Uncertain Random Dependent-Chance Programming for Flow-Shop Scheduling Problem

Achraf Touil, Abdelwahed Echchatbi
Laboratory of Engineering, of Industrial Management and Innovation,
Faculty of Sciences and Technology, Hassan 1st University, PO Box 577, Settat, Morocco
Email: (ac.touil;abdelwahed.echchatbi)@uhp.ac.ma

***Abstract*—In this paper, a type of uncertain random programming model based on the chance measure for the permutation flow shop (PFSP) scheduling problem is proposed with uncertain random job's processing times, i.e., dependent chance programming model (DCPM). The objective is to minimize the total wasted energy consumption induced by the machine idling. Moreover, to solve the proposed model, the uncertain random simulation and a two-stage eagle strategy (ES) are integrated to produce a hybrid intelligent algorithm. In the first stage of ES, the so-called Lévy Flights is employed as the global search algorithm. While in the second stage, the grey-wolf optimizer (GWO) is used as the local search algorithm. The generated hybridization ensures the proper balance between exploration and exploitation. Besides, the Variable Neighborhood Search (VNS) is adopted as local search methods to improve the performance of the highlighted algorithm. The numerical results are reported to demonstrate the applicability of the proposed model.**

***Keywords*—Permutation Flow Shop Problem (PFSP), Total Wasted Energy Consumption (TWEC), Uncertainty Random Theory (URT), Eagle Strategy (ES), Grey-Wolf Optimizer (GWO), Variable Neighbourhood Search (VNS)**

I. Introduction

Green strategies gather the intention of all stakeholders to drive a global performance for energy monitoring and environmental protection. For instance, companies have to deal with the environment's official consensus to cut the different harmful emissions while improving their market competitiveness. Supply chain optimization models have to consider sustainability's aspects like energy consumption and natural resources control. Pointing out effective interactions designing between the pillars of these aspects classify the global mapping of climate goals and customer service level dilemma. Thus, manufacturing processes hold an important role in such a context as companies' productivity is liaised with environmental factors also. In that way, the permutation flow shop problem (PFSP) that is a scheduling problem draws a significant interest for practitioners and researchers, it is considered as the simplified version of the flow shop issues where the sequence of jobs to the process will be the same on each machine. For industry modeling purposes, the previous description feet with many sectors as semiconductor and motor ones [1]. It has also been proved to be non-deterministic-polynomial-time (NP)-hard [2].

The makespan in addition to the total completion time is considered as the most studied scheduling criterion in the literature. The first objective represents the utilization of machines, whereas the second objective is related to the jobs processing rapidity behavior. In recent years, with the growth of sustainable manufacturing, the minimization of energy consumption has been considered as a challenging topic in the scheduling problem. Recently, [3] proposed a new scheduling problem to minimize the energy consumption due to idle times of machines in the permutation flow shop. This type of problem is denoted as $Fm|prum|W$. To solve this problem, a branch and bound algorithm with two lower bounds, and an initial upper bound generated by using a variant of the NEH heuristic algorithm were proposed. The reported numerical results showed the efficiency of the proposed algorithm for small benchmark problems. To overcome the traditional B&B drawbacks for larger benchmarks, a hybrid method combined BSA and simulated annealing algorithm (SA) based on insert local search methods have been proposed by [4]. The experimental results show that HBSA can obtain better performance than B&B and GA.

In these works, the job's processing times are assumed to be deterministic. However, this assumption may be inappropriate in many practical situations. In real production systems, there are many imprecise or uncertain factors involved in the scheduling problems, such as demand and processing times. In the literature, the uncertainty is usually manipulated by three approaches. The first is probability theory. The second is a fuzzy set theory as proposed by [5]. The third is a rough set theory, initialized by [6]. Regarding the PFSP under uncertainty, [7] demonstrated how to integrate the concept of the credibility measurement for the mentioned problem. As a result, three types of fuzzy models are presented, namely, the expected value model (EVM), the chance-constrained model (CCPM), and the dependent chance model (DCPM). A hybrid intelligent algorithm is then designed to solve the proposed models. The processing times have been considered as rough variables by [8]. The objective is to minimize the makespan while considering the same decision models as [7]. To solve this problem, they used a hybrid intelligent algorithm that combined the rough simulation and genetic algorithm.

Many studies emphasized the imprecise information which is neither randomness nor like fuzziness or roughness. In the decision-making process, experts can give a belief degree to some quantities to reflect the human uncertainty statement. A new theory has been developed by [9] accordingly to deal with the highlighted indeterminacy, it is known as uncertainty theory which is a branch of axiomatic mathematics for modeling human uncertainty. Nowadays, uncertainty theory has been adopted for many research fields such as project management and machine scheduling [10],[11]. It is worth mentioning that in many real cases, uncertainty and randomness are incorporated simultaneously in a system. For example, when we processed new jobs and existing jobs in machines, we can estimate the probability distribution of the processing times for existing jobs from historical data. But we cannot derive the probability distribution of processing times of new jobs. To overcome this situation, we can invite experienced experts to estimate the processing times for new jobs. Therefore, the chance theory as a new mathematical tool was presented [12], [13] to addresses uncertain random problems. Until now, the chance theory has been applied to several areas by many studies [14], [15].

The above-mentioned permutation flow shop scheduling problem deals with uncertainty and randomness simultaneously. Therefore, based on chance theory, this paper will present a new model to minimize the total wasted energy consumption (WEC) that occurred by machine idling. The new model is called Dependent-Chance Programming Model(DCPM). In such a non-deterministic environment, traditional methods cannot provide a solution. To overcome this issue, researchers give a central focus for the well known hybrid intelligent algorithm (HIA). In HIA, a type of simulation (i.e. stochastic, or uncertain simulation) is used for calculating the non-deterministic measure (i.e. probability, or uncertain measures) while another method of metaheuristic such as genetic algorithm is adopted to find the quasi-optimal solution. In this paper, a new hybrid intelligent algorithm is proposed. It starts with a simulation technique that is based on uncertain random simulation of the objective function. Then, the global search is performed with an eagle strategy method [16] according to Levy Flights while the local search is completed through Grey Wolf Optimizer (GWO) [17]. Besides, the Variable Neighborhood Search (VNS) [18] is adopted from a performance improvement standpoint.

