

Article

A Fuzzy Simheuristic for the Permutation Flow Shop Problem under Stochastic and Fuzzy Uncertainty

Juliana Castaneda ¹, Xavier A. Martin ¹, Majsja Ammouriouva ¹, Javier Panadero ¹ and Angel A. Juan ^{2,*}

¹ Computer Science Department, Universitat Oberta de Catalunya, 08018 Barcelona, Spain; jcastanedaji@uoc.edu (J.C.); xmartinso@uoc.edu (X.A.M.); mammouriouva@uoc.edu (M.A.); jpanaderom@uoc.edu (J.P.)

² Department of Applied Statistics and Operations Research, Universitat Politècnica de València, 03801 Alcoy, Spain

* Correspondence: ajuanp@upv.es

Abstract: Stochastic, as well as fuzzy uncertainty, can be found in most real-world systems. Considering both types of uncertainties simultaneously makes optimization problems incredibly challenging. In this paper, we analyze the permutation flow shop problem (PFSP) with both stochastic and fuzzy processing times. The main goal is to find the solution (permutation of jobs) that minimizes the expected makespan. However, due to the existence of uncertainty, other characteristics of the solution are also taken into account. In particular, we illustrate how survival analysis can be employed to enrich the probabilistic information given to decision-makers. To solve the aforementioned optimization problem, we extend the concept of a simheuristic framework so it can also include fuzzy elements. Hence, both stochastic and fuzzy uncertainty are simultaneously incorporated in the PFSP. In order to test our approach, classical PFSP instances have been adapted and extended, so that processing times become either stochastic or fuzzy. The experimental results show the effectiveness of the proposed approach when compared with more traditional ones.



Citation: Castaneda, J.; Martin, X.A.; Ammouriouva, M.; Panadero, J.; Juan, A.A. A Fuzzy Simheuristic for the Permutation Flow Shop Problem under Stochastic and Fuzzy Uncertainty. *Mathematics* **2022**, *10*, 1760. <https://doi.org/10.3390/math10101760>

Academic Editor: Frank Werner

Received: 28 March 2022

Accepted: 18 May 2022

Published: 21 May 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: simulation-optimization; simheuristics; fuzzy techniques; stochastic optimization; permutation flow shop problem

MSC: 68T20; 90-08; 90-10; 90Bxx; 90B36

1. Introduction

In industrial sectors such as manufacturing and production, logistics and transportation, finance and insurance, or smart cities, real-world systems are modeled to be simulated and optimized under different working conditions. These models aim to present the main elements in these systems and their relations [1]. Uncertainty in real-world systems forms one of its main characteristics and should be included in the employed models when making decisions on these systems. Uncertainty could be presented using probabilistic distributions, fuzzy sets, interval models, or convex models [2]. Uncertainty might have several causes, such as lack of information, complexity, ambiguity, conflicting evidence, or subjectivity [3]. Accordingly, researchers employ different approaches to deal with uncertainty in their studies. For instance, random variables and simulation models are utilized to study stochastic models [4].

Since simulation is not the proper tool to optimize systems, researchers integrated simulation within metaheuristics to optimize systems under uncertain conditions. As a result, the simheuristic approach was defined [5]. In this iterative approach, a metaheuristic algorithm constructs solutions that might optimize a real system, and a simulation component is employed to assess these solutions under stochastic conditions and provide feedback to the optimization component. The metaheuristic uses this feedback to improve the search and construct new solutions in subsequent iterations. Examples of simheuristic

applications to solve stochastic problems can be found in Juan et al. [6] and Caldeira and Gnanavelbabu [7]. The simheuristic framework described in Rabe et al. [8] handles systems with moderate stochastic uncertainty. However, as discussed in Zadeh [9], stochastic uncertainty constitutes a special case of uncertainty. Hence, researchers such as Gojković et al. [10] have employed fuzzy uncertainty in their models.

In order to combine both types of uncertainty (stochastic and fuzzy), the simheuristic framework [5] is extended to include fuzzy uncertainty [11]. Like in the simheuristic framework, a deterministic version of a problem is defined and solved using a metaheuristic algorithm. The ‘promising’ solutions found in this initial stage are evaluated considering both a stochastic and fuzzy simulation, and their assessment guides the metaheuristic algorithm to search for new promising solutions. The construction of solutions and their evaluations is repeated until a maximum computing time is reached. Afterward, a list of ‘elite’ solutions is constructed. This list of elite solutions is further examined using more intensive simulations [8].

The permutation flow shop scheduling problem (PFSP) is a manufacturing optimization problem that has attracted many researchers because of its high applicability to real industrial problems [12]. Different researchers have proposed many heuristics and metaheuristics to solve the deterministic version of the PFSP [13]. In real industrial problems, uncertainty characterizes the PFSP. For that reason, we propose a fuzzy simheuristic algorithm to solve this richer variant of the PFSP. Hence, the goal is to find the permutation of jobs that minimizes the expected makespan, taking into account a scenario in which some processing times will be stochastic while others cannot be modeled via random variables and need to be modeled using fuzzy rules provided by experts. To the best of our knowledge, this is the first time a PFSP with both types of uncertainty (stochastic and fuzzy) is addressed in the scientific literature. In effect, although both types of uncertainty have been considered in the past for different PFSP variants, it is not usual to include both of them simultaneously in a PFSP, probably due to the lack of hybrid simulation-optimization-fuzzy methodologies as the one introduced here. Figure 1 shows a flow shop process with three machines (M1, M2, and M3), and four jobs. Notice that the non-deterministic nature of the processing times of job i on machine j , O_{ij} , transforms the makespan associated with a given solution (permutation of jobs) into a random variable, which is likely to take different values every time the solution is put into practice (either in real life or in a simulated environment).

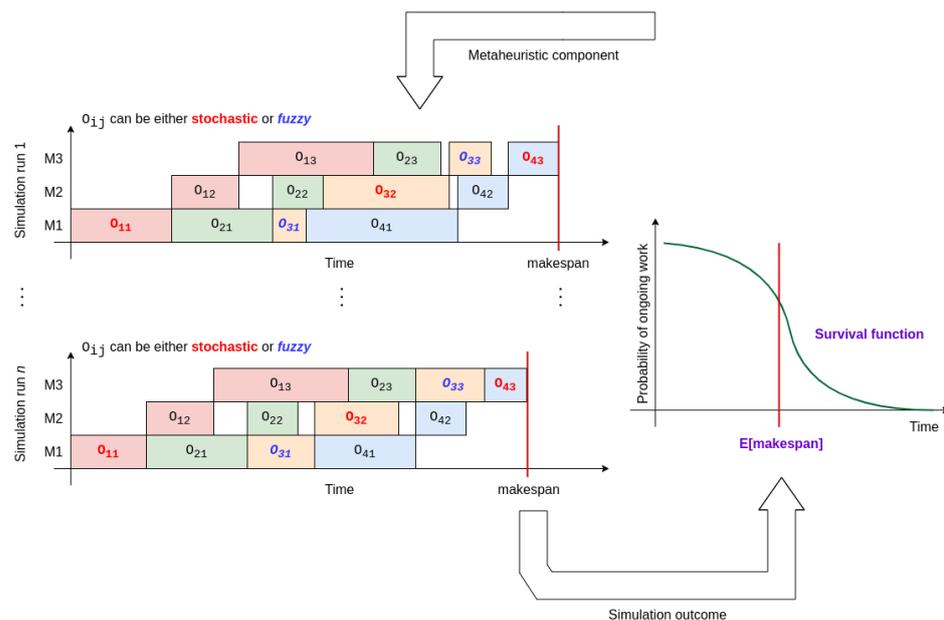


Figure 1. A schematic representation of the main characteristics in our methodology.

Another major contribution of this manuscript is the introduction of statistical reliability concepts to evaluate the PFSP solutions in a scenario under uncertainty. Hence, we discuss how to generate a survival function associated with a given solution from the feedback provided by the simulation component. For each future target time t_0 , this survival function offers the probability that the makespan associated with a given solution exceeds t_0 . The remaining of the paper is structured as follows. First, Section 2 reviews the background concepts on which this work is based, while Section 3 presents some of the related work available in the literature. Our approach to solving the PFSP is described in Section 4. Section 5 provides more details on how the hybrid fuzzy simheuristic is applied to solve the PFSP under a scenario with both stochastic and fuzzy uncertainty. A series of computational experiments are described in Section 6. Section 7 discusses the results, and deterministic solutions under uncertainty are analyzed and compared to solutions found using the proposed fuzzy simheuristic approach. Finally, Section 8 highlights the main outcomes of this work.

2. Background Concepts

This section reviews the background concepts and works related to the methodology presented in this paper, starting with the concepts of fuzzy systems (Section 2.1), and following with simheuristics (Section 2.2).

