth chromosome in the population. This selection process maintains the species diversity.

The selection operator is repeated to select two chromosomes as parents. Based on the crossover probability C_{prob}, parents may generate offspring with a uniform crossover, which is a commonly used and well-convergent operator (Lim et al., 2017). Bortolini et al. (2022) demonstrated that a uniform crossover can perform well in a manufacturing reconfiguration problem. Figure 3 shows an example of implementing a uniform crossover. It generates a $w \times m$ binary mask first, and then the gene sequences of the parents' X and Y are reversed according to the positions of the elements, with a value of 1 in the mask. If Constraint (6) is not fulfilled, the offspring must be repaired to be noteworthy. For instance, workstations $n_{1,4}$ and $n_{2,4}$ choose the third supplier in offspring B. Then, the quantity of the third supplier in the fourth process is five (i.e., $q_3 = 5$). Because $y_{1,4} = y_{2,4} = 6 > 5$, $y_{2,4}$ must be changed to an arbitrary integer in the interval $[1, 5 - y_{1,4}]$.

The mutation operator may further mutate the offspring based on the mutation probability M_{prob}. Figure 4 shows an example of this mutation. It also generates a $w \times m$ binary mask. If a gene in the mask equals 1, the corresponding gene in X is mutated by randomly reselecting another supplier. The corresponding position's gene in Y is then randomly assigned a value
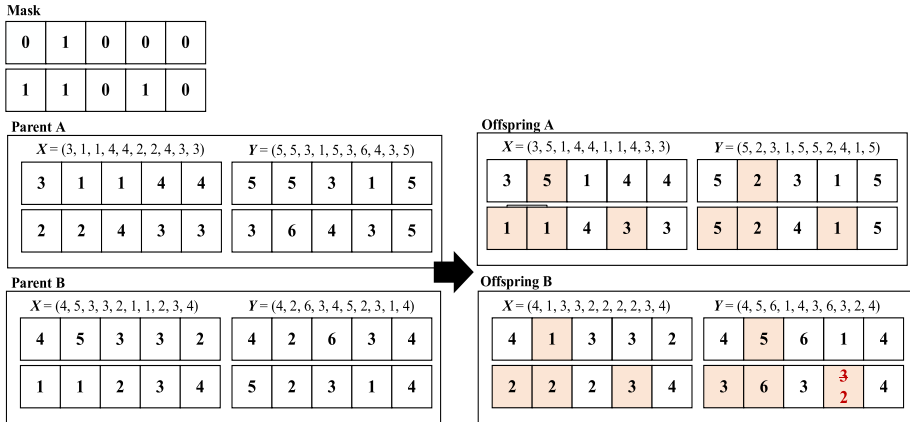
Mask

0	1	0	0	0
1	1	0	1	0

Parent A

$X = (3, 1, 1, 4, 4, 2, 2, 4, 3, 3)$ $Y = (5, 5, 3, 1, 5, 3, 6, 4, 3, 5)$

3	1	1	4	4	5	5	3	1	5
2	2	4	3	3	3	6	4	3	5

Parent B

$X = (4, 5, 3, 3, 2, 1, 1, 2, 3, 4)$ $Y = (4, 2, 6, 3, 4, 5, 2, 3, 1, 4)$

4	5	3	3	2	4	2	6	3	4
1	1	2	3	4	5	2	3	1	4

Offspring A

$X = (3, 5, 1, 4, 4, 1, 1, 4, 3, 3)$ $Y = (5, 2, 3, 1, 5, 5, 2, 4, 1, 5)$

3	5	1	4	4	5	2	3	1	5
1	1	4	3	3	5	2	4	1	5

Offspring B

$X = (4, 1, 3, 3, 2, 2, 2, 2, 3, 4)$ $Y = (4, 5, 6, 1, 4, 3, 6, 3, 2, 4)$

4	1	3	3	2	4	5	6	1	4
2	2	2	3	4	3	6	3	3/2	4

Fig. 3 Crossover representation

Mask

0	1	0	0	1
1	0	0	0	0

Chromosome

$X = (4, 1, 3, 3, 2, 2, 2, 2, 3, 4)$ $Y = (4, 5, 6, 1, 4, 3, 6, 3, 2, 4)$

4	1	3	3	2	4	5	6	1	4
2	2	2	3	4	3	6	3	2	4

Chromosome

$X = (4, 1, 3, 3, 2, 2, 2, 2, 3, 4)$ $Y = (4, 5, 6, 1, 4, 3, 6, 3, 2, 4)$

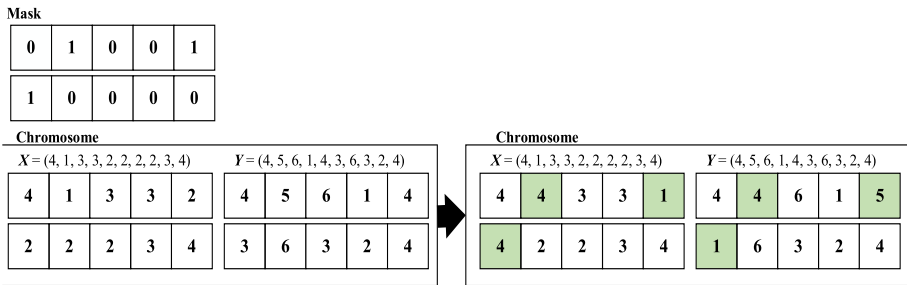| 4	4	3	3	1	4	4	6	1	5
4	2	2	3	4	1	6	3	2	4

Fig. 4 Mutation representation

that satisfies the available quantity of the supplier. The evolution process continues until a total of P_{size} new chromosomes are produced. Step 2 is then used to compute the production reliability and purchase cost of the new chromosomes.

4.5 Population update

After generating the new chromosomes, combining them with the current population results in $2 \times P_{size}$ chromosomes in the pool. Before updating the population to consist of P_{size} chromosomes, Step 3 is adopted to evaluate the rank and crowded distance of the $2 \times P_{size}$ chromosomes. Then, P_{size} chromosomes are selected from the pool according to their ranks. However, suppose that the number of selected chromosomes exceeds P_{size}, then the chromosomes with the worst rank and lower crowded distance are eliminated, such that the number of selected chromosomes, which comprise the next population, is P_{size}.