The rest of this paper is organized as follows: Section 2 contains some preliminaries about uncertainty theory as well as chance theory. In Section 3, the uncertain random energy consumption in the PFSP based on the dependent-chance programming model is presented. In Section 4, a hybrid intelligent algorithm is proposed to solve the model. Section 5 provides the computational results to illustrate the performance of the proposed model and algorithm. Finally, the conclusion is summarized in Section 6

II. Preliminaries

In this section, we will recall some basic concepts that will help establish the permutation flow-shop scheduling problem under uncertainty.

A. Uncertainty theory

Let Γ be a non-empty set and let $\mathcal{L}$ be a σ-algebra on Γ. A set function $\mathcal{M} : \mathcal{L} \to [0, 1]$ is called an uncertain measure if it satisfies the four axioms normality, duality, sub-additivity and product [1].

Definition 1 [9]. An uncertain variable is a function ξ from an uncertainty space $(\Gamma, \mathcal{L}, \mathcal{M})$ to the set of real numbers such that $\{\xi \in \mathcal{B}\}$ is an event for any Borel set $\mathcal{B}$.

Definition 6([1]). The uncertainty distribution ϕ of an uncertain variable ξ is defined by

$$\Phi(x) = \mathcal{M}\{\xi \le x\} \tag{1}$$

for any real number x.

Definition 6([1]). An uncertainty distribution $\Phi(x)$ is said to be regular if it is a continuous and strictly increasing function with respect to x at which $0 < \Phi(x) < 1$, and $\lim_{x\to-\infty} \Phi(x) = 0$, $\lim_{x\to+\infty} \Phi(x) = 1$

[9] gave some types of uncertainty distributions to describe uncertain variables. The paper used only linear uncertainty distribution. Therefore, we only state them in the following text.

Definition 6([1]). An uncertain variable ξ is called linear if it has a linear uncertainty distribution

$$\Phi(x) = \begin{cases} 0, & \text{if } x < a \\ (x-a)/(b-a), & \text{if } a \le x \le b \\ 1, & \text{if } x > b \end{cases} \tag{2}$$

Denoted by $\mathcal{L}(a, b)$ where a, b are real number with $a < b$.

Definition 6([1]). Let ξ be an uncertain variable with regular uncertainty distribution $\phi(x)$. Then the inverse function $\phi^{-1}(\alpha)$ is called the inverse uncertainty distribution of ξ. For example, the inverse uncertainty distribution of linear uncertain variable $\mathcal{L}(a, b)$ is

$$\phi^{-1}(\alpha) = \alpha \times a + (1-\alpha) \times b \tag{3}$$

Definition 6([1]). Let $\xi_1, \xi_2, \ldots, \xi_n$ be independent regular uncertain variables with uncertainty distributions $\Phi_1, \Phi_2, ..., \Phi_n$, respectively. If the function is strictly increasing with respect to $x_1, x_2, ..., x_m$, and strictly decreasing with respect to $x_{m+1}, x_{m+2}, ..., x_n$, then $\xi = f(\xi_1, \xi_2, \ldots, \xi_n)$ is an uncertain variable with inverse uncertainty distribution

$$\Psi^{-1}(\alpha) = f(\Phi_1^{-1}(\alpha), \Phi_2^{-1}(\alpha), ..., \Phi_m^{-1}(\alpha), \Phi_{m+1}^{-1}(1-\alpha), \Phi_{m+2}^{-1}(1-\alpha), \Phi_n^{-1}(1-\alpha)) \tag{4}$$

B. Chance theory

Based on the definitions of uncertain variable and random variable, the concept of an uncertain random variable can be given as follows:

Let $(\Omega, \mathcal{A}, \Pr)$ be a probability space and $(\Gamma, \mathcal{L}, \mathcal{M})$ be an uncertainty space. Then $(\Omega \times \Gamma, \mathcal{A} \times \mathcal{L}, \Pr \times \mathcal{M})$ is called a chance space.

Definition 6([1]). Let ξ be an uncertain random on a chance space $(\Omega \times \Gamma, \mathcal{A} \times \mathcal{L}, \Pr \times \mathcal{M})$. Then its chance distribution Φ is defined by

$$\Phi(x) = \mathrm{Ch}\{\xi \le x\} = \int_0^1 \Pr\{\omega \in \Omega | \mathcal{M}\{\gamma \in \Gamma | \xi(\omega, \gamma) \le x\} \ge r\} dr \quad (5)$$

Theorem 6([1]). Let $\eta_1, \eta_2, ..., \eta_m$ be independent random variables with probability distributions $\Psi_1, \Psi_2, ..., \Psi_m$, respectively, and let $\tau_1, \tau_2, ..., \tau_n$ be independent uncertain variables. Assume f is a measurable function. Then the uncertain random variable $\xi = f(\eta_1, \eta_2, ..., \eta_m, \tau_1, \tau_2, ..., \tau_n)$ has a chance distribution

$$\Phi(x) = \int_{\mathcal{R}^m} F(x; y_1, y_2, ..., y_m) d\Psi_1(y_1) d\Psi_2(y_2) ... d\Psi_m(y_m) \quad (6)$$

where $F(x; y_1, y_2, ..., y_m)$ is the uncertainty distribution of the variable $f(y_1, y_2, ..., y_m, \tau_1, \tau_2, ..., \tau_n)$.