2.1. Overview of Fuzzy Concepts

Our paper considers uncertainty that forms the main characteristics of real systems. Uncertainty is a broad concept not limited to stochastic uncertainty represented by probability theory [9]. Thus, we also consider the fuzzy theory used to formulate the uncertainty in systems. The fuzzy logic theory is a mathematical theory based on fuzzy sets that represent the degree of truth of a particular result, whether it belongs to a specific category or not. It is not related to the degree of probability that a particular outcome is observed. Instead, this theory was designed to model the uncertainty and vagueness of human cognitive processes [14]. Vagueness is a term that arises when an outcome cannot be adequately observed, color perception being a clear example of vagueness [15].

Fuzzy logic models are characterized by allowing the categorization of things according to a human logic that is based on prototypical definitions or characteristics of studied objects [14]. This categorization defines the membership functions of fuzzy sets. These functions quantify the membership of elements (objects) and a fuzzy set within a closed unit interval [16]. Thus, fuzzy logic allows us to represent linguistic constructions such as ‘very high’, ‘very low’, ‘medium’, ‘high’, and ‘low’ that generate a structure of inference with adequate reasoning. Behind this categorization, it adopts a probability theory to explain the occurrence of an event. This, in turn, allows us to estimate the probability that a given event occurs [17].

Accordingly, a fuzzy inference system (FIS) is defined [17]. The FIS represents the knowledge of an expert in an analyzed topic. This system takes an input value and transforms it into an output category after applying an inference mechanism based on fuzzy sets and fuzzy rules. The fuzzy rules in fuzzy logic might be formulated in the form of *if-then* statements [3]. As pointed out by Kovac et al. [18], fuzzy knowledge bases can be formed from the automatic generation of rules based on previously measured data in a FIS. In general, the knowledge base will always have the same form. For example, in a system with one input and one output, the knowledge base is a set of n rules like $R = \{R_1, R_2, \dots, R_n\}$, where each i -th rule has the following form:

$$\forall i \in \{1, 2, \dots, n\}, \text{ IF } (\text{premise}_i) \text{ THEN } (\text{consequent}_i) \quad (1)$$

In the previous expression, premise_i and consequent_i have the form X is A and Y is B , respectively. A and B are linguistic values defined by fuzzy sets in the numerical ranges X and Y .

An FIS consists of five components [17]: (i) a fuzzification component, which is responsible for input processing; (ii) fuzzy sets, which are the membership functions; (iii) fuzzy rules, which are based on logical operations and *if-then-else* rules as described by Equation (1); (iv) a fuzzy inference unit, which makes decisions based on the fuzzy sets and rules; and (v) a defuzzification component, which takes care of converting the fuzzy sets into real numbers. Figure 2 presents the components of an FIS and their relations.

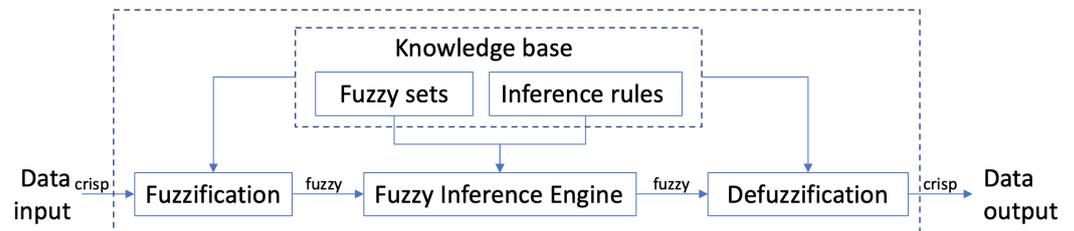


Figure 2. Architecture of a fuzzy inference system.

Whenever the processing time of a job in a machine is uncertain but not stochastic, the selected FIS is used to estimate its value, hence allowing for computing the makespan. The elements of the FIS for this case are described below:

- Fuzzification is transforming the inputs (crisp values) into fuzzy variables. In the considered PFSP, the job type and machine type are selected as input variables.
- Fuzzy sets is the database with the membership functions defining the linguistic labels. In an PFSP instance, the two input variables will have three membership functions: low, medium, and high. Meanwhile, the output variable representing the processing time will have five membership functions: very low, low, medium, high, and very high.
- Inference rules are designed by expert criteria. In the considered problem, we employ rules such as “if a job type is low and a machine type is low, then the processing time is very low”, as well as “if a job type is medium and a machine type is low, or a job type is low and a machine type is medium, then the processing time is low”.
- A fuzzy inference engine applies the inference rules to obtain fuzzy results. The fuzzy sets are combined using logical and relational operators. In the PFSP, the input variables with membership functions are combined with the operators to produce the fuzzy output following the inference rules.
- Finally, defuzzification is the process of transforming the fuzzy results into a crisp value. This transformation can be implemented through different methods, such as the weighted mean, the mean of maximums, or the center of gravity method, which is the one employed in our case.

2.2. Overview of Simheuristics

Simulation is a technique that allows mimicking operations in a system during a time interval, starting from an initial state, depending on input data, and collecting output data about the system [1]. This technique is characterized by its ability to model complex systems with static variables and, accordingly, study their behavior, performance, and reliability [19]. In addition, simulation is used to evaluate different scenarios that a system could face and to foresee possible errors. However, simulation only allows for analyzing a system and not for optimizing its configuration. Many solutions to system optimization problems under stochastic uncertainty have been based on hybrid techniques that combine both tools, optimization, and simulation. These hybrid techniques are known as simulation-optimization approaches and typically involve the optimization of a stochastic objective function subject to possible stochastic constraints [20]. To deepen the understanding of this hybrid methodology, the reader is referred to Amaran et al. [20].

Simheuristics is a simulation-optimization methodology that combines simulation with metaheuristics [5]. One of the seminal works in the area of combining heuristics with simulation is that of Glover et al. [21]. Simheuristics have the potential to generate

several quality solutions starting from well-known or expected values taken initially by the random variables of a problem. Fuzzy simheuristics extend the simheuristic concept in order to solve combinatorial optimization problems with random and fuzzy elements in the objective function or the probabilistic constraints. Hence the type of problems we aim to solve can be expressed as follows:

$$\text{optimize } f(s) = \Phi(s, X, Y) \quad (2)$$

subject to:

$$\Pr(g_i(s, X, Y) \leq l_i(X, Y)) \leq t_i \quad \forall i \in I \quad (3)$$

$$\Pr(g_j(s, X, Y) > l_j(X, Y)) \leq t_j \quad \forall j \in J \quad (4)$$

$$g_k(s) \leq t_k \quad \forall k \in K \quad (5)$$

$$s \in S \quad (6)$$

where: s is an element of the solution space S , X is a vector of random variables, and Y is a vector of fuzzy variables. The objective function in Equation (2), $\Phi(s, X, Y)$, is a stochastic-fuzzy measure to be optimized (minimized or maximized). Equations (3) and (4) present probabilistic constraints. For example, the probability that the time delay in solution s , $g_j(s, X, Y)$, exceeds a certain limit, $l_j(X, Y)$, is within a user-defined threshold. Equation (5) is a typical deterministic constraint.

The ability of simulation to handle uncertainty is used to evaluate the set of different solutions generated in the iterative process of the metaheuristic algorithm. Thus, each event is modeled by the probability distribution that best fits it. Then, results generated by the simulation component are used to provide feedback to the metaheuristic component. This feedback includes information that can be valuable to enhance the search for better stochastic solutions. Finally, when a stopping criterion is satisfied, a reduced set of 'elite' solutions are intensively simulated to ensure accurate and precise estimates. The simulation component also provides dispersion measures that allow us to evaluate the risk of the proposed solutions [5].

3. Related Work

This section offers a brief review of the applications of fuzzy logic and simheuristics to solve optimization problems with uncertainties. It also presents the PFSP, including some of the most recent work on the methodologies that have been used to solve it.

3.1. Fuzzy and Simheuristic Approaches in Optimization with Uncertainty

Many combinatorial optimization problems associated with real-life systems and processes are NP-hard in nature. The PFSP is proved to be NP-hard for instances with three or more machines [22]. Two machine instances, and a special case with three machines, can be solved in polynomial time [23]. In addition, mathematical models associated with real-life applications usually need to consider certain degrees of uncertainty, which imposes additional challenges in their resolution. On the one hand, simheuristics have been developed to address optimization problems with stochastic uncertainty. Applications of simheuristics are very wide, especially in the field of logistics and transportation [24]. Recent works on simheuristics have focused on solving different optimization problems with stochastic uncertainty such as waste collection management [25], stochastic permutation flow shop problems and job shop problems [7], financial problems [26], etc. On the other hand, fuzzy logic-based systems have been utilized in optimization problems with uncertain non-probabilistic variables. The fuzzy logic is a technique used under the criterion of experts and the lack of data to model the probability distribution of variables. In the optimization of production and distribution processes, for example, a robust model of the biodiesel system based on fuzzy logic has been developed, which is efficiently optimized with a particle swarm algorithm to determine the best operating parameters of

the system [27]. A similar approach was implemented for the bio-methanol production process [28]. Khalifehzadeh and Fakhrzad [29] solved a multi-stage production-distribution planning problem with multiple stakeholders. In their problem, production capacity, and customer demand were modeled as fuzzy variables and probability distributions. They used the selective firefly algorithm to solve it. In the transportation sector, the vehicle routing problem has been solved by considering the customer visit as a fuzzy preference index. The uncertainty in customer demand in this type of problem has also been modeled as a fuzzy variable. Tohidifard et al. [30] solved a multi-depot home care routing problem with time windows and fuzzy demands using a genetic algorithm and particle swarm optimization. Bahri et al. [31] developed a solution based on a genetic algorithm for a multi-objective vehicle routing problem with uncertain demands expressed using fuzzy triangular numbers. Methodologies based on fuzzy logic have also been applied in other areas such as the optimization of financial processes [32] and logistics operations [33].