5 TOPSIS to determine the best compromise alternative

TOPSIS is a popular method for determining a compromise from a set of alternatives based on multiple criteria (Tzeng & Huang, 2011). It defines the positive and negative ideal alternatives. A positive ideal alternative can be identified as the combination of the highest maximal production reliability and lowest purchase cost. Conversely, the negative ideal alternative can be identified as a combination of the smallest production reliability and largest purchase cost. The best compromise alternative has the shortest geometric distance from the positive ideal alternative and longest distance from the negative ideal alternative. That is, TOPSIS can rank nondominated solutions based on their geometric distances. The following procedure uses TOPSIS to settle on a compromise alternative.

TOPSIS

Step 1. Construct a decision matrix $\mathbf{G} = [[g_{1,1}, g_{1,2}], [g_{2,1}, g_{2,2}], \ldots, [g_{\pi,1}, g_{\pi,2}]]$, where π is the number of Pareto solutions and $g_{\lambda,1}$ and $g_{\lambda,2}$ represent the production reliability and purchase cost of the λth Pareto solution, respectively.

Step 2. Use the following equation to obtained a standardized decision matrix $\hat{\mathbf{G}} = [[\hat{g}_{1,1}, \hat{g}_{1,2}], [\hat{g}_{2,1}, \hat{g}_{2,2}], \ldots, [\hat{g}_{\pi,1}, \hat{g}_{\pi,2}]]$.

$$\hat{g}_{\lambda,b} = \frac{g_{\lambda,b}}{\sqrt{\sum_{i=1}^{\pi} g_{i,b}^2}} \quad \text{for } l = 1, 2, \ldots, p \text{ and } b = 1, 2. \tag{30}$$

Step 3. Use the following equation to create a weighted decision matrix $\overline{\mathbf{G}} = [[\overline{g}_{1,1}, \overline{g}_{1,2}], [\overline{g}_{2,1}, \overline{g}_{2,2}], \ldots, [\overline{g}_{\pi,1}, \overline{g}_{\pi,2}]]$.

$$\overline{g}_{\lambda,b} = \hat{g}_{\lambda,b} \times \hat{w}_b \quad \text{for } \lambda = 1, 2, \ldots, \pi \text{ and } b = 1, 2, \tag{31}$$

where $\hat{w}_b$ is the weight of the bth objective. These weights can be determined either subjectively or objectively. Analytic Hierarchy Process and information entropy are popular objective weighting methods (Chen et al., 2020a; Yu et al., 2020).

Step 4. Determine the positive ideal alternative (g_1^+, g_2^+) and negative ideal alternative (g_1^-, g_2^-) as:

$$\left(g_1^+, g_2^+\right) = \left(\max\left(\overline{g}_{1,1}, \overline{g}_{2,1}, \ldots, \overline{g}_{\pi,1}\right), \min\left(\overline{g}_{1,2}, \overline{g}_{2,2}, \ldots, \overline{g}_{\pi,2}\right)\right), \text{ and} \tag{32}$$

$$\left(g_1^-, g_2^-\right) = \left(\min\left(\overline{g}_{1,1}, \overline{g}_{2,1}, \ldots, \overline{g}_{\pi,1}\right), \max\left(\overline{g}_{1,2}, \overline{g}_{2,2}, \ldots, \overline{g}_{\pi,2}\right)\right). \tag{33}$$

Step 5. Calculate the geometric distance of each alternative to the positive ideal alternative (g_1^+, g_2^+) using Eq. (34).

$$GD_\lambda^+ = \sqrt{\sum_{b=1}^{2} \left(\overline{g}_{\lambda,b} - g_b^+\right)^2} \quad \text{for } \lambda = 1, 2, \ldots, \pi. \tag{34}$$

Step 6. Calculate the geometric distance of each alternative to the negative ideal alternative (g_1^-, g_2^-) using Eq. (35).

$$GD_\lambda^- = \sqrt{\sum_{b=1}^{2} \left(\overline{g}_{\lambda,b} - g_b^-\right)^2} \quad \text{for } \lambda = 1, 2, \ldots, \pi. \tag{35}$$

Step 7. Calculate the relative proximity Ω_λ of each alternative using Eq. (36).

$$\Omega_l = \frac{GD_\lambda^-}{GD_\lambda^+ - GD_\lambda^-} \quad \text{for } \lambda = 1, 2, \ldots, \pi. \tag{36}$$

Step 8. Choose the alternative with maximum relative proximity as the compromise alternative.

6 Numerical experiments of solar cell manufacturing

Solar cell manufacturing typically follows a flow shop model with multiple parallel machines at each workstation performing the same process (Chen et al., 2020b). The production line for solar cell manufacturing involves eight steps: texturing, diffusion, phosphorus glass etching, anti-reflective coating, screen printing, fast firing, edge isolation, and testing. In addition, the production line includes three repair workstations for the second, third, and fourth processes. If defective WIP flows are produced at these workstations, they are returned to the first process for further processing (Raval & Reddy, 2019; Song & Lin, 2018). Figure 5 illustrates the FSS network for solar cell manufacturing, which comprises two production lines. Table 1 presents the technical specifications of the machines used in the process, which are sourced from the solar cell device suppliers. These parameters are based on historical usage experience, or can be obtained by requesting product information from the suppliers.

In this section, the applicability and efficiency of integrating the NSGA-II, AMC, RSDP, and TOPSIS are demonstrated using a solar cell manufacturing case. All algorithms are programmed in Python and executed on Windows 11, Intel Core i7-9700, CPU 3.00 GHz, and 16 GB RAM.

6.1 Pareto set determination and measurement

This study considers three demand scenarios: $d = 4,000$, $d = 5,000$, and $d = 6,000$. For each scenario, we perform ten trials using the NSGA-II and generate a reference set that includes all the nondominated solutions in the 10 Pareto sets. Based on parameter recommendations from bi-objective component assignment studies (Lin & Yeh, 2012; Lin et al., 2019; Yeh et al., 2023), we set the NSGA-II parameters to $(P_{size}, C_{prob}, M_{prob}) = (100, 0.6, 0.025)$. During each trial, we run the NSGA-II for 1500 s.