Theorem 6([1]). Let $\eta_1, \eta_2, ..., \eta_m$ be independent random variables with probability distributions $\Psi_1, \Psi_2, ..., \Psi_m$, and let $\tau_1, \tau_2, ..., \tau_n$ be independent uncertain variables with regular uncertainty distributions $\Upsilon_1, \Upsilon_2, ..., \Upsilon_n$, respectively. If $f(\eta_1, \eta_2, ..., \eta_m, \tau_1, \tau_2, ..., \tau_n)$ is strictly increasing with respect $\tau_1, \tau_2, ..., \tau_k$ and strictly decreasing with respect $\tau_k + 1, \tau_2, ..., \tau_n$, then

$$\mathrm{Ch}\{f(\eta_1, \eta_2, ..., \eta_m, \tau_1, \tau_2, ..., \tau_n) \le 0\} = \int_{\mathcal{R}^m} G(y_1, y_2, ..., y_m) d\Psi_1(y_1) d\Psi_2(y_2) ... d\Psi_m(y_m) \quad (7)$$

where $G(y_1, y_2, ..., y_m)$ is the root α of the equation

$$f(y_1, y_2, ..., y_m, \Upsilon_1^{-1}(\alpha), \Upsilon_2^{-1}(\alpha), ..., \Upsilon_k^{-1}(\alpha), \Upsilon_{k+1}^{-1}(1-\alpha), \Upsilon_{k+2}^{-1}(1-\alpha), ..., \Upsilon_n^{-1}(1-\alpha)) = 0 \quad (8)$$

III. Uncertain Random permutation flow shop scheduling problem

The Permutation Flow Shop Problem (PFSP) is a typical combinatorial optimization problem, which determines the processing sequence of jobs over machines in order to minimizes many objectives. Assumptions and notation described as follows:

- All jobs are available at time zero and the processing time is assumed to be an uncertain random variable.
- Each machine can process at most one job at any time;
- Each job can be processed on at most one machine at any time;
- Each job must complete processing without preemptions;
- Machines are turned on as soon as the first job arrives and turned off when the last job leaves;
- Machines will keep being idle during two successive jobs;

A. Symbols and formulation

n: Denotes the number of jobs;
m: Denotes the number of machines;
$\xi_{i,j} = (\eta_{i,j}, \tau_{i,j})$: Denotes the uncertain random processing time of the job i on machine j where $\eta_{i,j}$ are independent random variables with probability distributions $\Psi_{i,j}$, and $\tau_{i,j}$ are independent uncertain variables with uncertainty distributions $\Upsilon_{i,j}$ respectively.
$\boldsymbol{\xi} = (\xi_{1,1}, \xi_{1,2}, ..., \xi_{n,m})$: Denotes the uncertain random vector;
λ_i: Denotes the rated output power of machine j;
$\boldsymbol{x} = (x_1, x_2, ..., x_n)$: The integer decision vector that represents the schedule

$$x_1 \to x_2 \to ... \to x_n \quad (9)$$

For simplicity, we use $\boldsymbol{\xi} = (\xi_{1,1}, \xi_{1,2}, ..., \xi_{n,m})$ (i.e. $= (\eta_{1,1}, \eta_{1,2}, ..., \eta_{n,m}; \tau_{1,1}, \tau_{1,2}, ..., \tau_{n,m})$ to denote the uncertain random vector and note that the full schedule is represented by the integer decision vector $\boldsymbol{x} = (x_1, x_2, ..., x_n)$, which is representing a permutation of n jobs with $1 \le x_i \le n$ and $x_i \ne x_k$ for all $i \ne k, \ \ i, k = 1, 2, ..., n$.

Let $C(x_i, j, \boldsymbol{\xi})$ and $IT(x_i, j, \boldsymbol{\xi})$ be the uncertain random completion time and idle time of job x_i on the machine j respectively. Then are calculated by the following equations:

$$\begin{cases} C(x_1, 1, \boldsymbol{\xi}) = \xi_{x_1,1} \\ C(x_i, 1, \boldsymbol{\xi}) = C(x_{i-1}, 1, \boldsymbol{\xi}) + \xi_{x_i,1}, \quad i = 2, ..., n \\ C(x_1, j, \boldsymbol{\xi}) = C(x_1, j-1, \boldsymbol{\xi}) + \xi_{x_1,j}, \quad j = 2, ..., m \\ C(x_i, j, \boldsymbol{\xi}) = max\{C(x_{i-1}, j, \boldsymbol{\xi}), C(x_i, j-1, \boldsymbol{\xi})\} + \xi_{x_i,j} \\ \qquad i = 2, ..., n, j = 2, ..., m \end{cases} \quad (10)$$

$$\begin{cases} IT(x_i, j, \xi) = 0, \quad i = 1, j = 1 \\ IT(x_i, j, \boldsymbol{\xi}) = max\{C(x_i, j-1, \boldsymbol{\xi}) - C(x_{i-1}, j, \boldsymbol{\xi}), 0\} \\ \qquad i = 2, ..., n, j = 2, ..., m \end{cases} \quad (11)$$

Let $TEC(\boldsymbol{x}, \xi)$ be the total uncertain random energy consumption of the schedule $\boldsymbol{x}$. According to [3], we have:

$$\begin{cases} TEC(\boldsymbol{x}, \xi) = TUEC(\boldsymbol{x}, \xi) + TWEC(\boldsymbol{x}, \xi), \\ TUEC(\boldsymbol{x}, \xi) = \sum_{i=1}^{n} U_i = \sum_{i=1}^{n} \sum_{j=1}^{m} \lambda_j * \xi_{i,j} \\ TWEC(\boldsymbol{x}, \xi) = \sum_{j=1}^{m} W_j = \sum_{j=1}^{m} \lambda_j * \sum_{i=1}^{n} IT(x_i, j, \boldsymbol{\xi}) \end{cases} \quad (12)$$

where $TUEC(\boldsymbol{x}, \xi)$ denotes the total uncertain random useful energy consumption and $TWEC(\boldsymbol{x}, \xi)$ denotes the total uncertain random wasted energy consumption.

IV. Uncertain random dependent chance model

Dependent-chance programming (DCP) initialized by [19], is a powerful decision-making criterion which concerns the risk of some unfavorable event occurring. [15] extended the dependent-chance programming to deal with an uncertain random time-cost trade-off problem (TCTP) which is a type of project scheduling problem.

In scheduling problems, decision-maker may want to control the total wasted energy consumption. Hence, it is natural for decision-maker to maximize the chance degree that the total uncertain random wasted energy consumption does not exceed some given target wasted energy consumption. According to this idea, the uncertain random dependent-chance programming model for TWEC-PFSP is written as follows:

$$\begin{cases} \max \ \ \mathrm{Ch}\left\{TWEC(\boldsymbol{x},\boldsymbol{\xi}) \leq TWEC_0\right\} \\ \text{Subject to:} \\ 1 \leq x_i \leq n, i = 1,2,...,n \\ x_i \neq x_k, i \neq k \ \ i,k = 1,2,...,n \\ x_i \in Z, \ \ i = 1,2,...,n \end{cases} \tag{13}$$

where $TWEC_0$ is the predetermined target of the wasted energy consumption.