Several problems with stochastic and fuzzy variables have been solved with simheuristic techniques. Hussain et al. [34] minimized the waiting time of electric vehicles at public charging stations modeled as a fuzzy integer linear programming problem combined with a simulator. Oliva et al. [11] proposed the concept of fuzzy simheuristics, and to illustrate its applicability, they solved the team orienteering problem under a reward uncertainty scenario. Similarly, Tordecilla et al. [35] presented another simheuristic based on a multi-start metaheuristic, Monte Carlo simulation, and an FIS to solve the vehicle routing problem and the team orienteering problem with stochastic and fuzzy variables.

3.2. The Stochastic Permutation Flow Shop Scheduling Problem

The PFSP with stochastic processing times has been addressed with hybrid methods such as simheuristics. For example, González-Neira et al. [36] efficiently solved a multi-objective stochastic PFSP with a simheuristic approach based on a tabu search metaheuristic, an evolutionary strategy, and Monte Carlo simulation. For a bi-objective PFSP, González-Neira and Montoya-Torres [37] proposed a simheuristic approach based on a multi-objective greedy randomized adaptive search procedure (GRASP) coupled with Monte Carlo simulation to obtain the expected duration and the expected delay. Similarly, Villarinho et al. [38] analyzed the PFSP with cumulative delivery dates and stochastic processing times. They solved the deterministic version of the problem with a biased-randomized (BR) heuristic, and the stochastic version with a simheuristic approach combining a metaheuristic algorithm that encapsulates the heuristic into a variable neighborhood descent (VND) framework with Monte Carlo simulation. Their results show the ability of simheuristics to efficiently address the impact of stochastic processing times on the production process makespan, which tends to increase as the number of jobs increases.

The fuzzy concept has also been considered in the PFSP to model processing times, capacity, and workload parameters. Optimization methodologies based on efficient algorithms have been proposed to solve different versions of the fuzzy PFSP. Metaheuristics, such as the improved artificial bee colony (ABC) algorithm, have been proposed to solve the PFSP with fuzzy completion time and machine workload [39]. Vela et al. [40] solved the PFSP with uncertain duration and flexible expiration dates modeled as fuzzy variables using tabu search. PFSPs with uncertain processing times were solved with hybrid methods, such as fuzzy ant colony optimization (ACO) incorporated with a fuzzy local optimization algorithm [41]. The PSFP with fuzzy processing times was also solved with an ABC algorithm [42].

Table 1 summarizes some of the recent works in the literature that have addressed the PFSP with deterministic, stochastic, or fuzzy variables, as well as the different solution methodologies. These papers, in general, show a strong tendency to address PFSP with fuzzy variables, which have mostly been solved using different metaheuristics. However, some authors have employed more traditional solution methodologies, such as exact methods, heuristics, and simulation. According to Pan et al. [43], research on efficient algorithms for uncertain scheduling problems has been a point of interest in recent years,

and neglecting uncertainty will limit the practical application of the scheduling results. Therefore, it is important to design efficient optimization methodologies capable of solving problems with different types of uncertainty.

Table 1. Recent work on the PFSP with uncertainty available in the literature.

Authors	Problem Characteristics				Solution Methods				
	Determ.	Stoch.	Fuzzy	Exact	Heuristic	Metaheuristic	Simulation	Simheuristic	Hybrid
González-Neira et al. [36]		•				Tabu search & PAES	•	•	•
González-Neira and Montoya-Torres [37]		•				GRASP	•	•	•
Villarinho et al. [38]	•	•	•		•	VND	•	•	•
Gao et al. [39]			•			ABC			
Vela et al. [40]			•			Tabu search			
Jia et al. [41]			•			ACO			
Emin Baysal et al. [42]			•			ABC			
Pan et al. [43]			•		•	Evolutionary algorithm with adaptation mechanism			
Ouchiekh et al. [44]	•	•				Eagle stratrgy			•
						–			
						Sine-cosine algorithm			
Amirghasemi [45]	•					LND–VNS			
Fathollahi-Fard et al. [46]	•				•				
Parviznejad and Asgharizadeh [47]			•	•					
Gonzalez-Neira et al. [48]		•				GRASP–PAES	•	•	•
de Fátima Morais et al. [49]	•					Discrete differential evolution			
Engin and İşler [50]			•			Parallel greedy algorithm			
Chao [51]		•				Online bees algorithm	•		
Abtahi and Sahraeian [52]		•			•				
Xu et al. [53]			•			Genetic algorithm			

4. Extending the Simheuristic Framework with Fuzzy Techniques

This section describes the fuzzy simheuristic framework we employ in this work (Figure 3). This framework extends the simheuristic approach proposed in Juan et al. [54] to solve the PFSP with stochastic processing times. The extension considers stochastic and fuzzy uncertainty altogether. The fuzzy simheuristic framework includes the following main steps:

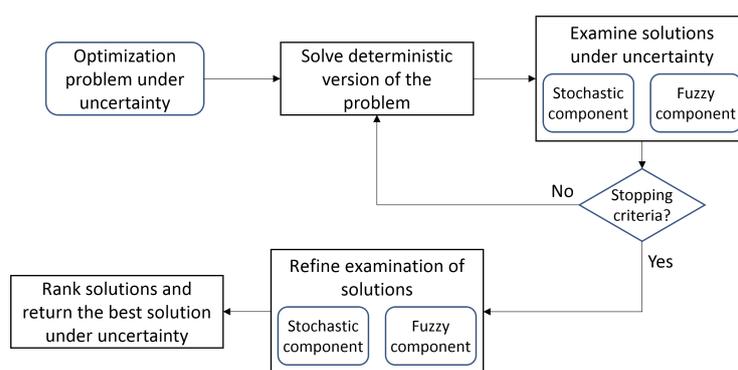


Figure 3. Schema of the fuzzy-simheuristic approach.

1. The deterministic version of the optimization problem is defined. Thus, variables in the optimization problems are replaced by their expected or most likely values. The newly defined problem is deterministic since the uncertainty is removed from it.
2. The deterministic version of the problem is solved using a metaheuristic framework. Several feasible solutions for this version are generated. These solutions are considered good-quality solutions for the deterministic version of the problem.
3. The deterministic good-quality solutions are examined according to the uncertainty elements in the problem. In this examination, a relatively small number of simulation runs are utilized to evaluate the effect of stochastic and fuzzy uncertainty on the objective value of these solutions; a different value is assigned to random variables or fuzzy elements in each run according to the probability distribution or fuzzy rules. In addition, constraints are also evaluated under uncertainty conditions. Finally, descriptive statistics are obtained for each solution, which provides detailed information on them.
4. The examination of solutions guides the metaheuristic generation of new solutions. For example, the evaluation of solutions could improve the local search in the heuristic, or 'good' solutions could form the start to generate new solutions in the subsequent iterations of the metaheuristic. This generation of new solutions continues until a stopping criterion is met, e.g., reaching a maximum allowed time.
5. After terminating the previous stage, the 'elite' solutions are examined further under uncertainty. More runs are utilized to examine solutions' quality intensively in this step compared to the first examination.
6. According to the intensive examination, solutions are ranked, and the best solution is recommended. Identifying the best solution might take into account measures of interest other than the expected value, e.g., the variance or the reliability of each solution.