To evaluate the quality of the obtained Pareto set, we employ three different metrics: the number of solutions in the reference set (NSR), ratio of nondominated individuals (RNI), and generational distance (GD) (Yeh, 2019; Yen & He, 2013). Let $\mathbf{P_{set}}$ be a nondominated
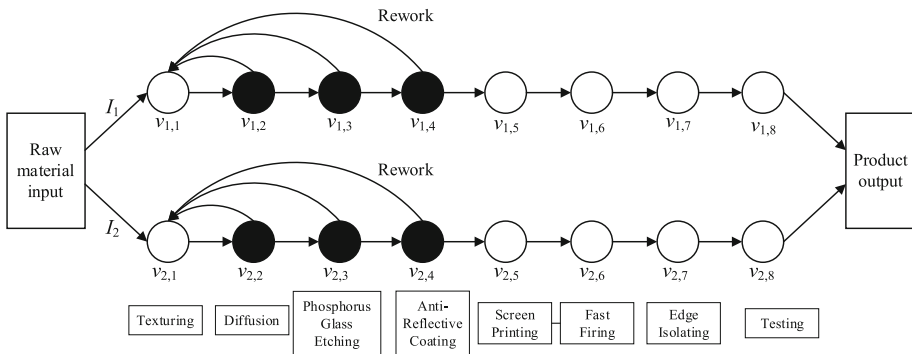

Fig. 5 The network model of the solar cell manufacturing system

Table 1 The machine data of the solar cell device suppliers

Workstation	Supplier #	Capacity (pcs/per hour)	Yield rate	Operation probability	Machine cost (10,000 NTD)	Machine quantity
$n_{1,1}/n_{2,1}$	1	2000	0.983	0.9854	122	5
	2	5000	0.978	0.9890	280	2
	3	3600	0.974	0.9872	200	4
	4	2200	0.985	0.9853	183	5
	5	2400	0.979	0.9820	186	6
$n_{1,2}/n_{2,2}$	1	1600	0.974	0.9884	153	7
	2	2000	0.978	0.9795	165	7
	3	2000	0.969	0.9910	180	5
	4	1000	0.975	0.9921	125	8
$n_{1,3}/n_{2,3}$	1	4000	0.968	0.9852	182	3
	2	3600	0.973	0.9830	176	3
	3	4000	0.976	0.9862	202	3
$n_{1,4}/n_{2,4}$	1	3600	0.977	0.9923	92	4
	2	4800	0.979	0.9894	126	3
$n_{1,5}/n_{2,5}$	1	3000	0.994	0.9915	107	3
	2	2800	0.998	0.9796	103	5
	3	1400	0.993	0.9890	81	6
	4	600	0.989	0.9920	62	10
	5	990	0.988	0.9888	65	8
$n_{1,6}/n_{2,6}$	1	5300	0.986	0.9784	144	2
	2	3000	0.988	0.9835	113	4
	3	2100	0.984	0.9822	106	4
$n_{1,7}/n_{2,7}$	1	3600	0.986	0.9654	131	3
	2	900	0.984	0.9788	75	7
	3	1500	0.98	0.9817	112	6
$n_{1,8}/n_{2,8}$	1	1500	0.998	0.9863	67	7
	2	2500	0.992	0.9830	92	5
	3	1000	0.995	0.9745	61	8
	4	3000	0.993	0.9780	106	6

set and $\mathbf{R}_{\text{set}}$ be a reference set, where all $\varepsilon_i \in \mathbf{P}_{\text{set}}$ and $\gamma_j \in \mathbf{R}_{\text{set}}$ are the normalized elements. These three metrics are defined as:

$$\text{NSR} = |\text{Pareto set} \cap \text{reference set}|, \tag{37}$$

$$\text{RNI} = \frac{|\text{Pareto set} \cap \text{reference set}|}{|\text{Pareto set}|}, \text{ and} \tag{38}$$

$$\text{GD} = \sum_{i=1}^{P_{size}} \sum_{i=1}^{R_{size}} (\varepsilon_i - \gamma_j)^2. \tag{39}$$

A smaller GD, higher NSR, and higher RNI indicate a superior Pareto set. If all solutions in the nondominated set are members of the reference set, GD is equal to 0 and RNI is equal to 1.

Table 2 lists the result of each demand scenario with the three metrics. For each scenario, the best trail, whose Pareto set has a lower GD, higher RNI, and NSR, has been marked bold. In particular, these best trails have significantly better GA and RNI than those of the average GD and RNI. Therefore, they can be used for further decision-making.

6.2 Compromise alternative selection

The TOPSIS can determine a compromise alternative from a set of solutions. Because the compromise alternative involves two criteria–production reliability and total purchase cost, it is necessary to set weights for both the criteria before executing TOPSIS. Suppose a system administrator subjectively considers three weight settings: $(\hat{w}_1, \hat{w}_2) = (0.8, 0.2)$ (reliability preference), $(\hat{w}_1, \hat{w}_2) = (0.2, 0.8)$ (cost preference), and $(\hat{w}_1, \hat{w}_2) = (0.5, 0.5)$ (no preference). According to the reference set for each demand scenario, as described in Sect. 6.1, we use the TOPSIS to determine the compromise solution for each weight setting. The compromise alternatives and corresponding machine configurations are listed in Table 3. Figures 6, 7 and 8 show the scatter plots of all the points in the three reference sets and highlight the compromise alternatives for all the weight settings. This information can support the system administrators' decision-making. For example, if the system administrator expects that the solar cell FSS has a production reliability of more than 0.98 and approximate budget of 4, 000 NTD (unit: ten thousand NTD), the machine configuration $(X, Y) = ((1, 3, 1, 1, 1, 2, 2, 2, 1, 3, 2, 1, 2, 1, 1, 1), (1, 1, 1, 1, 1, 1, 3, 1, 4, 3, 2, 2, 3, 2, 3, 5))$ can be considered for the FSS design. In addition, as shown in Fig. 6, the compromise alternatives are more inclined to lower costs under the three weight settings because each alternative has a higher production reliability and less difference in production reliability. System administrators can make decisions by focusing on purchase cost criteria.

