Scientia Iranica D (2023) 30(6), 2098–2115

SCIENTIA IRANICA

Sharif University of Technology
**Scientia Iranica**
*Transactions D: Computer Science & Engineering and Electrical Engineering*
http://scientiairanica.sharif.edu

# The role of bi-level uncertain architecture inward smart manufacturing: Process orchestration

**S. Saraeian[a,*] and B. Shirazi[b]**

a. *Department of Computer Engineering, Gorgan Branch, Islamic Azad University, Gorgan, Iran.*
b. *Department of Industrial Engineering, Mazandaran University of Science and Technology, Babol, Iran.*

Received 14 May 2022; received in revised form 28 March 2023; accepted 9 May 2023

**KEYWORDS**
uBPMS architecture;
Smart manufacturing process;
Bi-level optimization;
Uncertainty;
Orchestration.

**Abstract.** Smart manufacturing in the context of a smart factory is allowed through different uncertain processes, which creates significant challenges. In this case, smart manufacturing should be applied reliably, interoperably, and consistently. Thus, it faces the requirement of orchestrating services provided by uncertain processes to satisfy the challenges. These uncertain processes are commonly managed by an uncertain Business Process Management System (uBPMS), which is specifically designed to address unknown conditions. The current uBPMS architecture does not consider business process orchestration, and the objective of this paper is to achieve an extension of uBPMS architecture with a business process orchestration feature to make a response in real time and satisfy the uncertainty conditions in a smart factory. The proposed extension can perform autonomous orchestration of business processes inward traditional uBPMS architecture based on desired values of different objectives optimization. This new architecture operates based on a robust bi-level optimization approach. The Rousselot smart factory in Belgium as a simulated case study was studied. The results show the robustness of the new architecture for process orchestration design in this case. Also, uncertain business process management based on the process orchestration feature presents efficiency and accuracy improvement in smart manufacturing systems.



## 1. Introduction

A smart factory as digitized manufacturing is a good chance to make new formats of flexibility and productivity. It causes increases in mobile device usage in different operations and data-driven sensors. Smart manufacturing in the context of a smart factory is permitted through different processes, big data analytics, intra-machine connectivity, cloud computing, and the Industrial Internet of Things (IIoTs) [1–4]. Also, various elements in such environments, like different complex processes, complicated constraints, and uncertainties, generate big challenges [5] (Figure 1). In this case, smart manufacturing inside different systems and devices should be applied reliably, interoperably, and consistently. Therefore, smart manufacturing encounters the need for orchestrating services provided by business processes to satisfy the challenges.

Business process orchestration refers to managing the entire business process lifecycle, including development, testing, and monitoring from a single location. The produced business process orchestration can satisfy the flexibility and responsibility for managing uncertainties and minimizing process errors [4,6]. Business processes are generally applied for directing high-level organizational and digital processes in smart

*. *Corresponding author.*
*E-mail addresses: shideh.saraeian@iau.ac.ir (S. Saraeian); shirazi_b@icloud.com (B. Shirazi)*