5. Application to the Stochastic and Fuzzy PFSP

In this paper, we consider a set $J = \{1, 2, \dots, n\}$ of jobs that have to be sequentially processed by a set $M = \{1, 2, \dots, m\}$ of machines. All jobs and machines are available at the starting time, and there is no job preemption. Processing job i on machine j requires a time $O_{i,j}$, which is modeled either as an independent random variable or as a fuzzy function ($\forall i \in J, \forall j \in M$). Given a sequence of jobs $\pi = \{\pi_1, \pi_2, \dots, \pi_n\}$, let job π_i be the job assigned to position i , and $C_{\pi_i,j}$ denote the completion time of job π_i on machine j ($\forall i \in J, \forall j \in M$). The goal is to find a sequence of jobs that minimizes the expected value of the total completion time or makespan, $C_{max} = C_{\pi_n,m}$ [55]. However, since we are considering uncertainty, other statistical characteristics of the proposed solution—such as its variability or reliability—, might also be provided to decision-makers. We present the mathematical model of the PFSP using three terms: decision variables, objective function, and constraints [13]. Regarding decision variables, they can be represented as follows:

$$X_{\pi_i,k} = \begin{cases} 1 & \text{if job } i \text{ is assigned to position } k \text{ in sequence } \pi, \\ 0 & \text{otherwise} \end{cases} \quad \forall i, k \in \{1, 2, \dots, n\} \quad (7)$$

Then, our objective function can be expressed as:

$$\text{Minimize } E[C_{max}] \quad (8)$$

While the following constraints need to be respected:

$$\sum_{k=1}^n X_{\pi_i,k} = 1 \quad \forall i \in \{1, 2, \dots, n\} \tag{9}$$

$$\sum_{i=1}^n X_{\pi_i,k} = 1 \quad \forall k \in \{1, 2, \dots, n\} \tag{10}$$

$$C_{\pi_1,1} \geq \sum_{i=1}^n X_{\pi_i,1} \times O_{\pi_i,1} \tag{11}$$

$$C_{\pi_{k,j}} \geq C_{\pi_{k,j-1}} + \sum_{i=1}^n X_{\pi_i,k} \times O_{\pi_i,j} \quad \forall k \in \{1, 2, \dots, n\}; \forall j \in \{2, 3, \dots, m\} \tag{12}$$

$$C_{\pi_{k,j}} \geq C_{\pi_{k-1,j}} + \sum_{i=1}^n X_{\pi_i,k} \times O_{\pi_i,j} \quad \forall k \in \{2, 3, \dots, n\}; \forall j \in \{1, 2, \dots, m\} \tag{13}$$

$$C_{\pi_{k,j}} \geq 0 \quad \forall k \in \{1, 2, \dots, n\}; \forall j \in \{1, 2, \dots, m\} \tag{14}$$

$$X_{\pi_i,k} \in \{0, 1\} \quad \forall i, k \in \{1, 2, \dots, n\} \tag{15}$$

Hence, our main goal is to minimize the expected makespan, as represented by Equation (8). Equations (9) and (10) guarantee that each position of the sequence will be assigned to only one job, and each job can only occupy one position. Equation (11) defines the completion time of the job at the first position on the first machine. Equation (12) guarantees that, for a job at position k , the completion time on machine j is greater than or equal to its completion time on the previous machine, $j - 1$, plus its processing time on machine j . Equation (13) ensures that the completion time of the job at position k on machine j is greater than or equal to the completion time of the job at position $k - 1$ plus its processing time on the machine j . Equations (14) and (15) ensure that the completion times are positive, and also that the assignment of jobs to the positions of the sequence is binary, respectively. As described in Juan et al. [56], a biased-randomized iterated local search algorithm can be employed to efficiently solve the deterministic version of the PFSP, using Taillard’s acceleration matrices to speed up the computation of the makespan associated with any proposed solution.

6. Details on the Solving Approach and Computational Experiments

To consider stochastic and fuzzy processing times, we have modified the classical set of PFSP instances proposed by Taillard [57], which can be found at <http://mistic.heig-vd.ch/taillard/problemes.dir/ordonnancement.dir/ordonnancement.html> (accessed on 2 March 2022). These instances are grouped in 12 sets of 10 instances each, depending on the number of jobs and machines. For each instance, the nomenclature “ $taia_b_c$ ” is used, where a , b , and c represent the instance number, the number of jobs, and the number of machines, respectively. In the stochastic-fuzzy version of the problem, the log-normal probability distribution is used to model some of the processing times. The log-normal and the Weibull probability distributions are frequently employed to model positive random variables, such as processing or failure times in reliability analysis [58]. For modeling the processing time of job i on machine j , the log-normal distribution parameters are set as follows: the mean is set to the deterministic processing time (the value provided in the original instances), while the variance is set as the product of the mean and a constant $k = 0.5$. For the fuzzy processing times, we have defined a FIS responsible for estimating the total makespan obtained after assigning jobs to different machines. The processing time of each operation is highly uncertain and cannot be modeled using a probability distribution. The elements of the FIS for this case are described below:

- In the considered PFSP, the job type and machine type are selected as input variables for the fuzzification.
- The job type and the machine type are expressed as values ranging from 0 to 1, and are classified as low, medium, and high, following a triangular distribution as

shown in Figure 4. Meanwhile, the output variable represents the processing time and has five membership functions: very low, low, medium, high, and very high. The processing time ranges between 0 and the maximum processing time in a specific instance, and follows a triangular distribution.

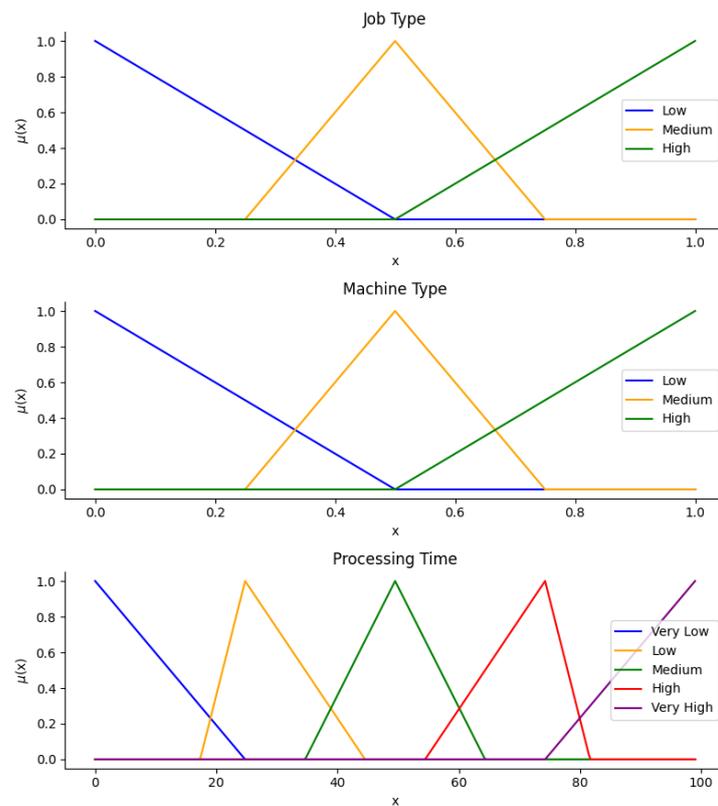


Figure 4. Fuzzy sets of the input and output variables used to generate the fuzzy system.

- Inference rules are designed by expert criteria. Figure 5 shows fuzzy rules in the FIS, where each cell in the grid defines a rule. A total of nine rules are defined. These rules determine the processing time of a job on a machine based on the job and machine types. For example, “if a job type is low and a machine type is low, then the processing time is very low” and “If a job type is medium and a machine type is low, or a job type is low and a machine type is medium, then the processing time is low”.

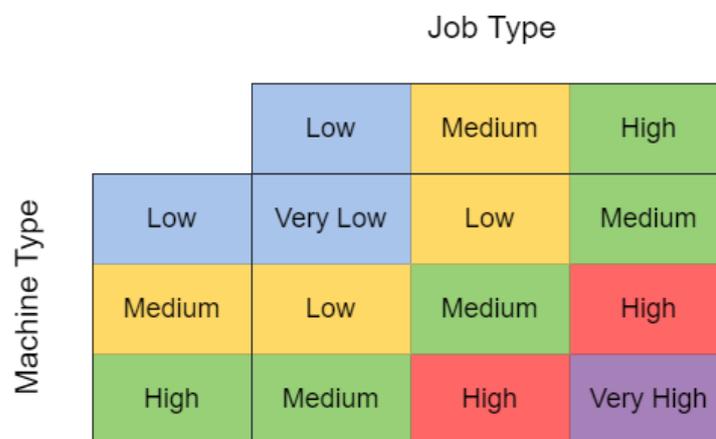


Figure 5. A representation of fuzzy rules used to determine the processing times in the fuzzy system.

- The fuzzy sets are combined using logical and relational operators. In the PFSP, the input variables with membership functions are combined with the operators to produce the fuzzy output following the inference rules.
- In this work, the fuzzification process is performed using the center of gravity method [59].