6.3 Efficiency validation for NSGA-II

This subsection focuses on validating the efficiency of the NSGA-II. SPEA2 is an efficient multi-objective genetic algorithm (GA) for optimization problems (Zitzler et al., 2001). The SPEA2 is also a population-based multi-objective GA, similar to the NSGA-II, which adapts significantly to our crossover and mutation operators (Biswas & Pal, 2021). In addition, the SEAP2 has a domination-based framework similar to that of the NSGA-II, and thus can maintain a great spread of solutions and converge well to the exact Pareto optimal front (Ma et al., 2023). Thus, it is considered comparable to the NSGA-II. Under the same parameter setting $(P_{isze}, C_{prob}, M_{prob}) = (100, 0.6, 0.025)$ and terminal condition of 1500-s runtime, the SPEA2 is also performed for ten trials for the three scenarios given in Sect. 6.1. For each scenario, the Pareto set of the best NSGA-II trail is combined with one of the best SPEA2 trail to form a reference set. Then both methods' GD, RNI, and NSR are calculated according to the reference set and Eq. (37)–(39). Table 4 lists the reference set and three metrics for both the methods. The SPEA2 can determine more nondominated solutions, such that it has a better NSR. However, the NSGA-II performs better in terms of both the GD and NRI. Under the same terminal conditions, the nondominated solutions from the NSGA-II usually belong to the reference set. Therefore, the Pareto set from the NSGA-II is more informative in this case.

Table 2 The experimental results from the NSGA-II

Demand		$d = 4000$	$d = 5000$	$d = 6000$
Analysis of 10 trials	Size of reference set	28	19	30
	Average GD	0.095634434	0.055931389	0.046817421
	Average RNI	0.258739316	0.243981019	0.369102564
Trial 1	Size of Pareto set	**10**	11	6
	GD	**0.007512**	0.036493	0.045215
	NRS	**7**	1	3
	RNI	**0.7**	0.090909	0.5
Trial 2	Size of Pareto set	9	7	9
	GD	0.146881	0.062899	0.009053
	NRS	0	2	6
	RNI	0	0.285714	0.666667
Trial 3	Size of Pareto set	7	6	8
	GD	0.10754	0.064532	0.027664
	NRS	0	0	6
	RNI	0	0	0.75
Trial 4	Size of Pareto set	10	9	13
	GD	0.047459	0.13795	0.053374
	NRS	1	0	4
	RNI	0.1	0	0.307692
Trial 5	Size of Pareto set	9	7	8
	GD	0.125693	0.111476	0.048443
	NRS	3	0	1
	RNI	0.333333	0	0.125
Trial 6	Size of Pareto set	12	8	11
	GD	0.053699	0.03084	0.083051
	NRS	4	2	0
	RNI	0.333333	0.25	0
Trial 7	Size of Pareto set	8	7	15
	GD	0.145574	0.027073	0.052101
	NRS	4	3	2
	RNI	0.125	0.428571	0.133333
Trial 8	Size of Pareto set	13	**6**	8
	GD	0.125198	**0**	0.077082
	NRS	5	**6**	3
	RNI	0.384615	**1**	0.375
Trial 9	Size of Pareto set	9	8	**6**
	GD	0.060851	0.05253	**0.00342**
	NRS	1	0	**5**

Table 2 (continued)

Demand		$d = 4000$	$d = 5000$	$d = 6000$
	RNI	0.111111	0	**0.833333**
Trial 10	Size of Pareto set	12	13	7
	GD	0.135937	0.035521	0.06877
	NRS	6	5	0
	RNI	0.5	0.384615	0

*Numbers in bold indicate the best trial for the corresponding scenario

Table 3 The compromise alternatives determined by TOPSIS

| Demand	Weight	Compromise alternative		Corresponding machine configuration
		Production reliability	Total purchase cost	
$d = 4000$	(0.2, 0.8)	0.979044345	3420	$X = (1,1,1,1,1,1,1,1,1,1,1,1,1,3,3,1)$ $Y = (3,4,2,2,2,2,3,4,1,1,1,1,2,1,1)$
	(0.5, 0.5)	0.979044345	3420	$X = (1,1,1,1,1,1,1,1,1,1,1,1,1,3,3,1)$ $Y = (3,4,2,2,2,2,3,4,1,1,1,1,2,1,1)$
	(0.8, 0.2)	0.98650787	3583	$X = (1,1,1,1,1,1,1,1,1,1,1,1,1,3,3,1)$ $Y = (3,3,2,2,2,2,3,4,1,1,1,1,2,1,1)$
$d = 5000$	(0.2, 0.8)	0.853830563	3478	$X = (1,3,2,1,1,2,1,2,1,3,1,1,1,2,1,4)$ $Y = (3,2,2,2,2,2,2,2,2,2,2,2,2,2,2)$
	(0.5, 0.5)	0.951245016	3705	$X = (1,3,1,1,1,2,1,4,3,3,1,1,1,2,1,2)$ $Y = (3,3,2,2,2,2,2,3,4,2,1,1,1,2,2,2)$
	(0.8, 0.2)	0.990810732	4002	$X = (1,3,1,1,1,2,2,2,1,3,2,1,2,1,1,1)$ $Y = (1,1,1,1,1,1,3,1,4,3,2,2,3,2,3,5)$
$d = 6000$	(0.2, 0.8)	0.919476822	3919	$X = (2,2,2,1,1,1,1,2,1,1,1,1,1,3,3,1)$ $Y = (1,3,2,2,2,2,2,3,2,2,1,1,1,2,2,3)$
	(0.5, 0.5)	0.934157673	4014	$X = (2,3,2,1,2,1,3,2,2,1,1,1,5,3,2,1)$ $Y = (1,3,2,2,3,2,3,3,1,2,1,1,3,2,2,3)$
	(0.8, 0.2)	0.973047234	4296	$X = (2,3,1,1,1,1,1,2,3,1,1,1,1,2,1,1)$ $Y = (1,4,2,2,3,2,2,4,1,2,1,1,1,2,2,2)$
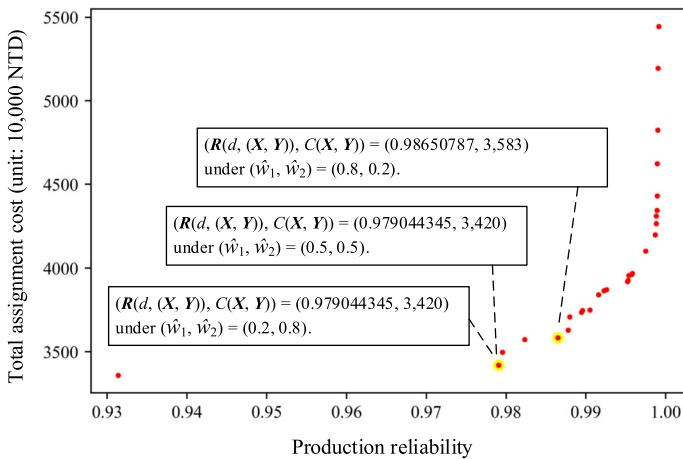
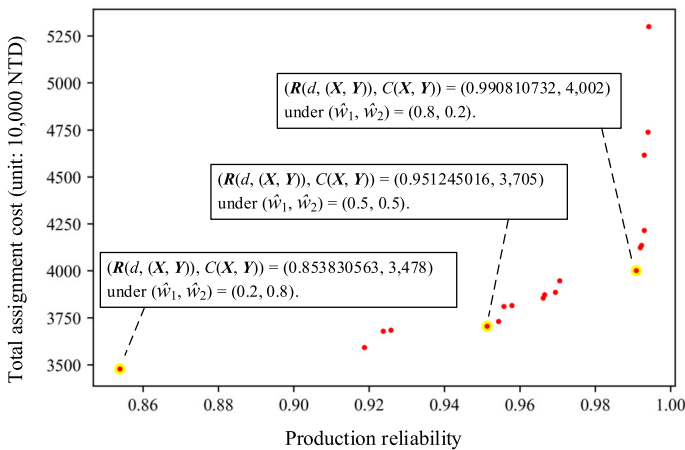**Fig. 6** The compromise alternative selection for $d = 4000$