Remark 1:If the uncertain random variable ξ degenerates to an uncertain variable, the proposed model becomes an uncertain dependent chance model with uncertain processing times.

Remark 2:If the uncertain random variable ξ degenerates to a random variable, the proposed model becomes a stochastic dependent chance model with random processing times

V. HYBRID INTELLIGENT ALGORITHM

In this section, we will design a hybrid intelligent algorithm to solve the uncertain random dependent-chance model, where uncertain random simulation and eagle strategy with grey wolf optimizer based on variable neighborhood search are used.

A. Uncertain Random Simulation

Recently, [15] propose a new HIA for project scheduling problems within the uncertain random theory. The proposed method combine two uncertain random simulation algorithms that are based on stochastic and uncertain simulation to estimate the expected value and dependent chance value embedded with the genetic algorithm to find the quasi-optimal solution.

In this paper, we will use the proposed uncertain random simulation of chance value. The following steps are given in Algorithm 1.

$$U : \boldsymbol{x} \to \mathrm{Ch}\left\{TWEC(\boldsymbol{x},\xi) \leq TWEC_0\right\} \tag{14}$$

999-method

The 999-method was proposed by [9] to calculate uncertain variables. It is suggested that an uncertain variable can be represented by a 999-table. The first row contains the values of uncertainty distribution, while the second row presents the corresponding values of inverse uncertainty distribution $\Omega^{-1}(x,\alpha)$ of the total wasted energy consumption as stated in table 1.

Algorithm 1 Uncertain-Random Simulation for Chance value:

1: Set $e = 0$;
2: Generate $\omega_1, \omega_2, ..., \omega_N$ from Ω according to the probability Pr.
3: Consider $\omega_i = \{\eta_{1,1}, \eta_{1,2}, ..., \eta_{n,m}\}, i = 1,2,..,N$
4: $e \leftarrow e + \mathcal{M}\left\{TWEC(\boldsymbol{x},\xi(\omega_k)) \leq TWEC_0\right\}, k = 1,2,...,N$ where $\mathcal{M}\left\{TWEC(\boldsymbol{x},\xi(\omega_k)) \leq TWEC_0\right\}$ is given by uncertain simulation (i.e. 999-method) as given in the following section.
5: Repeat the second and third steps N times, where N is a sufficiently large number;
6: Return $\dfrac{e}{N}$;

TABLE I
INVERSE UNCERTAINTY DISTRIBUTION OF $\Omega^{-1}(x,\alpha)$

α	*0.01*	*0.02*	...	*0.999*
$\Omega^{-1}(x,\alpha)$	s_1	s_2	...	s_{999}

The value of $\mathcal{M}\left\{TWEC(\boldsymbol{x},\boldsymbol{\xi}) \leq TWEC_0\right\}$ is equivalent to $\Omega(TWEC(\boldsymbol{x},TWEC_0))$ which can be approximately estimated by :

$$\Omega(\boldsymbol{x},TWEC_0) = \frac{k}{100}, \quad \text{if } s_k \leq TWEC_0 < s_{k+1} \text{ for some } k \tag{15}$$

where $s_1, ..., s_{999}$ are given by Table 1.

The main steps of uncertain simulation of $\mathcal{M}\left\{TWEC(\boldsymbol{x},\boldsymbol{\xi}) \leq TWEC_0\right\}$ is given in Algorithm 2.

Algorithm 2 Uncertain simulation of $\mathcal{M}\left\{TWEC(\boldsymbol{x},\boldsymbol{\xi}) \leq TWEC_0\right\}$

1: Set $DCM \leftarrow 0$, $u \leftarrow 1$;
2: If ($s^u \leq TWEC_0 < s^{u+1}$), let $DCM \leftarrow \dfrac{u}{100}$;
3: If ($u < 998$), let $u \leftarrow u + 1$. Turn back to Step 2;
4: Report DCM as the estimation of $\mathcal{M}\left\{TWEC(\boldsymbol{x},\boldsymbol{\xi}) \leq TWEC_0\right\}$;

B. Brief introduction to Eagle Strategy (ES)

Eagle strategy is a two-stage optimization strategy that was presented by [16]. This algorithm mimics the hunting behavior of eagles in nature. Eagles forage using two components: random search performed by flying freely and intensive search to catch prey when sighted. In this two-stage strategy, the first stage explores the search space globally by using a Levy flight; if it finds a promising solution, then an intensive local search is employed using a more efficient local optimizer, such as hill-climbing and the downhill simplex method. Then, the two-stage process starts again with new global exploration, followed by a local search in a new region.

C. Brief introduction to Grey Wolf Optimizer (GWO)

GWO is an efficient population-based optimizer recently proposed by [17] which can provide a more efficient performance compared to other well-established optimizers. The

GWO mimics the ideal hunting behavior of wolf packs through the leadership hierarchy as well as the hunting mechanism of grey wolves in nature. Alpha (α), beta (β), delta (δ), and omega (ω) are used to denote four separate types of grey wolves to properly simulate the leadership hierarchy. Three main steps are used to ensure optimal performance which is known as: searching for prey, encircling prey, and attacking prey formulated mathematically as below :

$$\begin{aligned} \overrightarrow{D} &= |\overrightarrow{C}.\overrightarrow{X}_p(t) - \overrightarrow{X}| \\ \overrightarrow{X}(t+1) &= \overrightarrow{X}_p(t) - \overrightarrow{A}.\overrightarrow{D} \end{aligned} \tag{16}$$

$$\begin{aligned} \overrightarrow{A} &= 2.\overrightarrow{a}.\overrightarrow{r}_1 - \overrightarrow{a} \\ \overrightarrow{C} &= 2.\overrightarrow{r}_2. \end{aligned} \tag{17}$$

$$\begin{aligned} \overrightarrow{X}_1 &= \overrightarrow{X}_\alpha - \overrightarrow{A}_1.\overrightarrow{D}_\alpha; \overrightarrow{D}_\alpha = |\overrightarrow{C}_1.\overrightarrow{X}_\alpha - \overrightarrow{X}| \\ \overrightarrow{X}_2 &= \overrightarrow{X}_\beta - \overrightarrow{A}_2.\overrightarrow{D}_\beta; \overrightarrow{D}_\beta = |\overrightarrow{C}_2.\overrightarrow{X}_\beta - \overrightarrow{X}| \\ \overrightarrow{X}_3 &= \overrightarrow{X}_\delta - \overrightarrow{A}_3.\overrightarrow{D}_\delta; \overrightarrow{D}_\delta = |\overrightarrow{C}_3.\overrightarrow{X}_\delta - \overrightarrow{X}| \end{aligned} \tag{18}$$

$$\overrightarrow{X}(t+1) = \frac{\overrightarrow{X}_1 + \overrightarrow{X}_2 + \overrightarrow{X}_3}{3} \tag{19}$$