In our experiments, we utilized Monte Carlo simulation and varied the number of simulation runs in the different phases. To examine solutions during the iterative phase, we set the number of runs to 100 per solution. During the refinement stage (Figure 3), the number of simulation runs was increased up to 1000. The length of the list of elite solutions was set to 3. The fuzzy-simheuristic algorithm has been implemented using Python 3.9 (Python Software Foundation, <https://www.python.org/>, accessed on 2 March 2022), and experiments were run on a standard PC with an Intel i5 CPU at 1.60 GHz and 4 GB RAM. Our approach was tested in the following scenarios:

- Deterministic scenario: In this scenario, no uncertainty is considered. Thus, the processing time of job i on machine j is constant and known in advance. This time is provided in the benchmark dataset. The solution to this scenario is found using a biased-randomized iterated local search (BR-ILS) algorithm [60], which is summarized in Algorithm 1 (in the algorithm, 'cost' refers to the makespan value).
- Stochastic scenario: Processing times of jobs on machines follow log-normal probability distributions. A pure simheuristic approach, similar to the one proposed in Ferone et al. [60], is utilized to find solutions in this scenario.
- Stochastic-fuzzy scenario: This scenario introduces fuzzy uncertainty to half of the processing times. Hence, half of the processing times (randomly selected) are modeled with the log-normal probability distribution, and the remaining processing times are modeled as fuzzy elements, as described in the previous section.
- Completely fuzzy scenario: All processing times are modeled as fuzzy elements in the last scenario. This scenario represents the highest degree of uncertainty in the processing times.

Algorithm 1 BR-ILS metaheuristic for the deterministic PFSP [60]

```

1: baseSol ← biasedRandNEH(inputs)
2: baseSol ← localSearch(baseSol)
3: bestSol ← baseSol
4: while (stopping condition not met) do % Iterated local search
5:   currentSol ← perturbation(baseSol)
6:   currentSol ← localSearch(currentSol)
7:    $\Delta$  ← cost(currentSol) – cost(baseSol) % Acceptance criterion
8:   if ( $\Delta < 0$ ) then % Improvement case
9:     credit ←  $(-\Delta)$ 
10:    baseSol ← currentSol
11:    if (cost(baseSol) < cost(bestSol)) then
12:      bestSol ← baseSol
13:    end if
14:  end if
15:  if ( $0 < \Delta \leq \text{credit}$ ) then % Deterioration case
16:    credit ← 0
17:    baseSol ← currentSol
18:  end if
19: end while
20: return bestSol

```

7. Results and Discussion

Table 2 shows the obtained results. The first column identifies the instances, while the rest columns show the makespan under the four considered scenarios. For each scenario with uncertainty, two solution approaches were used. The BR-ILS columns show

the expected cost obtained when our best deterministic solution is evaluated under a specific uncertainty scenario, with the corresponding level of uncertainty. To determine the values in these columns, we have executed the BR-ILS algorithm (with disabled uncertainty components) followed by the ‘intensive’ simulation process to the best deterministic solution. The main idea behind this process is to assess our best deterministic solution under different levels of uncertainty. The second approach utilizes the above-mentioned fuzzy-simheuristic approach (Sim BR-ILS). This approach considers the uncertainty during the metaheuristic search.

Our best deterministic solutions (Column 2) are compared with the best-known solution (BKS) in the literature (Column 1), which allows us to assess the quality of our deterministic algorithm. Notice that the BR-ILS algorithm employed to solve the deterministic version of the PFSP is highly competitive: in a few seconds of computation per instance, it obtains an average gap of 0.55% with respect to the BKS. In the second section of the table, we present the obtained solutions considering the stochastic scenario. Column 3 displays the expected makespan when the solutions found using the BR-ILS approach are simulated under the stochastic scenario. Similarly, the next column shows the results obtained using our Sim BR-ILS approach. The next section of the table offers the obtained solutions for the fuzzy-stochastic scenario. Column 5 reports the results when the BR-ILS solutions are simulated under a fuzzy-stochastic scenario. Moreover, the following column shows the expected makespan obtained using the fuzzy-simheuristic BR-ILS approach. Similarly, the last section of the table exhibits the solutions obtained for the completely fuzzy scenario.

Figure 6 depicts an overview of Table 2, where the horizontal and vertical axes represent the four uncertainty scenarios and the percentage gap with respect to the BKS, respectively. The deterministic scenario can be considered as an ideal scenario with perfect information, which is not the case in scenarios with stochastic or fuzzy processing times. Concerning the uncertainty scenarios, the computed gaps show that the simheuristic BR-ILS outperforms the BR-ILS algorithm when uncertainty is introduced, i.e., using the BR-ILS approach for the uncertainty scenarios leads to sub-optimal solutions. This result proves the importance of integrating fuzzy-simulation methods when dealing with optimization problems with uncertainty. Another notice is that the expected makespan increases due to the uncertainty of the processing times and, consequently, increases the completion time of jobs being processed by the machines. Thus, the search for solutions that minimize the expected makespan by considering the fuzzy or stochastic processing times gives better results than only considering the makespan given by the deterministic processing times.

As mentioned before, when working under uncertain conditions, other aspects of the solution should also be taken into account. Thus, for instance, managers might be interested in analyzing the reliability or survival function associated with a given permutation of jobs, i.e., the probability that the random makespan exceeds a given target time. To illustrate these concepts, the following analysis has been conducted for the *tai002_20_5* instance. Monte Carlo simulation has been used to generate 1000 random observations of the makespan associated with the proposed solution. As illustrated in Figure 7, this makespan is a random variable that can be fit by a log-normal probability distribution with estimated parameters: location = 7.22332 and scale = 0.0162057 (these estimated parameters were obtained using the maximum likelihood method, with an Anderson-Darling statistic = 0.626).

Based on the fitted probability distribution, Figure 8 shows the ‘survival’ function associated with the proposed solution. This function also includes the 95% confidence intervals. Notice that the probability that the proposed solution offers a makespan longer than 1356 time units is approximately 0.75. Similarly, the probability that it offers a makespan above 1371.2 time units is approximately 0.50. Finally, the probability that the makespan reaches a value above 1,386 time units is approximately 0.25 (i.e., 75% of the times the proposed solution will provide a makespan lower than 1386 time units).

Table 2. Results of the the four considered scenarios of the PFSP.

Instance	Deterministic Scenario			Stochastic Scenario		Stoch-Fuzzy Scenario		Fuzzy Scenario	
	BKS	BR-ILS	GAP(%)	BR-ILS	Sim.	BR-ILS	Sim.	BR-ILS	Sim.
	(1)	(2)	(1-2)	(3)	BR-ILS (4)	(5)	BR-ILS (6)	(7)	BR-ILS (8)
tai002_20_5	1359	1359	0.00	1371.38	1369.77	1553.77	1548.30	1552.51	1552.51
tai010_20_5	1108	1108	0.00	1133.81	1129.65	1390.46	1338.65	1560.28	1520.87
tai017_20_10	1484	1484	0.00	1517.35	1505.46	1757.69	1623.89	1994.44	1874.61
tai018_20_10	1538	1538	0.00	1576.16	1575.23	1837.13	1708.41	1936.43	1778.29
tai020_20_10	1591	1591	0.00	1636.77	1634.26	1772.16	1730.28	1876.22	1796.11
tai031_50_5	2724	2724	0.00	2746.91	2742.10	3126.52	3112.56	3693.19	3650.43
tai034_50_5	2751	2751	0.00	2797.29	2785.58	3336.54	3336.26	3732.11	3732.11
tai035_50_5	2863	2863	0.00	2888.09	2876.81	3369.97	3368.28	3732.34	3732.34
tai036_50_5	2829	2829	0.00	2865.12	2861.57	3324.02	3313.48	3732.38	3610.18
tai039_50_5	2552	2552	0.00	2589.39	2586.07	3267.29	3194.62	3651.99	3609.64
tai040_50_5	2782	2782	0.00	2814.79	2798.58	3260.32	3228.04	3669.33	3611.01
tai041_50_10	2991	3037	1.54	3103.37	3098.23	3765.06	3517.88	4021.97	3882.23
tai042_50_10	2867	2923	1.95	2993.04	2991.07	3545.02	3461.89	4040.56	4025.21
tai046_50_10	3006	3031	0.83	3100.32	3098.08	3571.18	3443.81	3996.58	3799.48
tai049_50_10	2897	2924	0.93	2986.59	2974.89	3646.91	3466.14	4078.46	4003.57
tai053_50_20	3591	3640	1.36	3714.60	3709.35	4319.35	4310.47	4444.99	4442.28
tai069_100_5	5448	5448	0.00	5492.12	5487.66	6689.11	6576.69	7183.16	7138.34
tai073_100_10	5676	5689	0.23	5774.57	5768.57	6919.97	6739.91	7512.29	7406.61
tai074_100_10	5781	5832	0.88	5939.19	5922.87	6747.73	6709.67	7470.61	7430.16
tai076_100_10	5303	5321	0.34	5402.84	5394.99	6685.58	6609.32	7344.73	7333.26
tai077_100_10	5595	5621	0.46	5707.56	5704.90	6404.91	6393.98	7551.84	7551.84
tai081_100_20	6106	6202	1.57	6297.16	6262.59	7154.82	7035.24	8276.60	8213.53
tai085_100_20	6262	6314	0.83	6406.03	6394.65	7203.21	7142.67	8465.49	8403.16
tai093_200_10	10,922	11,045	1.13	11,115.86	11,102.08	12,984.46	12,945.09	13,954.32	13,823.54
tai103_200_20	11,281	11,411	1.15	11,544.97	11,524.49	13,334.12	13,281.89	14,265.83	14,215.64
tai105_200_20	11,259	11,340	0.72	11,392.55	11,383.12	13,204.56	13,123.23	14,166.19	14,068.98
tai107_200_20	11,337	11,360	0.20	11,437.28	11,423.71	13,273.78	13,198.54	14,198.64	14,164.35
tai108_200_20	11,301	11,455	1.36	11,494.95	11,489.36	13,318.82	13,234.61	14,401.19	14,364.54
tai112_500_20	26,500	26,702	0.76	26,796.92	26,792.43	30,693.94	30,605.16	32,612.91	32,544.43
tai118_500_20	26,560	26,654	0.35	26,752.67	26,745.54	30,543.56	30,458.25	32,593.75	32,468.16
Average:	6275.47	6317.67	0.55	6379.66	6371.12	7400.07	7325.24	8057.04	7991.58