Fig. 7 The compromise alternative selection for $d = 5000$

7 Conclusions

This study proposes a hybrid method that integrates the NSGA-II, AMC, RSDP, and TOP-SIS to determine the reliability-cost-oriented machine configuration for an FSS with repair actions. The problem addressed involves the evaluation of production reliability. The AMC-based approach uses a transition matrix to calculate the required raw material input and product flow traveling through each workstation according to the order demand d without enumerating all the processing paths, as in the path decomposition method proposed by Lin and Chang (2015). Subsequently, using this approach the d-MCCPs in terms of these flows are determined. The RSDP then calculates the union probability of the d-MCCPs to acquire the production reliability.
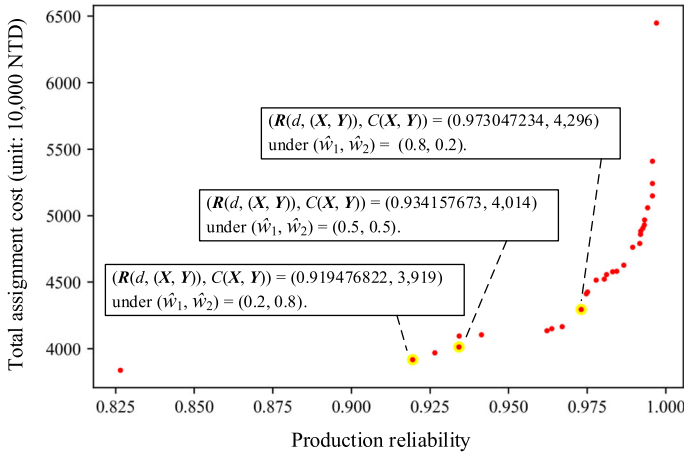
Fig. 8 The compromise alternative selection for $d = 6000$

Table 4 Comparisons of the NSGA-II and the SPEA2

Demand	$d = 4000$		$d = 5000$		$d = 6000$	
	NSGA-II	SPEA2	NSGA-II	SPEA2	NSGA-II	SPEA2
Size of reference set	27		13		15	
Size of Pareto set	10	22	6	11	6	14
GD	0.004331	0.016985	0	0.022563	0.016999	0.041453
NRS	9	18	6	7	5	10
RNI	0.9	0.818182	1	0.636364	0.833333	0.714286

The proposed method adopts NSGA-II to identify the Pareto solutions that balance the production reliability and total purchase cost. To achieve this, we integrate the AMC and RSDP into the NSGA-II for production reliability calculation. TOPSIS is then employed to select a compromise alternative based on the given weights. In Sect. 6, we elaborate the applicability of integrating the NSGA-II and TOPSIS. The information derived from the hybrid method could support decision making regarding FSS design. Additionally, our experimental results demonstrate that the NSGA-II is computationally more efficient than the SPEA2, especially under the terminal condition of a 1500-s runtime. Overall, this study makes the following contributions:

(1) The tradeoff between the production reliability and purchase cost in the machine configuration of an FSS with repair actions is considered.
(2) A mathematical model of the addressed bi-objective problem is built, in which the FSS with repair actions is modeled as a typical multistate flow network.
(3) An alternate way is proposed to evaluate production reliability by combining the AMC with RSDP.
(4) A hybrid method that integrates the AMC, RSDP, NSGA-II, and TOPSIS to solve this problem is built.

(5) The applicability of our hybrid method using the solar cell FSS is illustrated.

(6) The superiority of NSGA-II over SPEA2 is demonstrated by comparing their efficiency.

In certain FSSs, the product flow can undergo multiple repairs in different production lines. However, the proposed AMC-based approach is only applicable when the product flow can be repaired at the same workstation. Future research could expand on this problem by considering scenarios where the product flow can undergo multiple repairs at other production lines and then explore the impact of this situation on the production reliability.

Declarations

Conflict of interest The authors declare that they have no conflict of interest.

Ethical approval This article does not contain any studies with human participants or animals performed by any of the authors.