doi: 10.24200/sci.2023.60384.6768

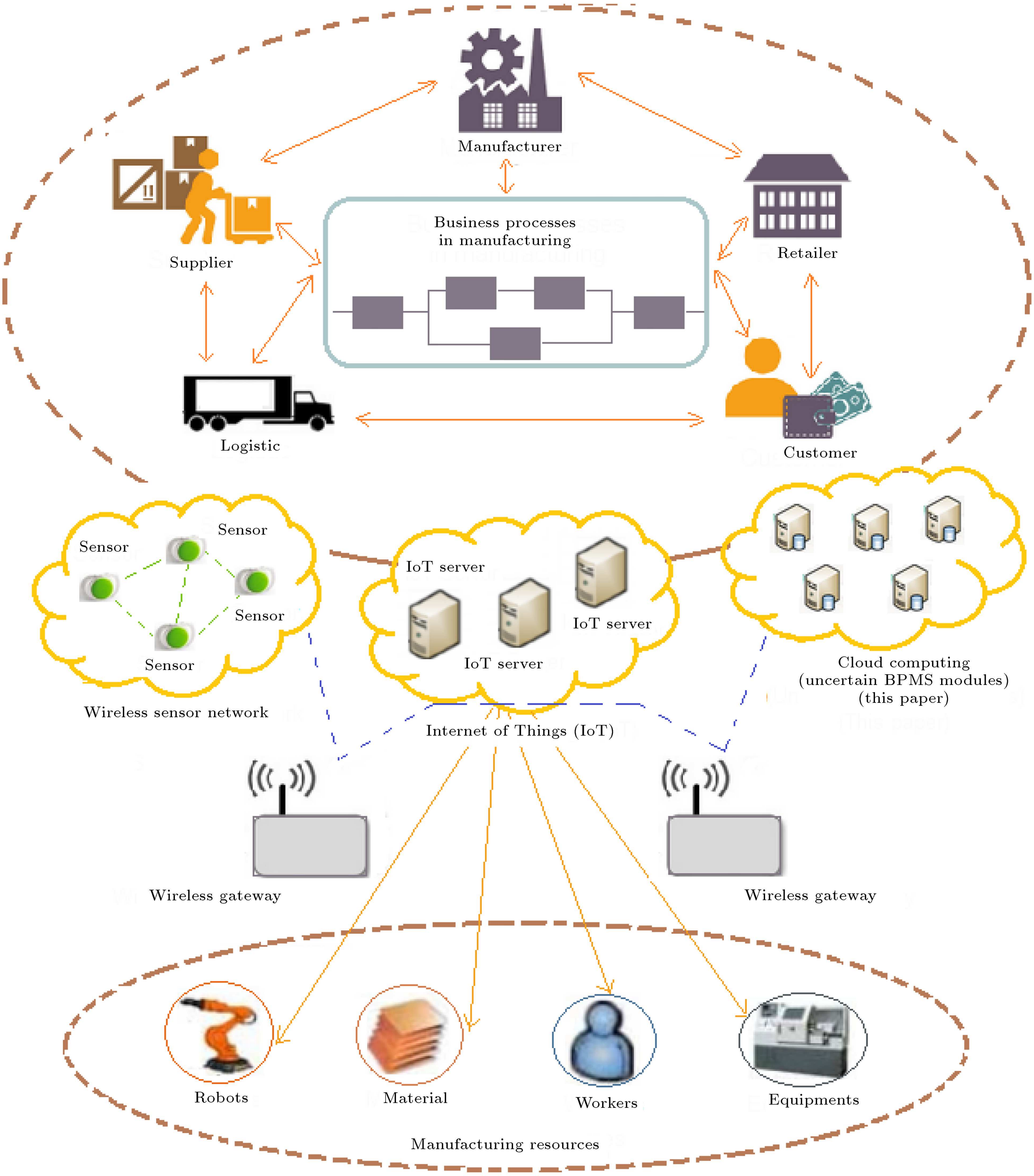


**Figure 1.** Smart manufacturing in the content of a smart factory.

factories. These processes are commonly supported by process-aware information systems such as Business Process Management System (BPMS).

BPMS, as an information system, can examine business processes to guarantee activities and tasks have profitable execution [7]. The current uncertain Business Process Management System (uBPMS) architecture [7,8] does not consider the business process orchestration. This paper aims to achieve an extension of uBPMS architecture [7,8] with a business process orchestration feature to make a response in real time and satisfy the demands, uncertainty conditions, and so on in a smart factory. The proposed extension can perform autonomous orchestration of business processes inward traditional uBPMS architecture based on desired values of objectives optimization.