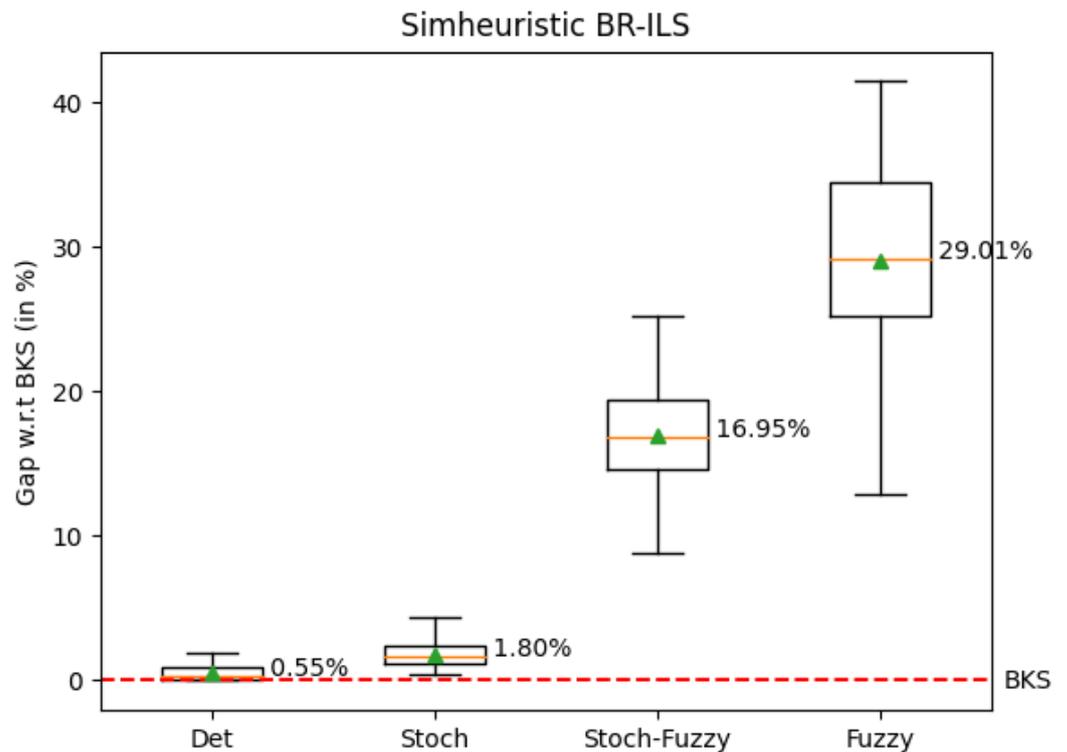


Figure 6. Percentage gaps of our SimBR-ILS approach w.r.t. deterministic BKS (lowerbound).

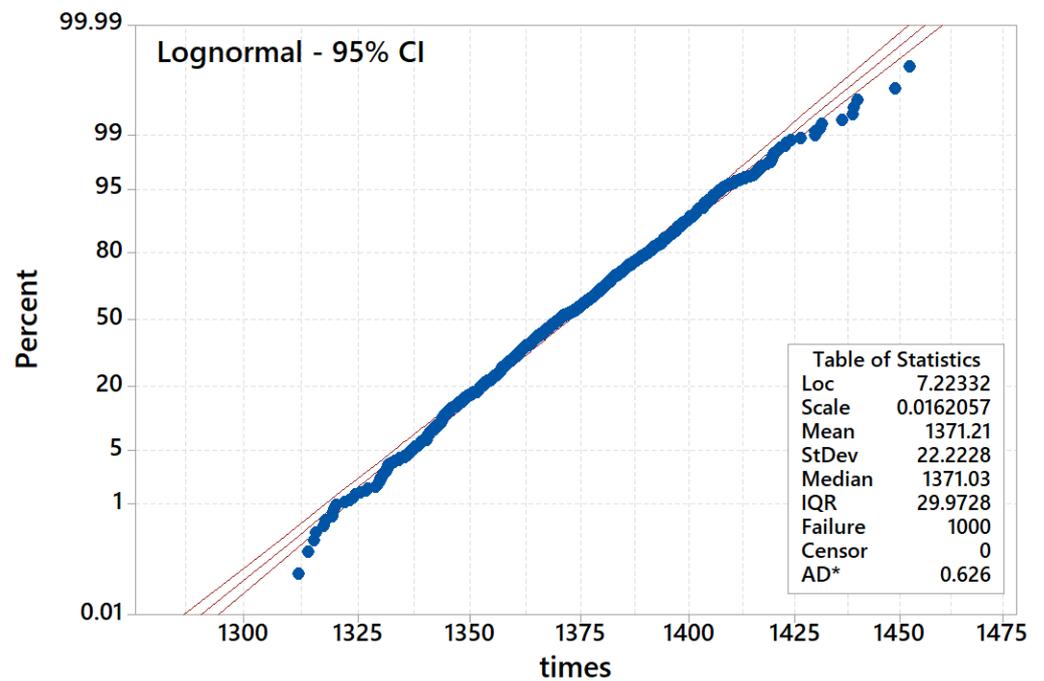


Figure 7. Fitting the makespan observations obtained via simulation by a log-normal distribution.

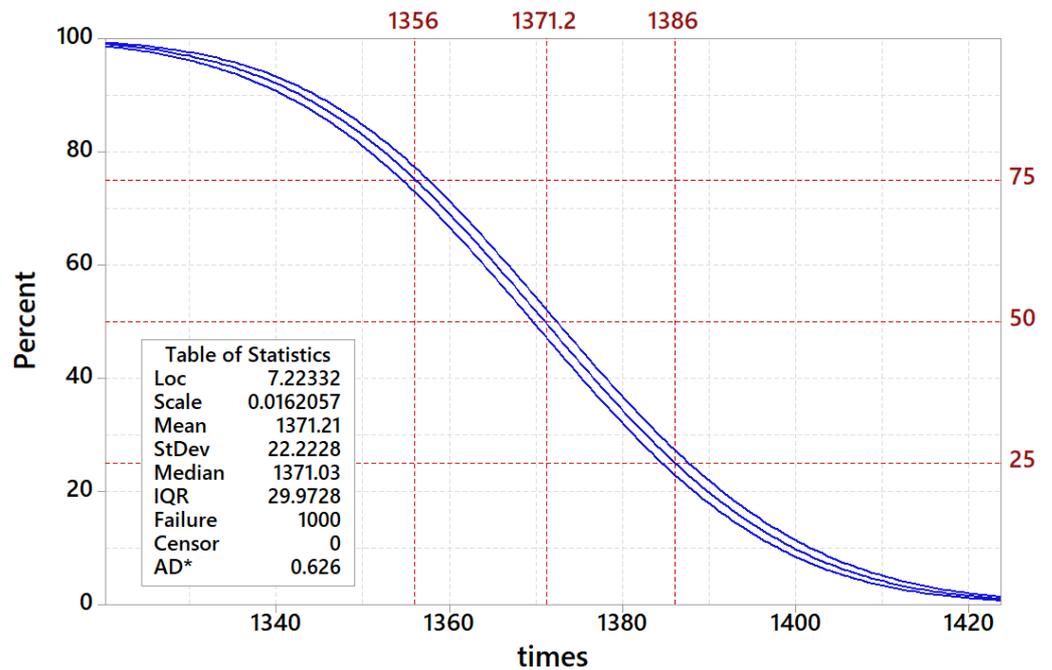


Figure 8. Survival function associated with the proposed solution (includes 95% confidence intervals).

The obtained results illustrate the flexibility of the presented methodology to solve the PFSP under uncertainty scenarios, both probabilistic and fuzzy. This is something that has not been fully explored in the existing literature despite being present in many real-life flow shop systems. Considering stochastic and fuzzy uncertainty is necessary when a subset of elements can be modeled using probability distributions while another subset requires fuzzy techniques –due, for instance, to the lack of enough historical data.