Equations (17) and (18) are used to calculate the encircling behavior where t indicates the current iteration, $\overrightarrow{A}$ and $\overrightarrow{C}$ are coefficient vectors, $\overrightarrow{X}_p$ is the position vector of the prey, and $\overrightarrow{X}$ indicates the position vector of a grey wolf. where components of $\overrightarrow{a}$ are linearly decreased from 2 to 0 over the course of iterations and $\overrightarrow{r}_1$, $\overrightarrow{r}_2$ are random vectors in [0, 1]. Equations (19) and (20) simulate the hunting behavior, in a mathematical setting, we assume that the alpha (α), beta (β), delta (δ) know the potential location of prey. When random value $|A| < 1$, the wolves are forced to attack the prey as exploitation mode. When $|A| > 1$, the members of the population are enforced to diverge from the prey as exploration mode. In every iteration, the three best individuals are saved and guide the others to update their positions. More details of the GWO can be found in [17].

D. Brief introduction to Variable Neighborhood Search (VNS)

Variable Neighborhood Search (VNS) [18] is explained as a systematic change in both global and combinatorial problems with optimization. VNS has been used in a wide selection of literature as the local search methodology. Regardless of the obvious performance of hybridization with VNS, the performance depends on the neighborhood operation used.

E. The proposed hybrid intelligent algorithm

Since eagle strategy and grey wolf optimizer; both of which were designed to solve continuous optimization problems. To solve a discrete problem such as TWEC-PSP, they are incorporated with several methods mainly: representation, population initialization, and generate solutions. Furthermore, the uncertain random simulation is used the compute the fitness. All these steps will be discussed in the following subsections. Algorithm 3 describes the pseudo-code of the proposed hybrid intelligent algorithm where R is the number of rounds, and N_p is the population size.

Algorithm 3 The proposed hybrid intelligent algorithm (HIA)

Initialize the parameters and use NEH heuristic (Algorithm 4) to produce 10% agents and the rest of agents are generated randomly according to Equation (22);
Rank the wolf pack as : X_α, X_β and X_δ;
while ($R > r$) **do**
  Generate a set of agents — solutions X for global exploration using the Levy flight according to Equation (21), where the feasibility must be offered;
  Convert each agent X_i of the set X to a job permutation π_i by using the LOV rule;
  For each permutation, calculate the total wasted energy consumption ($TWEC(\boldsymbol{x}, \boldsymbol{\xi})$) according to Equation (12) using uncertain random simulation using Algorithms 1 & 2;
  Update the best solutions obtained so far (X_α, X_β and X_δ)
Inner loop :
  Generate randomly a set of agents around this promising solution, where feasibility must be offered;
Carry out an intensive local search via the Grey Wolf Optimizer
  Execute the local search by using VNS (Algorithm 5);
  for $i = 1 \;\; to \;\; NP$ **do**
    Update the position by Equation (19);
  end for
  Update A, C and a;
  Convert each wolf X_i of the set X to a job permutation π_i by using the LOV rule;
  For each permutation, calculate the total wasted energy consumption ($TWEC(\boldsymbol{x}, \boldsymbol{\xi})$) according to Equation (12) using uncertain random simulation using Algorithms 1 & 2;
  Rank the updated wolf pack as alpha, beta and delta (X_α), beta (X_β), delta (X_δ);
  if (a better solution is found) **then**
    Update the current best
  end if
End Inner loop
  Update $r = r + 1$
end while
Report the best solution as the optimal schedule

*1) **Solution representation**:* The proposed strategy, combine the so-called Levy flight and the grey wolf optimizer, both of which were designed to solve continuous optimization problems, and cannot be applied directly to the discrete problem, such as TWEC-PFSP. One key to apply ES-GWO to solve TWEC-PFSP is to construct a direct relationship between the job sequence and the vector solution of ES-GWO. In this paper, we will use the largest-order-value (LOV) mechanism from the research literature of [20] to map the ES-GWO solution $X_i = [X_{i,1}, X_{i,2}, ..., X_{i,n}]$ to job solution/permutation vector $\pi_i = [\pi_{i,1}, \pi_{i,2}, ..., \pi_{i,n}]$. The basic idea of this rule is ranked X_i by descending order to obtain a sequence $\phi_i = [\phi_{i,1}, \phi_{i,2}, ..., \phi_{i,n}]$. Then, the job permutation π_i is calculated by the following formula $\phi_{i,k} = k$ where the dimension k varies from 1 to n.

*2) **Population initialization**:* To guarantee an initial population with certain quality and diversity, an adopted version of NEH heuristic for total wasted energy proposed by [3] is used to generate 10% of N. The pseudo-code is shown in algorithm 4. The rest of 90% of N of solutions generated

Algorithm 4 NEH heuristic

α : Jobs ordered by decreasing order of $TUEC(\boldsymbol{x}, \xi$ according to (Eq.12) $\alpha = [\alpha_1, \alpha_2, ..., \alpha_n]$
$\pi : \{\alpha_1\}$
for $k = 2$ to n **do**
    Test job α_k in any possible position of π
    π : Permutation obtained by inserting α_k in the position of π which less total wasted energy consumption ($TWEC(\boldsymbol{x}, \boldsymbol{\xi})$) calculated according to Equation (12) by using Algorithms 1 & 2;
end for

randomly according to the following equation:

$$X_{p,j} = X_j^{max} - RND() \times (X_j^{max} - X_j^{min}) \tag{20}$$

where $p = 1, 2, ..., NP$, $j = 1, 2, ..., n$, and r is a uniform random number between 0 and 1.