Appendix

A.1. Abbreviations

FSS	Flow shop system
MFSN	Multistate flow shop network
d-MCCP	Minimal-consumed capacity patterns for demand d
WIP	Work-in-process
RSDP	Recursive Sum of Disjoint Products
AMC	Absorptive markov chain
NSGA-II	Nondominated sorting genetic algorithm ii
SPEA2	Improved Strength Pareto Evolutionary Algorithm
TOPSIS	Technique for Order Preference by Similarity to an Ideal Solution
NSR	Number of solutions in the reference set
RNI	Ratio of nondominated individuals
GD	Generational distance

A.2. Notations

w	Number of production lines
m	Number of workstations (processes) in each production line
L_j	Production line #j
$n_{j,i}$	ith workstation in production line L_j
N	$\{n_{j,i} \mid j = 1, 2, ..., w,\ i = 1, 2, ..., m\}$: set of m nodes (workstations), where v_1 represents a source station and v_m represents a sink station
A	Set of arcs (transport devices)
(N, A)	FSS network
z	Number of repairing workstations in each production line
I_j	Raw material input of production line L_j
O_j	Product output of production line L_j
$\hat{I}$	$(I_1, I_2, ..., I_w)$: input raw material vector
β_e	Index of the eth repairing workstation in a production line, where $e \in \{1, 2, ..., z\}$
n_{j,β_e}	eth repairing workstation in production line L_j
α_e	Index of the initial workstation corresponding to the repairing workstation n_{j,β_e}
n_{j,α_e}	Initial workstation corresponding to the repairing workstation n_{j,β_e}
q_i	Number of equipment suppliers for the ith process
$c_{u,i}$	Assignment cost of the uth supplier for the ith process
$r_{u,i}$	Machine reliability of the uth supplier for the ith process
$h_{u,i}$	Production capacity of the uth supplier for the ith process
$p_{u,i}$	Yield rate of the uth supplier for the ith process
$x_{j,i}$	Index of the selected supplier for the workstation $n_{j,i}$
X	$(x_{1,1}, x_{2,1}, ..., x_{1,m}, x_{2,1}, x_{2,2}, ..., x_{2,m}, ..., x_{w,1}, x_{w,2}, ..., x_{w,m})$:
$y_{j,i}$	Number of machines provided by the supplier $x_{j,i}$
Y	$(y_{1,1}, y_{2,1}, ..., y_{1,m}, y_{2,1}, y_{2,2}, ..., y_{2,m}, ..., y_{w,1}, y_{w,2}, ..., y_{w,m})$
(X, Y)	Machine assignment (chromosome)
$M_i^{x_{j,i}}$	Number of machines provided by the supplier $x_{j,i}$
$C(\mathbf{X}, \mathbf{Y})$	Total cost of machine assignment
$s_{j,i}$	Current production capacity of workstation
S	Current production capacity vector
$\mathbf{S}_{(\mathbf{X}, \mathbf{Y}), d}$	Set of production vectors meeting demand d under machine assignment $(\mathbf{X}, \mathbf{Y})$

$\boldsymbol{R}(d, (\boldsymbol{X}, \boldsymbol{Y}))$	Production reliability
$\mathbf{B}_j$	AMC transition matrix of production line L_j
$\mathbf{U}_j$	Transient-to-transient status matrix of production line L_j
$\mathbf{V}_j$	Transient-to-absorbing status matrix of production line L_j
$\mathbf{0}$	Zero matrix
$\mathbf{I}$	Identity matrix
$\Gamma_{j,e}$	$\{n_{j,\alpha_e}, n_{j,\alpha_e+1}, \ldots, n_{j,\beta_e-1}, n_{j,\beta_e}\}$: set of sequential workstations for the repairing process from n_{j,α_e} to n_{j,β_e}
$\mathbf{T}_j$	Expected value matrix of production line L_j
$\mathbf{E}_j$	Absorbing probability matrix of production line L_j
$f_{j,i}$	Product flow processed by the workstation $n_{j,i}$
$\boldsymbol{F}$	$(f_{1,1}, f_{1,2}, \ldots, f_{1,m}, f_{2,1}, f_{2,2}, \ldots, f_{w,m})$: flow vector
d_j	Demand amount assigned to production line L_j
$\boldsymbol{d}$	$(d_1, d_2, \ldots, d_m)$: order demand vector
$\boldsymbol{\Phi}_{j,i}$	Set of the statuses related to workstation $n_{j,i}$
$\mathbf{F}_{(\boldsymbol{X}, \boldsymbol{Y}), d}$	Set of feasible flow vectors satisfying the maximal capacity under demand d and machine assignment $(\boldsymbol{X}, \boldsymbol{Y})$
$\boldsymbol{\Omega}_{(\boldsymbol{X}, \boldsymbol{Y})}$	Set of all d-mccps associated with $(\boldsymbol{X}, \boldsymbol{Y})$
P_{size}	Population size
$\boldsymbol{\Lambda}$	Set of chromosome indices in order
$R(\lambda)$	Production reliability of chromosome λ, $\lambda \in \boldsymbol{\Lambda}$
$C(\lambda)$	Total assignment cost of chromosome λ, $\lambda \in \boldsymbol{\Lambda}$
D_λ	Index set of the chromosomes dominated by chromosome λ, $\lambda \in \boldsymbol{\Lambda}$
N_λ	Number of the chromosomes dominating chromosome λ, $\lambda \in \boldsymbol{\Lambda}$
Ψ_ρ	Set of the chromosomes with rink ρ
CD_λ	Crowded distance of chromosome λ, $\lambda \in \boldsymbol{\Lambda}$
$Rank_\lambda$	Rank of chromosome λ, $\lambda \in \boldsymbol{\Lambda}$
$\Delta_{\lambda,b}$	Distance of chromosome λ regarding to the bth objective
C_{prob}	Crossover probability
M_{prob}	Mutation probability
$k_{\lambda,1}$	Production reliability of the λth Pareto solution
$k_{\lambda,2}$	Assignment cost of the λth Pareto solution
π	Number of Pareto solutions
$\mathbf{G}$	Decision matrix
$\hat{\mathbf{G}}$	Standardized decision matrix
$\overline{\boldsymbol{G}}$	Weighted decision matrix

$\hat{w}_b$	Weight of the bth objective
(g_1^+, g_2^+)	Positive ideal alternative
(g_1^-, g_2^-)	Negative ideal alternative
GD_λ^+	Geometric distance of alternative λ to the positive ideal alternative
GD_λ^-	Geometric distance of alternative λ to the negative ideal alternative
Ω_λ	Relative proximity of alternative λ

References

Alexopoulos, C. (1995). A note on state-space decomposition methods for analyzing stochastic flow networks. *IEEE Transactions on Reliability, 44*(2), 354–357.