8. Conclusions

This paper analyzes the permutation flow shop problem under both stochastic and fuzzy uncertainty, which is the case in some real-life systems. In order to solve this challenging optimization problem, we propose an extended simheuristic algorithm, which also includes a fuzzy component. The definition of the fuzzy inference system enriches and extends the simulation-optimization approach. Fuzzy processing times are modeled based on a job and machine type, and, accordingly, the processing time is determined in a fuzzy inference system. The approach has been tested on several PFSP instances, and the results have shown that the approach can handle the problem effectively. Hence, our fuzzy simheuristic algorithm constitutes a general approach that can be employed to solve different PFSP variants regardless of the type of uncertainty being considered, and even those simultaneously combining stochastic and fuzzy uncertainties.

Moreover, the paper discusses the relevance of considering statistical metrics, other than the expected makespan, when dealing with non-deterministic versions of flow shop problems. In particular, the manuscript illustrates how reliability concepts can be employed to provide probabilistic information on the proposed solutions. Thus, the feedback provided by the simulation component is utilized to fit the probabilistic behavior of the proposed solution by a theoretical probability distribution (whenever possible) or by an empirical one. This, in turn, allows us to plot the survival function associated with any given solution (permutation of jobs in this case), and obtain the probability that it can be completed before any given target time.

Some open research lines are highlighted next: (i) the incorporation of machine learning methods inside the fuzzy simheuristic framework to enhance the utility of the feedback provided by the simulation component, which can be utilized to identify new ‘promising’ solutions as well as to develop surrogate models that contribute to speed up computations; and (ii) test other fuzzy components based on type-2 fuzzy sets, which can also reduce computational times even further.

Author Contributions: Conceptualization, A.A.J. and J.P.; methodology, all authors; software, X.A.M., J.P. and J.C.; validation, all authors; writing—original draft preparation, J.C. and M.A.; writing—review and editing, all authors; supervision, M.A., J.P. and A.A.J. All authors have read and agreed to the published version of the manuscript.

Funding: This work has been partially supported by the Spanish Ministry of Science (PID2019-111100RB-C21/AEI/10.13039/501100011033), as well as by the Barcelona Council and the “la Caixa” Foundation under the framework of the Barcelona Science Plan 2020–2023 (grant 21S09355-001).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

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

References

1. Law, A.M. *Simulation Modeling and Analysis*, 5th ed.; McGraw-Hill: New York, NY, USA, 2015.
2. Wazed, M.; Ahmed, S.; Nukman, Y. Uncertainty factors in real manufacturing environment. *Aust. J. Basic Appl. Sci.* **2009**, *3*, 342–351.
3. Zimmermann, H.J. *Fuzzy Set Theory and Its Applications*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2011.
4. Liu, R.; Xie, X.; Yu, K.; Hu, Q. A survey on simulation optimization for the manufacturing system operation. *Int. J. Model. Simul.* **2018**, *38*, 116–127. [[CrossRef](#)]
5. Chica, M.; Juan, A.A.; Bayliss, C.; Cordon, O.; Kelton, W.D. Why simheuristics? Benefits, limitations, and best practices when combining metaheuristics with simulation. *SORT-Stat. Oper. Res. Trans.* **2020**, *44*, 1–24. [[CrossRef](#)]
6. Juan, A.A.; Keenan, P.; Martí, R.; McGarraghy, S.; Panadero, J.; Carroll, P.; Oliva, D. A review of the role of heuristics in stochastic optimisation: From metaheuristics to learnheuristics. *Ann. Oper. Res.* **2021**, 1–31. [[CrossRef](#)]
7. Caldeira, R.H.; Gnanavelbabu, A. A simheuristic approach for the flexible job shop scheduling problem with stochastic processing times. *Simulation* **2021**, *97*, 215–236. [[CrossRef](#)]

8. Rabe, M.; Deininger, M.; Juan, A. Speeding up computational times in simheuristics combining genetic algorithms with discrete-event simulation. *Simul. Model. Pract. Theory* **2020**, *103*, 102089. [[CrossRef](#)]
9. Zadeh, L.A. Toward a generalized theory of uncertainty (GTU)—an outline. *Inf. Sci.* **2005**, *172*, 1–40. [[CrossRef](#)]
10. Gojković, R.; Đurić, G.; Tadić, D.; Nestić, S.; Aleksić, A. Evaluation and selection of the quality methods for manufacturing process reliability improvement—Intuitionistic fuzzy sets and genetic algorithm approach. *Mathematics* **2021**, *9*, 1531. [[CrossRef](#)]
11. Oliva, D.; Copado, P.; Hinojosa, S.; Panadero, J.; Riera, D.; Juan, A.A. Fuzzy simheuristics: Solving optimization problems under stochastic and uncertainty scenarios. *Mathematics* **2020**, *8*, 2240. [[CrossRef](#)]
12. Yenisey, M.M.; Yagmahan, B. Multi-objective permutation flow shop scheduling problem: Literature review, classification and current trends. *Omega* **2014**, *45*, 119–135. [[CrossRef](#)]
13. Zaied, A.N.H.; Ismail, M.M.; Mohamed, S.S. Permutation flow shop scheduling problem with makespan criterion: Literature review. *J. Theor. Appl. Inf. Technol.* **2021**, *99*, 830–848.
14. Lootsma, F.A. *Fuzzy Logic for Planning and Decision Making*; Springer Science & Business Media: Berlin/Heidelberg, Germany, 2013; Volume 8.
15. Bustince, H.; Herrera, F.; Montero, J. *Fuzzy Sets and Their Extensions: Representation, Aggregation and Models: Intelligent Systems from Decision Making to Data Mining, Web Intelligence and Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2007; Volume 220.
16. Celikyilmaz, A.; Türksen, I.B. Fuzzy Sets and Systems. In *Modeling Uncertainty with Fuzzy Logic: With Recent Theory and Applications*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 11–50.
17. Sabri, N.; Aljunid, S.; Salim, M.; Badlishah, R.; Kamaruddin, R.; Malek, M. Fuzzy inference system: Short review and design. *Int. Rev. Autom. Control* **2013**, *6*, 441–449.
18. Kovac, P.; Rodic, D.; Pucovsky, V.; Savkovic, B.; Gostimirovic, M. Multi-output fuzzy inference system for modeling cutting temperature and tool life in face milling. *J. Mech. Sci. Technol.* **2014**, *28*, 4247–4256. [[CrossRef](#)]
19. Faulin, J.; Juan, A.A.; Serrat, C.; Bargueno, V. Predicting availability functions in time-dependent complex systems with SAEDS simulation algorithms. *Reliab. Eng. Syst. Saf.* **2008**, *93*, 1761–1771. [[CrossRef](#)]
20. Amaran, S.; Sahinidis, N.V.; Sharda, B.; Bury, S.J. Simulation optimization: A review of algorithms and applications. *Ann. Oper. Res.* **2016**, *240*, 351–380. [[CrossRef](#)]
21. Glover, F.; Kelly, J.P.; Laguna, M. New advances and applications of combining simulation and optimization. In Proceedings of the 1996 Winter Simulation Conference, Coronado, CA, USA, 8–11 December 1996; pp. 144–152.
22. Garey, M.R.; Johnson, D.S.; Sethi, R. The complexity of flowshop and jobshop scheduling. *Math. Oper. Res.* **1976**, *1*, 117–129. [[CrossRef](#)]
23. Johnson, S.M. Optimal two-and three-stage production schedules with setup times included. *Nav. Res. Logist. Q.* **1954**, *1*, 61–68. [[CrossRef](#)]
24. Onggo, B.S.; Panadero, J.; Corlu, C.G.; Juan, A.A. Agri-food supply chains with stochastic demands: A multi-period inventory routing problem with perishable products. *Simul. Model. Pract. Theory* **2019**, *97*, 101970. [[CrossRef](#)]
25. Gruler, A.; Quintero-Araújo, C.L.; Calvet, L.; Juan, A.A. Waste collection under uncertainty: A simheuristic based on variable neighbourhood search. *Eur. J. Ind. Eng.* **2017**, *11*, 228–255. [[CrossRef](#)]
26. Panadero, J.; Doering, J.; Kizys, R.; Juan, A.A.; Fito, A. A variable neighborhood search simheuristic for project portfolio selection under uncertainty. *J. Heuristics* **2020**, *26*, 353–375. [[CrossRef](#)]
27. Nassef, A.M.; Sayed, E.T.; Rezk, H.; Abdelkareem, M.A.; Rodriguez, C.; Olabi, A. Fuzzy-modeling with particle swarm optimization for enhancing the production of biodiesel from microalga. *Energy Sources Part A Recover. Util. Environ. Eff.* **2019**, *41*, 2094–2103. [[CrossRef](#)]
28. Yousef, B.A.; Rezk, H.; Abdelkareem, M.A.; Olabi, A.G.; Nassef, A.M. Fuzzy modeling and particle swarm optimization for determining the optimal operating parameters to enhance the bio-methanol production from sugar cane bagasse. *Int. J. Energy Res.* **2020**, *44*, 8964–8973. [[CrossRef](#)]
29. Khalifehzadeh, S.; Fakhrzad, M.B. A modified firefly algorithm for optimizing a multi stage supply chain network with stochastic demand and fuzzy production capacity. *Comput. Ind. Eng.* **2019**, *133*, 42–56. [[CrossRef](#)]
30. Tohidifard, M.; Tavakkoli-Moghaddam, R.; Navazi, F.; Partovi, M. A multi-depot home care routing problem with time windows and fuzzy demands solving by particle swarm optimization and genetic algorithm. *IFAC-PapersOnLine* **2018**, *51*, 358–363. [[CrossRef](#)]
31. Bahri, O.; Talbi, E.G.; Amor, N.B. A generic fuzzy approach for multi-objective optimization under uncertainty. *Swarm Evol. Comput.* **2018**, *40*, 166–183. [[CrossRef](#)]
32. Chen, W.; Xu, W. A hybrid multiobjective bat algorithm for fuzzy portfolio optimization with real-world constraints. *Int. J. Fuzzy Syst.* **2019**, *21*, 291–307. [[CrossRef](#)]
33. Tozanli, O.; Duman, G.M.; Kongar, E.; Gupta, S.M. Environmentally concerned logistics operations in fuzzy environment: A literature survey. *Logistics* **2017**, *1*, 4. [[CrossRef](#)]
34. Hussain, S.; Kim, Y.S.; Thakur, S.; Breslin, J.G. Optimization of Waiting Time for Electric Vehicles Using a Fuzzy Inference System. *IEEE Trans. Intell. Transp. Syst.* **2022**, 1–12. [[CrossRef](#)]
35. Tordecilla, R.D.; Martins, L.d.C.; Panadero, J.; Copado, P.J.; Perez-Bernabeu, E.; Juan, A.A. Fuzzy simheuristics for optimizing transportation systems: Dealing with stochastic and fuzzy uncertainty. *Appl. Sci.* **2021**, *11*, 7950. [[CrossRef](#)]