Generally, the goal is to integrate the physical elements of a smart factory with the uBPMS to manage

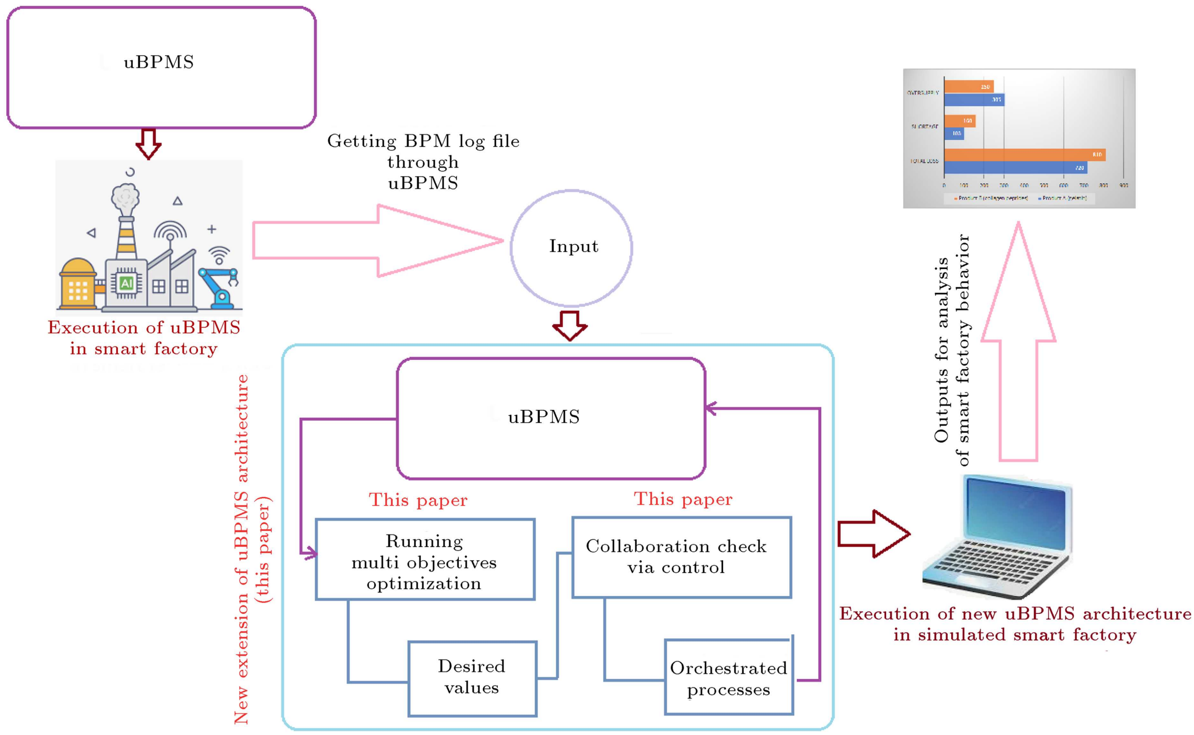


**Figure 2.** New extension of uBPMS architecture execution schematic.

uncertain processes. The notable mark is that process orchestration is an important optimization problem in the smart manufacturing industry [9]. Therefore, to provide orchestration, the current uBPMS architecture has to be extended using a bi-level optimization approach in which each level acts using different algorithms (Figure 2).

As Figure 2 presents, the main contributions of this study are as follows:

- ✓ Developing a new extension of uBPMS for acting in smart factories based on fuzzy-Markovian uncertainty sources,
- ✓ Defining new centralized control behavior based on orchestration concept to guarantee correct collaboration between processes inward uBPMS (This new control behavior acts through graph traversal (Algorithm 2)),
- ✓ Defining new optimization behavior based on desired value calculations of different objective functions for assurance of process orchestration (This new optimization behavior applies using the Pareto optimality method (Algorithm 1)).

This paper has been organized as follows: Section 2 illustrates the literature review. Section 3 defines a new extension of traditional uBPMS architecture. Section 4 evaluates the proposed architecture performance in the simulated model of the Rousselot smart factory in Belgium. Section 5 presents a discussion and managerial implications. Finally, Section 6 illustrates conclusions.

```
Input: GP, GC, Parameters
Output: value
Foreach gc_i in GC_i do
    value = execute (multi-objective function)
execute (Algorithm 2)
Update (value)
```

**Algorithm 1.** The upper-level algorithm of the proposed optimization approach.

## 2. Literature review

With the widespread business digitalization, there is an increasing need for smart manufacturing process management [5]. Different researches were made in different scopes based on the new architecture of an uBPMS development for control and optimization of uncertain business processes in smart manufacturing through orchestration's feature as follows.