Aven, T. (1985). Reliability evaluation of multistate systems with component. *IEEE Transactions on Reliability, 34*(5), 473–479.

Bai, G. H., Zuo, M. J., & Tian, Z. G. (2015). Ordering heuristics for reliability evaluation of multistate networks. *IEEE Transactions on Reliability, 64*(3), 1015–1023.

Bajestani, M. A., Rabbani, M., Rahimi-Vahed, A. R., & Khoshkhou, G. B. (2009). A multi-objective scatter search for a dynamic cell formation problem. *Computers and Operations Research, 36*(3), 777–794.

Biswas, A., & Pal, T. (2021). A comparison between metaheuristics for solving a capacitated fixed charge transportation problem with multiple objectives. *Expert Systems with Applications, 170*, 114491.

Bortolini, M., Cafarella, C., Ferrari, E., Galizia, F. G., & Gamberi, M. (2022). Reconfigurable manufacturing system design using a genetic algorithm. In *Sustainable design and manufacturing: Proceedings of the 8th International conference on sustainable design and manufacturing (KES-SDM 2021)* (pp. 130–139). Springer.

Bowling, S. R., Khasawneh, M. T., Kaewkuekool, S., & Cho, B. R. (2004). A Markovian approach to determining optimum process target levels for a multi-stage serial production system. *European Journal of Operational Research, 159*(3), 636–650.

Brammer, J., Lutz, B., & Neumann, D. (2022). Permutation flow shop scheduling with multiple lines and demand plans using reinforcement learning. *European Journal of Operational Research, 299*(1), 75–86.

Bukchin, J., & Tzur, M. (2000). Design of flexible assembly line to minimize equipment cost. *IIE Transactions, 32*(7), 585–598.

Bukchin, J., & Rubinovitz, J. (2003). A weighted approach for assembly line design with station paralleling and equipment selection. *IIE Transactions, 35*(1), 73–85.

Chambari, A., Azimi, P., & Najafi, A. A. (2021). A bi-objective simulation-based optimization algorithm for redundancy allocation problem in series-parallel systems. *Expert Systems with Applications, 173*, 114745.

Chan, F. T. S., Wong, T. C., & Chan, L. Y. (2005). A genetic algorithm-based approach to machine assignment problem. *International Journal of Production Research, 43*(12), 2451–2472.

Chehade, H., Dolgui, A., Dugardin, F., Makdessian, L., & Yalaoui, F. (2012). Multi-objective approach for production line equipment selection. *Management and Production Engineering Review, 3*(1), 4–17.

Chen, L., Cai, Y., & Liu, H. (2020a). Evaluation of intelligent transportation system projects under uncertainty: A hybrid TOPSIS-AHP-entropy method. *Journal of Intelligent Transportation Systems, 24*(5), 455–468.

Chen, T.-L., Cheng, C.-Y., & Chou, Y.-H. (2020b). Multi-objective genetic algorithm for energy-efficient hybrid flow shop scheduling with lot streaming. *Annals of Operations Research, 290*(1), 813–836.

Deb, K., & Jain, H. (2014). An evolutionary many-objective optimization algorithm using reference-point-based non-dominated sorting approach, Part I: Solving problems with box constraints. *IEEE Transactions on Evolutionary Computation, 18*(4), 577–601.

Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. A. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation, 6*(2), 182–197.

Dios, M., Fernandez-Viagas, V., & Framinan, J. M. (2018). Efficient heuristics for the hybrid flow shop scheduling problem with missing operations. *Computers & Industrial Engineering, 115*, 88–99.

Dolgui, A., Hashemi-Petroodi, S. E., Kovalev, S., & Kovalyov, M. Y. (2021). Profitability of a multi-model manufacturing line versus multiple dedicated lines. *International Journal of Production Economics, 236*, 108113.

Durak, İ, Arslan, H. M., & Özdemir, Y. (2022). Application of AHP–TOPSIS methods in technopark selection of technology companies: Turkish case. *Technology Analysis & Strategic Management, 34*(10), 1109–1123.

Hadjinicola, G. C. (2010). Manufacturing costs in serial production systems with rework. *Journal of the Operational Research Society, 61*(2), 342–351.

Hossain, M. S. J., & Sarker, B. R. (2016). Optimal locations of on-line and off-line rework stations in a serial production system. *International Journal of Production Research, 54*(12), 3603–3621.

Hudson, J. C., & Kapur, K. C. (1985). Reliability bounds for multistate systems with multistate components. *Operations Research, 33*(1), 153–160.

Hwang, C.-L., Lai, Y.-J., & Liu, T.-Y. (1993). A new approach for multiple objective decision making. *Computers and Operational Research, 20*(8), 889–899.

Jahromi, M. H. M. A., & Tavakkoli-Moghaddam, R. (2012). A novel 0–1 linear integer programming model for dynamic machine-tool selection and operation allocation in a flexible manufacturing system. *Journal of Manufacturing Systems, 31*(2), 224–231.

Juan, A. A., Faulin, J., Grasman, S. E., Rabe, M., & Figueira, G. (2015). A review of simheuristics: Extending metaheuristics to deal with stochastic combinatorial optimization problems. *Operations Research Perspectives, 2*, 62–72.

Li, S., Wang, H., Hu, S. J., Lin, Y. T., & Abell, J. A. (2011). Automatic generation of assembly system configuration with equipment selection for automotive battery manufacturing. *Journal of Manufacturing Systems, 30*(4), 188–195.

Lim, S. M., Sultan, A. B. M., Sulaiman, M. N., Mustapha, A., & Leong, K. Y. (2017). Crossover and mutation operators of genetic algorithms. *International Journal of Machine Learning and Computing, 7*(1), 9–12.

Lin, J.-S., Jane, C.-C., & Yuan, J. (1995). On reliability evaluation of a capacitated-flow network in terms of minimal pathsets. *Networks, 25*, 131–138.