36. González-Neira, E.M.; Urrego-Torres, A.M.; Cruz-Riveros, A.M.; Henao-García, C.; Montoya-Torres, J.R.; Molina-Sánchez, L.P.; Jimenez, J.F. Robust solutions in multi-objective stochastic permutation flow shop problem. *Comput. Ind. Eng.* **2019**, *137*, 106026. [[CrossRef](#)]
37. González-Neira, E.; Montoya-Torres, J. A simheuristic for bi-objective stochastic permutation flow shop scheduling problem. *J. Proj. Manag.* **2019**, *4*, 57–80. [[CrossRef](#)]
38. Villarinho, P.A.; Panadero, J.; Pessoa, L.S.; Juan, A.A.; Oliveira, F.L.C. A simheuristic algorithm for the stochastic permutation flow-shop problem with delivery dates and cumulative payoffs. *Int. Trans. Oper. Res.* **2020**, *28*, 716–737. [[CrossRef](#)]
39. Gao, K.Z.; Suganthan, P.N.; Pan, Q.K.; Chua, T.J.; Chong, C.S.; Cai, T.X. An improved artificial bee colony algorithm for flexible job-shop scheduling problem with fuzzy processing time. *Expert Syst. Appl.* **2016**, *65*, 52–67. [[CrossRef](#)]
40. Vela, C.R.; Afsar, S.; Palacios, J.J.; González-Rodríguez, I.; Puente, J. Evolutionary tabu search for flexible due-date satisfaction in fuzzy job shop scheduling. *Comput. Oper. Res.* **2020**, *119*, 104931. [[CrossRef](#)]
41. Jia, Z.; Yan, J.; Leung, J.Y.; Li, K.; Chen, H. Ant colony optimization algorithm for scheduling jobs with fuzzy processing time on parallel batch machines with different capacities. *Appl. Soft Comput.* **2019**, *75*, 548–561. [[CrossRef](#)]
42. Emin Baysal, M.; Sarucan, A.; Büyükoçkan, K.; Engin, O. Artificial bee colony algorithm for solving multi-objective distributed fuzzy permutation flow shop problem. *J. Intell. Fuzzy Syst.* **2022**, *42*, 1–11.
43. Pan, Z.X.; Wang, L.; Chen, J.F.; Wu, Y.T. A novel evolutionary algorithm with adaptation mechanism for fuzzy permutation flow-shop scheduling. In Proceedings of the 2021 IEEE Congress on Evolutionary Computation (CEC), Kraków, Poland, 28 June–1 July 2021; pp. 367–374.
44. Ouchiekh, R.; Fri, M.; Touil, A.; Echchatbi, A. Total Weighted Tardiness in the Permutation Flow Shop under Uncertainty. *IFAC-PapersOnLine* **2021**, *54*, 1174–1180. [[CrossRef](#)]
45. Amirghasemi, M. An Effective Decomposition-Based Stochastic Algorithm for Solving the Permutation Flow-Shop Scheduling Problem. *Algorithms* **2021**, *14*, 112. [[CrossRef](#)]
46. Fathollahi-Fard, A.M.; Woodward, L.; Akhrif, O. Sustainable distributed permutation flow-shop scheduling model based on a triple bottom line concept. *J. Ind. Inf. Integr.* **2021**, *24*, 100233. [[CrossRef](#)]
47. Parviznejad, P.S.; Asgharizadeh, E. Modeling and Solving Flow Shop Scheduling Problem Considering Worker Resource. *Int. J. Innov. Eng.* **2021**, *1*, 1–17.
48. Gonzalez-Neira, E.M.; Montoya-Torres, J.R.; Jimenez, J.F. A Multicriteria Simheuristic Approach for Solving a Stochastic Permutation Flow Shop Scheduling Problem. *Algorithms* **2021**, *14*, 210. [[CrossRef](#)]
49. de Fátima Morais, M.; Ribeiro, M.H.D.M.; da Silva, R.G.; Mariani, V.C.; dos Santos Coelho, L. Discrete differential evolution metaheuristics for permutation flow shop scheduling problems. *Comput. Ind. Eng.* **2022**, *166*, 107956. [[CrossRef](#)]
50. Engin, O.; İşler, M. An Efficient Parallel Greedy Algorithm for Fuzzy Hybrid Flow Shop Scheduling with Setup Time and Lot Size: A Case Study in Apparel Process. *J. Fuzzy Ext. Appl.* **2021**. [[CrossRef](#)]
51. Chao, W. Using Online Bees Algorithm for Real-time Permutation Flow Shop Problem in Car Disassembly Line. *Res. Sq.* **2021**. [[CrossRef](#)]
52. Abtahi, Z.; Sahraeian, R. Robust and Stable Flow Shop Scheduling Problem Under Uncertain Processing Times and Machines Disruption. *Int. J. Eng. Trans. A Basics* **2021**, *34*, 935–947.
53. Xu, W.J.; He, L.J.; Zhu, G.Y. Many-objective flow shop scheduling optimisation with genetic algorithm based on fuzzy sets. *Int. J. Prod. Res.* **2021**, *59*, 702–726. [[CrossRef](#)]
54. Juan, A.A.; Barrios, B.B.; Vallada, E.; Riera, D.; Jorba, J. A simheuristic algorithm for solving the permutation flow shop problem with stochastic processing times. *Simul. Model. Pract. Theory* **2014**, *46*, 101–117. [[CrossRef](#)]
55. Framinan, J.M.; Gupta, J.N.; Leisten, R. A review and classification of heuristics for permutation flow-shop scheduling with makespan objective. *J. Oper. Res. Soc.* **2004**, *55*, 1243–1255. [[CrossRef](#)]
56. Juan, A.A.; Lourenço, H.R.; Mateo, M.; Luo, R.; Castella, Q. Using iterated local search for solving the flow-shop problem: Parallelization, parametrization, and randomization issues. *Int. Trans. Oper. Res.* **2014**, *21*, 103–126. [[CrossRef](#)]
57. Taillard, E. Benchmarks for basic scheduling problems. *Eur. J. Oper. Res.* **1993**, *64*, 278–285. [[CrossRef](#)]
58. Kim, J.S.; Yum, B.J. Selection between Weibull and lognormal distributions: A comparative simulation study. *Comput. Stat. Data Anal.* **2008**, *53*, 477–485. [[CrossRef](#)]
59. Bai, Y.; Wang, D. Fundamentals of fuzzy logic control—Fuzzy sets, fuzzy Rules and defuzzifications. In *Advanced Fuzzy Logic Technologies in Industrial Applications*; Springer: Berlin/Heidelberg, Germany, 2006; pp. 17–36.
60. Ferone, D.; Hatami, S.; González-Neira, E.M.; Juan, A.A.; Festa, P. A biased-randomized iterated local search for the distributed assembly permutation flow-shop problem. *Int. Trans. Oper. Res.* **2020**, *27*, 1368–1391. [[CrossRef](#)]