### *2.1. Integrating process management in smart factory*

In [10], authors believe new system architectures for a smart production system are necessary. This architec-

*Input:* $A, R_{ch}, R_o, C_{ch}, C_o$

*Output:* $UBP: \begin{cases} A \rightarrow (R_{ch}, C_{ch}) \\ (A, E) \rightarrow (R_o, C_o) \end{cases}$

*for each* $a_i$ *in A do*

  $\Psi((R_{ch} \cup R_o) \rightarrow a_i)$

  $Z((C_{ch} \cup C_o) \rightarrow a_i)$

  $T(a_i \rightarrow T_s)$

*activity: new list[]* ^*used to create a list of BP activities with fuzzy-Markovian uncertainty*

*for each* $t_{s_i}$ *in* $T_s$ *do*

  *activity* $\leftarrow a_i$

*generate(graph)*

*select(traversal(graph))*

*update(graph)* ^*used to select the best collaboration through graph traversal*

**Algorithm 2.** The lower-level algorithm of the bi-level optimization approach.

ture is based on BPM and IIoT integration. In this case, they proposed an architecture that shows how to integrate the Cyber-Physical System (CPS) of a smart factory with BPM. This method is illustrated using multi-use cases, and the results show the benefits of BPM technology usage on the flexibility and adaptivity of production processes in IIoT. In [6], the authors show the need for industries to service orchestration, which is provided through business processes. They propose an architecture for managing Asset Administration Shell (AAS). The motivation of this paper is the orchestration of device services through business processes and AAS. The results show better interoperability between manufacturing systems. In [9], the logic of autonomous manufacturing is simplified. Also, the Autonomous Manufacturing Task Orchestration (AMTO) problem is presented. In order to formulate the problem and produce an optimal solution, an improved Hidden Markov Model (HMM) is applied. The results show the solution can tune the situation based on the real-time manufacturing data. In [11], the integration between the BPM life cycle and the data-based process is shown in industry 4.0. The authors present the application of machine learning and BPM standards in order to assist the development of industry 4.0 related phases. The results define more efficient smart growth and customer-oriented operations. In [12], the authors present a new method for increasing the manufacturing operations flexibility. This approach acts based on the different modules to orchestrate production plans immediately. The results show new functionalities in machine operations and adjustments between humans and manufacturers. In [13], authors believe in achieving processes monitor and control uninterrupted. They proposed object detection for smart factory process control through machine learning, which can provide flexible processes, decrease downtimes, and optimize supply chains in smart factories. In [14], integrating digital twin and big data techniques is applied to provide sustainable product management. The results show higher productivity, lower cost, and better quality in smart manufacturing industries. Table 1 also presents an overview of integrating process management in smart factory research.

### *2.2. BPMS architecture design methods*

There are different BPMSs in the business world with an emphasis on certain features [15]. Numerous studies have been focused on the design of BP. In [16], the authors proposed solutions for the modeling and simulating Dynamic BPs (DBPs). Their proposed architecture consists of several new components which are not domain-specific. Also, "components and their relationship are organized in such a way as to support rule and context-based DBP modeling and simulation". In [17], the solutions to modeling ubiquitous BP (uBP) are provided, too. The main aim of this modeling language is to provide an extension of Business Process Modeling Notation (BPMN) that allows the creation of end-to-end uBP. In addition, in [18], an academic generic BPMS user portal definition for the execution of BP in a certain mode is presented. In this user portal definition, some current BPMS engines exist that act with each other through the dynamic user portal. The main purpose of this research is to create a new BPMS architecture that can integrate with every process engine, such as Activiti, Bonita, and so on. Regarding the development of new uBPMS architecture, authors proposed [7,8] two uBPMS architectures. One has different engines that can manage uncertain BP through different uncertainty solution methods such as stochastic, fuzzy, and fuzzy-Markovian. The results indicated that the proposed architecture supports most uncertainty and improves BP performance. In [8], the authors presented an autonomous architecture of uBPMS, which can manage uncertain processes but cannot provide autonomous management in smart factories with smart manufacturing features.