Lin, Y.-K., & Chang, P.-C. (2012). Evaluate the system reliability for a manufacturing network with reworking actions. *Reliability Engineering & System Safety, 106*, 127–137.

Lin, Y.-K., & Chang, P.-C. (2013). Reliability-based performance indicator for a manufacturing network with multiple production lines in parallel. *Journal of Manufacturing Systems, 32*(1), 147–153.

Lin, Y.-K., & Chang, P.-C. (2015). Demand satisfaction and decision-making for a PCB manufacturing system with production lines in parallel. *International Journal of Production Research, 53*(11), 3193–3206.

Lin, Y.-K., Chang, P.-C., Yeng, L.C.-L., & Shih, P.-S. (2017). Reliability evaluation for an intermittent production system with stochastic number of normal machines. *Journal of Manufacturing Systems, 45*, 222–235.

Lin, Y.-K., Chang, P.-C., Yeng, L.C.-L., & Huang, S.-F. (2019). Bi-objective optimization for a multistate job-shop production network using NSGA-II and TOPSIS. *Journal of Manufacturing Systems, 52*, 43–54.

Lin, Y.-K., & Chen, S.-G. (2022). Reliability evaluation in terms of flow data mining for multistate networks. *Annals of Operations Research, 311*(1), 225–237.

Lin, Y.-K., & Yeh, C.-T. (2012). Multi-objective optimization for stochastic computer networks using NSGA-II and TOPSIS. *European Journal of Operational Research, 218*(3), 735–746.

Liu, D., Huang, Q., Yang, Y., Liu, D., & Wei, X. (2020). Bi-objective algorithm based on NSGA-II framework to optimize reservoirs operation. *Journal of Hydrology, 585*, 124830.

Ma, H., Zhang, Y., Sun, S., Liu, T., & Shan, Y. (2023). A comprehensive survey on NSGA-II for multi-objective optimization and applications. *Artificial Intelligence Review, 56*, 1–54.

Nguyen, T. P. (2022). Evaluation of network reliability for stochastic-flow air transportation network considering discounted fares from airlines. *Annals of Operations Research, 311*(1), 335–355.

Niroomand, S. (2021). Hybrid artificial electric field algorithm for assembly line balancing problem with equipment model selection possibility. *Knowledge-Based Systems, 219*, 106905.

Oesterle, J., Amodeo, L., & Yalaoui, F. (2019). A comparative study of multi-objective algorithms for the assembly line balancing and equipment selection problem under consideration of product design alternatives. *Journal of Intelligent Manufacturing, 30*(3), 1021–1046.

Pillai, V. M., & Chandrasekharan, M. (2008). An absorbing Markov chain model for production systems with rework and scrapping. *Computers & Industrial Engineering, 55*(3), 695–706.

Raval, M. C., & Reddy, S. M. (2019). *Industrial silicon solar cells*. IntechOpen.

Sarker, B. R., Jamal, A., & Mondal, S. (2008). Optimal batch sizing in a multi-stage production system with rework consideration. *European Journal of Operational Research, 184*(3), 915–929.

Saxena, L. K., & Jain, P. K. (2012). A model and optimisation approach for reconfigurable manufacturing system configuration design. *International Journal of Production Research, 50*(12), 3359–3381.

Silva, P. M., Gonçalves, J. N., Martins, T. M., Marques, L. C., Oliveira, M., Reis, M. I., Araújo, L., Correia, D., Telhada, J., Costa, L., & Fernandes, J. M. (2022). A hybrid bi-objective optimization approach for joint determination of safety stock and safety time buffers in multi-item single-stage industrial supply chains. *Computers & Industrial Engineering, 168*, 108095.

Song, W. T., & Lin, P. (2018). System reliability of stochastic networks with multiple reworks. *Reliability Engineering & System Safety, 169*, 258–268.

Tzeng, G.-H., & Huang, J.-J. (2011). *Multiple attribute decision making: Methods and applications*. Chapman and Hall/CRC.

Yarlagadda, R., & Hershey, J. (1991). Fast algorithm for computing the reliability of a communication network. *International Journal of Electronics, 70*(3), 549–564.

Yen, G. G., & He, Z. (2013). Performance metric ensemble for multiobjective evolutionary algorithms. *IEEE Transactions on Evolutionary Computation, 18*(1), 131–144.

Yeh, C.-T. (2019). An improved NSGA2 to solve a bi-objective optimization problem of multi-state electronic transaction network. *Reliability Engineering & System Safety, 191*, 106578.

Yeh, C.-T. (2020). A hybrid approach to solve a bi-objective optimization problem of a capacitated-flow network with a time factor. *Reliability Engineering & System Safety, 204*, 107191.

Yeh, C.-T., Chang, P.-C., & Chen, C.-Y. (2017). Minimal production level and reliability measurement for a maintainable production system under demand and budget constraints. *European Journal of Industrial Engineering, 11*(4), 526–547.

Yeh, C.-T., Lin, Y.-K., Yeng, L.C.-L., & Chao, Y.-L. (2023). Hybrid flow-shop manufacturing network reliability optimization using genetic algorithm and absorbing Markov chain. *Applied Mathematical Modelling, 120*, 595–611.

Yu, C., Semeraro, Q., & Matta, A. (2018). A genetic algorithm for the hybrid flow shop scheduling with unrelated machines and machine eligibility. *Computers & Operations Research, 100*, 211–229.

Yu, Q., Zhang, X., & Luo, X. (2020). A decision-making method for bank credit risk evaluation based on improved AHP-entropy-TOPSIS. *Journal of Ambient Intelligence and Humanized Computing, 11*(11), 5277–5291.

Zhang, X., & Bard, J. F. (2006). A multi-period machine assignment problem. *European Journal of Operational Research, 170*(2), 394–415.

Zitzler, E., Laumanns, M., & Thiele, L. (2001). SPEA2: Improving the strength Pareto evolutionary algorithm. TIK-report, 103.

Zuo, M. J., Tian, Z., & Huang, H. Z. (2007). An efficient method for reliability evaluation of multistate networks given all minimal path vectors. *IIE Transactions, 39*(8), 811–817.