*3) **Generate new solutions**:*

a) Exploration phase using Lévy Flights: As mentioned above, the ES is a two-stage strategy, and we can use different algorithms at different stages. In the first stage, ES uses the so-called Lévy flights, which represent a kind of non-Gaussian stochastic process whose step sizes are distributed based on a Levy stable distribution to generate new solutions. When a new solution is produced, the following Levy flight is applied:

$$X_i^{t+1} = X_i^t + \alpha \oplus Levy(\beta) \tag{21}$$

Here, α is the step size that is relevant to the scales of the problem generally chosen as $\alpha = 0.01$. The step length $Levy(\lambda)$ can be calculated by using Mantegna's algorithm [21].

b) Exploitation phase using GWO based on VNS: For the second stage, we can use various efficient meta-heuristic algorithms like grey wolf optimizer to do a strenuous local search. We know the GWO is a global search algorithm, but it can easily be tuned to do an efficient local search by limiting new solutions locally around the most promising region. As mentioned above in GWO, the other wolves (ω) are forced to update their positions according to the position of the best search agents. For further improvement of the computational performance of local search ability, a VNS is applied to the best agents. In this paper, three kinds of neighborhoods are used from literature mainly: Insertion, Interchange, and Swap; The pseudo-code of VNS is shown in algorithm 5.

Algorithm 5 Local Search by using VNS

Denotations
$Iter; Iter_{max}; N_{Iter}(.); TWEC(.)$ denotes index of the neighbourhood structure; Total of neighbourhood structures; the neighbourhood structure (i.e : Insert, Interchange and Swap; The objective function (i.e Total wasted energy consumption using uncertain random simulation)
Convert each best (alpha, beta and delta) agent X_1 to a discrete job permutation π_i
$Iter = 0$; $Iter_{max} = 3$
while ($Iter \leq Iter_{max}$) **do**
    Randomly generates a neighbor $\pi_i^{'} \in N_{Iter}(\pi_i)$;
    if ($TWEC(\pi_i, \boldsymbol{\xi}) > TWEC(\pi_i^{'}, \boldsymbol{\xi})$) **then**
        $\pi_i = \pi_i^{'}$
        $Iter = 1$
    else
        $Iter = Iter + 1$
    end if
end while

VI. Numerical Results

In this section, we illustrate the numerical results to (i) to demonstrate the application of the model, (ii) to test the robustness of the proposed algorithm. We assume that there are 10 jobs and 5 machines. The processing times are addressed by uncertain random variables $\xi_{i,j} = (\eta_{i,j} + \tau_{i,j}); i = 1, ..., n,\ j = 1, ..., m$ where $\eta_{i,j}; i = 1, ..., n,\ j = 1, ..., m$ are uncertain linear distributions and $\tau_{i,j}; i = 1, ..., n,\ j = 1, ..., m$ are random linear distributions are presented in Table 2. The output rated power ($\lambda_j; j = 1, ..., m$) is set to 1 for all machines. We consider 2000 cycles in stochastic simulation).The predetermined level of total wasted energy consumption ($TWEC_0$) is set to 7650.

Table 2 reports the computational results obtained for different set of parameters as follow :

- **Column 1 & 2**: The algorithm parameters ($N_{pop}; G_{max}$)
- **Column 3**: The obtained objective value ($TWEC$) by ES-GWO-VNS
- **Column 4**: The relative error computed according to equation 23 :

$$\frac{V_{optimal} - V_{actual}}{V_{optimal}} * 100\% \tag{22}$$

where $V_{optimal}$ represents the optimal objective value (i.e maximum) of all objective values obtained with different parameters and V_{actual} is the objective value obtained for a given experiment with the given parameters.

It follows from column 3 of table 2 that the error does not exceed 6.45%, which implies that the designed algorithm is robust to the parameter settings when solving the problem considered in this paper.

Figures 3 and 4 represent an intuitive analysis of objective value and error. Figures 3 and 4 shows that the objective value (TWEC) increases and error decrease significantly when the population size and maximum of generations increase.

TABLE II
UNCERTAIN RANDOM PROCESSING TIMES

	M1	***M2***	***M3***	***M4***	***M5***
J1	$\mathcal{L}(382,456)+\mathcal{U}(38,45)$	$\mathcal{L}(463,537)+\mathcal{U}(46,53)$	$\mathcal{L}(49,123)+\mathcal{U}(49,54)$	$\mathcal{L}(140,214)+\mathcal{U}(14,24)$	$\mathcal{L}(160,234)+\mathcal{U}(16,23)$
J2	$\mathcal{L}(715,789)+\mathcal{U}(71,78)$	$\mathcal{L}(780,854)+\mathcal{U}(78,85)$	$\mathcal{L}(151,225)+\mathcal{U}(15,22)$	$\mathcal{L}(454,528)+\mathcal{U}(44,52)$	$\mathcal{L}(49,123)+\mathcal{U}(14,24)$
J3	$\mathcal{L}(802,876)+\mathcal{U}(80,87)$	$\mathcal{L}(558,632)+\mathcal{U}(55,63)$	$\mathcal{L}(514,588)+\mathcal{U}(51,58)$	$\mathcal{L}(822,896)+\mathcal{U}(82,89)$	$\mathcal{L}(382,456)+\mathcal{U}(38,45)$
J4	$\mathcal{L}(469,543)+\mathcal{U}(46,54)$	$\mathcal{L}(71,145)+\mathcal{U}(7,14)$	$\mathcal{L}(595,669)+\mathcal{U}(59,66)$	$\mathcal{L}(251,325)+\mathcal{U}(25,32)$	$\mathcal{L}(715,789)+\mathcal{U}(71,78)$
J5	$\mathcal{L}(136,210)+\mathcal{U}(13,21)$	$\mathcal{L}(711,785)+\mathcal{U}(71,78)$	$\mathcal{L}(892,966)+\mathcal{U}(89,96)$	$\mathcal{L}(73,147)+\mathcal{U}(73,84)$	$\mathcal{L}(802,876)+\mathcal{U}(80,87)$
J6	$\mathcal{L}(49,123)+\mathcal{U}(9,12)$	$\mathcal{L}(140,214)+\mathcal{U}(14,21)$	$\mathcal{L}(258,332)+\mathcal{U}(25,33)$	$\mathcal{L}(782,856)+\mathcal{U}(78,85)$	$\mathcal{L}(469,543)+\mathcal{U}(46,54)$
J7	$\mathcal{L}(382,456)+\mathcal{U}(38,45)$	$\mathcal{L}(678,752)+\mathcal{U}(67,75)$	$\mathcal{L}(70,144)+\mathcal{U}(7,14)$	$\mathcal{L}(247,321)+\mathcal{U}(24,32)$	$\mathcal{L}(136,210)+\mathcal{U}(13,21)$
J8	$\mathcal{L}(715,789)+\mathcal{U}(71,78)$	$\mathcal{L}(69,143)+\mathcal{U}(6,14)$	$\mathcal{L}(681,755)+\mathcal{U}(68,75)$	$\mathcal{L}(353,427)+\mathcal{U}(35,42)$	$\mathcal{L}(49,123)+\mathcal{U}(4,12)$
J9	$\mathcal{L}(802,876)+\mathcal{U}(80,87)$	$\mathcal{L}(624,698)+\mathcal{U}(62,69)$	$\mathcal{L}(248,322)+\mathcal{U}(24,32)$	$\mathcal{L}(472,546)+\mathcal{U}(47,54)$	$\mathcal{L}(382,456)+\mathcal{U}(38,45)$
J10	$\mathcal{L}(469,543)+\mathcal{U}(46,54)$	$\mathcal{L}(458,532)+\mathcal{U}(45,53)$	$\mathcal{L}(26,100)+\mathcal{U}(6,10)$	$\mathcal{L}(247,321)+\mathcal{U}(24,32)$	$\mathcal{L}(715,789)+\mathcal{U}(71,78)$