In [19], authors believed that advances in smart sensors and digital process controls could radically

**Table 1.** An overview of integrating process management in a smart factory.

| Reference | Problem | Solution |
|---|---|---|
| [10] | Management of smart production system | New architecture based on BPM and IIoT integration |
| [6] | Services orchestration through business processes | New architecture based on business processes and asset administration shell |
| [9] | Autonomous manufacturing task orchestration | Improved version of hidden Markov model |
| [11] | Development of Industry 4.0 related phases | Machine learning and BPM standards |
| [12] | The flexibility of manufacturing operations | New approach based on the different modules to orchestrate production plans |
| [13] | Reading uninterrupted processes monitor and control | New object detection based on machine learning |
| [14] | Sustainable product management | New method based on digital twin and big data techniques |

enhance the efficiency of advanced manufacturing. Therefore, with new technologies (e.g., IIoT) embraced in smart manufacturing to address the management challenge, the future development trends of BPMS for uncertain smart manufacturing are necessary. The remarkable point is that a large number of processes in the smart factory are already fully or partially automated or are in the process of becoming automated. Also, they must know what to do and also what other processes to call after processes end their activities [20]. Since defined solutions in smart factory management will solve problems arising in a production facility with dynamic and rapidly changing boundary conditions [10], process management is an important topic in the IIoT environments. At the current stage of the state-of-the-art, no method applies the orchestration and optimization through smart factory-related business processes. Therefore, this study aims to present an integration of IIoT environments with BPM lifecycle using a new extension definition of uBPMS architecture.

## 3. Bi-level uncertain architecture of BPMS

The aim of the proposed architecture is actively smart manufacturing management based on integrating BPM with smart manufacturing features. This architecture can control and optimize uncertain business processes through bi-level optimization components in a smart factory. The mentioned operations will be added to uBPMS [7,8] and operate based on orchestration, choreography, and optimization algorithms, which are illustrated more in this section.

### *3.1. Proposed architecture description and preliminaries*

Generally, the fuzzy-Markovian uncertainty is typically presented in most real scenarios. There are two main categories of luck: stochastic, which models the stochastic variability, and fuzziness, which relates to the undefined boundaries of model parameters [21]. In the proposed approach, we consider the fuzzy-Markovian as a source of uncertainty and a powerful model. In this type of uncertainty, the fuzzy activity in the process could interact with other activity through Markovian properties when there is no sufficient information. In this case, the system status cannot be calculated precisely. These properties should be applied to dynamic system descriptions. This paper introduces the fuzzy Markov chains to reduce computational complexity and facilitate decision-making [22]. Also, the problem in this study is considered through a multi-objective optimization problem, and the Pareto optimality solution is applied to solve it. Pareto optimality is an idea in the optimization field and one method to detect good solutions to multi-objective problems. In this case, single objectives can be optimized simultaneously. Therefore, Pareto optimality is displayed as a set of non-inferior solutions in the objective space. It can define a boundary domain in which each objective can be optimized without abandoning other objectives [23]. The Pareto-optimality solution has been used in the proposed architecture, too.

The formulation of this architecture is as follows [7]:

- $I = \{i_z | z = 1, 2, 3, ..., n_z\}$ involves a set of the input, where $i_z$ represents $z$th input while $n_z$ represents the total number of inputs;
- $O = \{o_j | j = 1, 2, 3, ..., n_o\}$ involves a set of certain outputs, where $o_j$ represents $j$th output while $n_o$ represents the total number of outputs;
- $O' = \{o'_f | f = 1, 2, 3, ..., n_{o'}\}$ involves uncertain outputs set, where $o'_f$ represents the $k$th output while $n_{o'}$ represents the total number of uncertain outputs;
- $U = \{u_h | h = 1, 2, 3, ..., n_u\}$ involves uncertain factors set, where $u_h$ represents the $y$th uncertain factor while $n_u$ represents the total number of uncertain factors;
- $g(I, U) = O'_{IU}; (I_i, U_h) \rightarrow O'_{IU}$ involves the uncertain triple transform. $O'_{IU} = ((p'_n, \mu_{p'_n}), P_{(p'_n, \mu_{p'_n})})$ involves the uncertain result;
- $(r, m)$ defines the required resources and mechanisms.