TABLE III
COMPUTATIONAL RESULTS

N_{pop}	G_{max}	**ES-GWO-VNS**	
		Objective value	Error
30	100	0.87	6.45
	300	0.90	3.23
	500	0.93	0.00
50	100	0.89	4.30
	300	0.93	0.00
	500	0.93	0.00
80	100	0.90	4.26
	300	0.94	0.00
	500	0.94	0.00

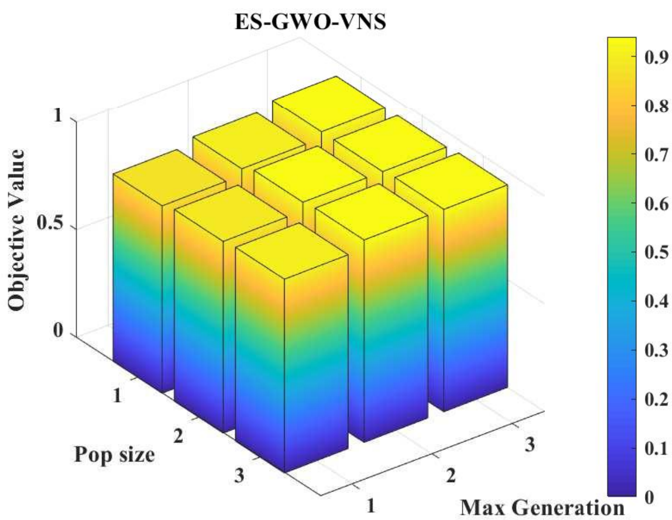

Fig. 1. Objective value results with different set of parameters

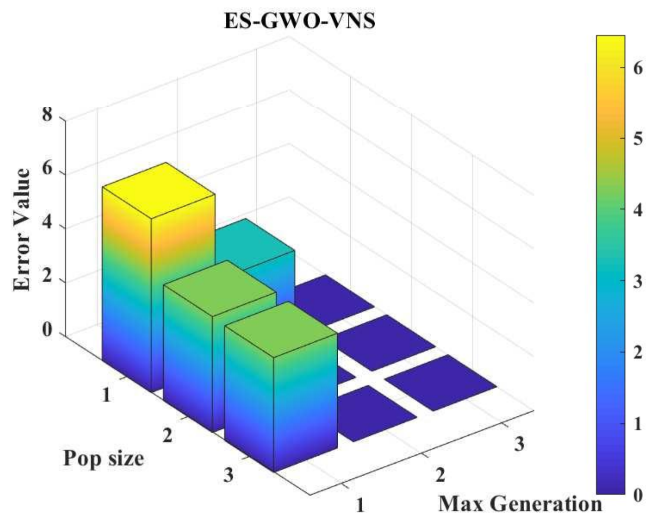

Fig. 2. Error value results with different set of parameters

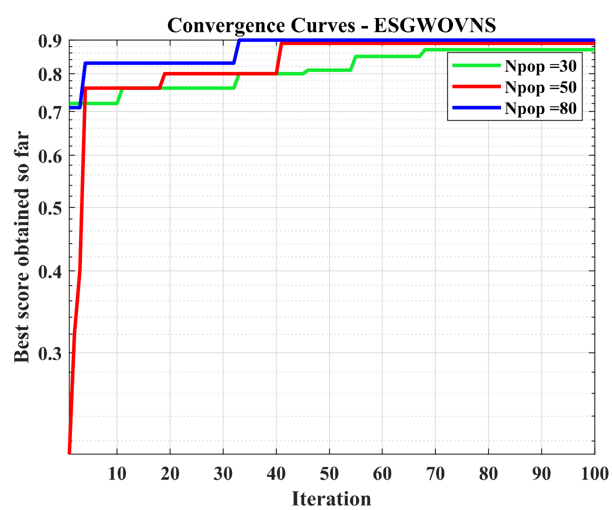

Fig. 3. Convergence Curve of $G_{max} = 100$

Figures 3-5 provide the convergence curves of ES-GWO-VNS for a given N_{pop} and G_{max}.

VII. CONCLUSION

In this paper, we investigate the total wasted energy consumption in the permutation flow-shop problem under uncertainty and randomness simultaneously. The main contributions of this paper are discussed below. First, the proposed

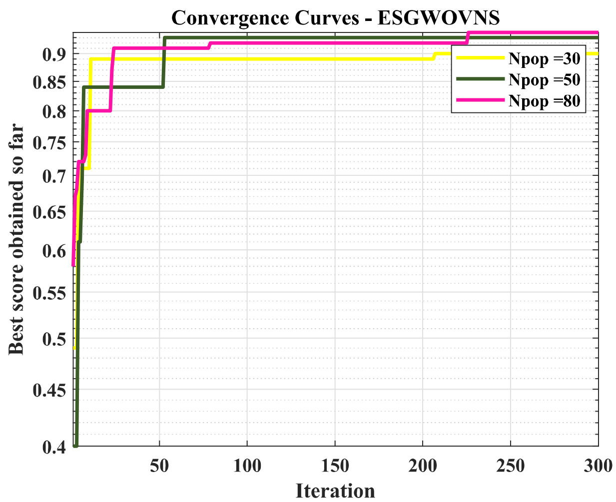

Fig. 4. Convergence Curve of $G_{max} = 300$

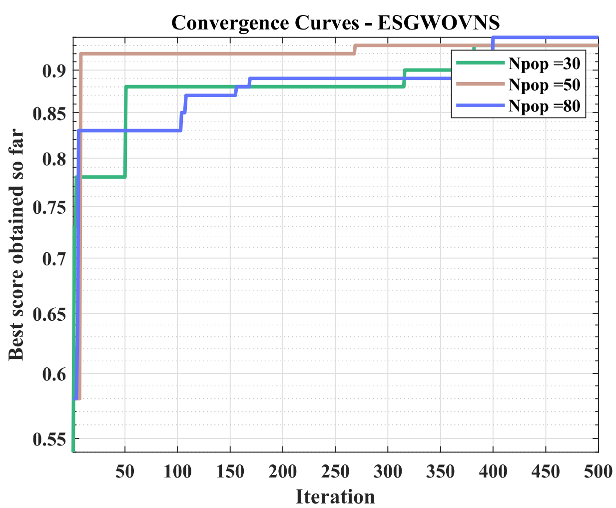

Fig. 5. Convergence Curve of $G_{max} = 500$

model uses the uncertainty random theory and the dependent-chance programming model. The uncertainty random theory represents a powerful alternative to other classical theories, such as uncertainty, random, fuzzy, and rough. Therefore, the proposed dependent-chance programming model can provide better theoretical directions for decision-makers. Second, a hybrid intelligence based on an eagle strategy combined with a grey-wolf optimizer based VNS is developed to solve the model. Furthermore, the computational experiments demonstrate that the proposed strategy is more competitive and efficient to solve the model

References

[1] Q.-K. Pan and L. Wang, "Effective heuristics for the blocking flowshop scheduling problem with makespan minimization," *Omega*, vol. 40, no. 2, pp. 218–229, 2012.

[2] A. R. Kan, *Machine scheduling problems: classification, complexity and computations*. Springer Science & Business Media, 2012.

[3] G.-S. Liu, B.-X. Zhang, H.-D. Yang, X. Chen, and G. Q. Huang, "A branch-and-bound algorithm for minimizing the energy consumption in the pfs problem," *Mathematical Problems in Engineering*, vol. 2013, 2013.

[4] P. Chen, L. Wen, R. Li, and X. Li, "A hybrid backtracking search algorithm for permutation flow-shop scheduling problem minimizing makespan and energy consumption," in *2017 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, Dec 2017, pp. 1611–1615.

[5] L. A. Zadeh, "Fuzzy sets," *Information and control*, vol. 8, no. 3, pp. 338–353, 1965.

[6] Z. Pawlak, "Rough sets," *International Journal of Parallel Programming*, vol. 11, no. 5, pp. 341–356, 1982.

[7] J. Peng and K. Song, "Fuzzy flow-shop scheduling models based on credibility measure," in *Fuzzy Systems, 2003. FUZZ'03. The 12th IEEE International Conference on*, vol. 2. IEEE, 2003, pp. 1423–1427.

[8] J. Peng and K. Iwarnura, "Flow-shop scheduling models with parameters represented by rough variables," *Tsinghua Science and Technology*, vol. 8, no. 1, pp. 55–59, 2003.

[9] X.-S. Yang and S. Deb, "Eagle strategy using lévy walk and firefly algorithms for stochastic optimization," in *Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)*. Springer, 2010, pp. 101–111.

[10] X.-S. Yang, S. Deb, and X. He, "Eagle strategy with flower algorithm," in *Advances in Computing, Communications and Informatics (ICACCI), 2013 International Conference on*. IEEE, 2013, pp. 1213–1217.

[11] A. Touil, A. Echchatbi, A. Charkaoui, and A. Mousrij, "Uncertain chance-constrained model for energy consumption in the permutation flow shop," *IFAC-PapersOnLine*, vol. 52, no. 11, pp. 152–157, 2019.

[12] Y. Liu, "Uncertain random variables: a mixture of uncertainty and randomness," *Soft Computing*, vol. 17, no. 4, pp. 625–634, 2013.

[13] Y. Liu, "Uncertain random programming with applications," *Fuzzy Optimization and Decision Making*, vol. 12, no. 2, pp. 153–169, 2013.

[14] S. Ding, "Uncertain random newsboy problem," *Journal of Intelligent & Fuzzy Systems*, vol. 26, no. 1, pp. 483–490, 2014.

[15] H. Ke, H. Liu, and G. Tian, "An uncertain random programming model for project scheduling problem," *International Journal of Intelligent Systems*, vol. 30, no. 1, pp. 66–79, 2015.

[16] A. Charnes and W. W. Cooper, *Management models and industrial applications of linear programming*. JSTOR, 1961, vol. 1.

[17] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Advances in engineering software*, vol. 69, pp. 46–61, 2014.

[18] P. Hansen and N. Mladenović, "Variable neighborhood search: Principles and applications," *European journal of operational research*, vol. 130, no. 3, pp. 449–467, 2001.

[19] B. Liu, "Dependent-chance programming: A class of stochastic optimization," *Computers & Mathematics with Applications*, vol. 34, no. 12, pp. 89–104, 1997.

[20] X. Li and M. Yin, "A hybrid cuckoo search via lévy flights for the permutation flow shop scheduling problem," *International Journal of Production Research*, vol. 51, no. 16, pp. 4732–4754, 2013.

[21] R. N. Mantegna, "Fast, accurate algorithm for numerical simulation of levy stable stochastic processes," *Physical Review E*, vol. 49, no. 5, p. 4677, 1994.