According to the above notations, a business process with a fuzzy-Markovian uncertainty state is presented in Figure 3.

Also, the formal representations of the method are as follows [24]:

- $\tilde{S} = \left\{\tilde{s}_a | a = 1, 2, ..., n_a\right\}$ involves a set of fuzzy states (e.g., $\tilde{s}_a$ with $\mu_{\tilde{s}_a}$ membership function).
- $P_{aa+1}(k)$: The system probability from state $\tilde{s}_a$ to state $\tilde{s}_{a+1}$ when the decision $k$ is occurred ($P(\tilde{s}_a) = E(\mu_{\tilde{s}_a})$).
- The conditional probability of the system moving from fuzzy state $\tilde{s}_a$ to fuzzy state $\tilde{s}_{a+1}$ when the decision $k$ is taken (Eq. (1)): ($N$ is the number of states, and $X$ is the state of the process.)

$$P_k(\tilde{s}_a | \tilde{s}_{a+1}) = \frac{P(\tilde{s}_a, \tilde{s}_{a+1})}{P(\tilde{s}_{a+1})},$$

$$P(\tilde{s}_{a+1} > 0) = P\{\tilde{X}_1 = \tilde{s}_{a+1} | \tilde{X}_0 = \tilde{s}_a\}$$

$$= \sum_{k=0}^{N} P(\tilde{s}_{a+1} | k) \frac{P_K \mu_{\tilde{s}_a}(k)}{P(\tilde{s}_a)}. \tag{1}$$

- The states of the fuzzy Markov chain in a business process are defined using the transition probability matrix, which can give the fuzzy initial transition probability of the state $\tilde{s}_a$ to fuzzy state $\tilde{s}_{a+1}, (a \in \{1, 2, ..., n_a\})$ (Eq. (2)):

$$\begin{bmatrix} P(\tilde{s}_1|\tilde{s}_1) & .. & .. & .. & P(\tilde{s}_{n_a}|\tilde{s}_1) \\ .. & .. & .. & .. & .. \\ .. & .. & .. & .. & .. \\ .. & .. & .. & .. & .. \\ P(\tilde{s}_1|\tilde{s}_{n_a}) & .. & .. & .. & P(\tilde{s}_{n_a}|\tilde{s}_{n_a}) \end{bmatrix}. \tag{2}$$

- The matrix $\tilde{P}$ is stochastic and calculates through $\tilde{P} = A * B * P$ (Eq. (3)). The matrix $A$ is a matrix of the membership function values. Thus:

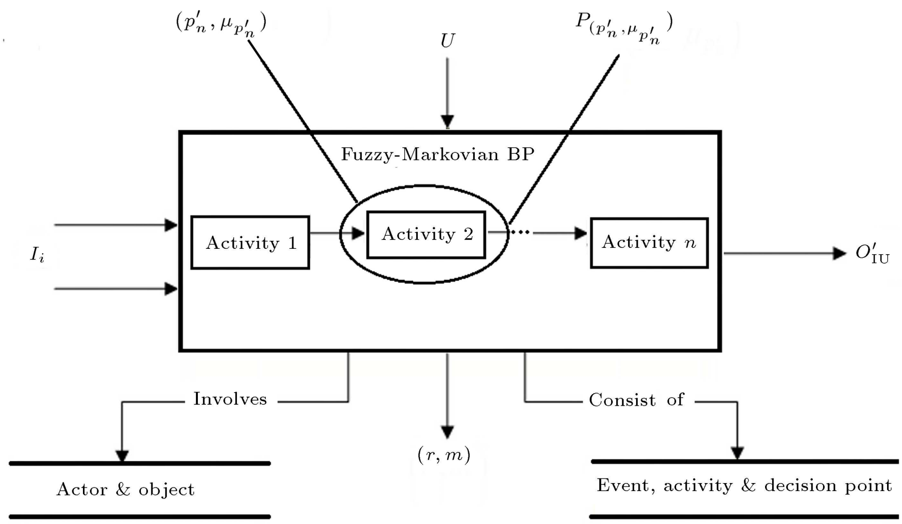


**Figure 3.** A schematic of a business process with fuzzy-Markovian uncertainty.

$$A = \begin{bmatrix} \mu_{\tilde{s}_1}(0) & \cdots & \cdots & \cdots & \mu_{\tilde{s}_{n_a}}(0) \\ \mu_{\tilde{s}_1}(1) & \cdots & \cdots & \cdots & \mu_{\tilde{s}_{n_a}}(1) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \mu_{\tilde{s}_1}(N) & \cdots & \cdots & \cdots & \mu_{\tilde{s}_{n_a}}(N) \end{bmatrix}$$

$$B = \begin{bmatrix} \frac{P_0 \mu_{\tilde{s}_1}(0)}{P(\tilde{s}_1)} & \cdots & \cdots & \cdots & \frac{P_N \mu_{\tilde{s}_1}(N)}{P(\tilde{s}_1)} \\ \frac{P_0 \mu_{\tilde{s}_2}(0)}{P(\tilde{s}_2)} & \cdots & \cdots & \cdots & \frac{P_N \mu_{\tilde{s}_2}(N)}{P(\tilde{s}_2)} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \frac{P_0 \mu_{\tilde{s}_{n_a}}(0)}{P(\tilde{s}_{n_a})} & \cdots & \cdots & \cdots & \frac{P_N \mu_{\tilde{s}_{n_a}}(N)}{P(\tilde{s}_{n_a})} \end{bmatrix}$$

$$P = \begin{bmatrix} P_{00} & \cdots & \cdots & \cdots & P_{0N} \\ P_{10} & \cdots & \cdots & \cdots & P_{1N} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ P_{N0} & \cdots & \cdots & \cdots & P_{NN} \end{bmatrix} = [P_{ij}],$$

$$\tilde{p} = \begin{bmatrix} \frac{P_0 \mu_{\tilde{s}_1}(0)}{P(\tilde{s}_1)} & \cdots & \cdots & \cdots & \frac{P_N \mu_{\tilde{s}_1}(N)}{P(\tilde{s}_1)} \\ p(\tilde{S}_1|\tilde{S}_1) & \cdots & \cdots & \cdots & p(\tilde{S}_{n_a}|\tilde{S}_1) \\ p(\tilde{S}_1|\tilde{S}_2) & \cdots & \cdots & \cdots & p(\tilde{S}_{n_a}|\tilde{S}_2) \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ p(\tilde{S}_1|\tilde{S}_{n_a}) & \cdots & \cdots & \cdots & p(\tilde{S}_{n_a}|\tilde{S}_{n_a}) \end{bmatrix}$$

$$= A * B * P. \tag{3}$$

An important operation in the proposed architecture is objective optimization. In the main, there are different objectives that could be identified in the business market. In this step, the Pareto-optimality solution has been used [25]. In the optimization component, the arithmetic representation is necessary.

As illustrated earlier, they can be described using fuzzy-Markovian definitions. These definitions try to find the acceptable values of all objective functions through a bi-level optimization approach on the upper level (Figure 4). The upper level of the proposed approach could be optimized for the fuzzy-Markovian uncertainty as following steps (Algorithm 1):

- Initialization of optimization and control engine objectives vectors (GP, GC) inside the definition of fuzzy-Markovian parameters;
- Presentation of feasible solutions through a Pareto-optimal method that satisfies defined objectives;
- Execution of Algorithm 2 (lower level algorithm) for selecting appropriate values (These values should guarantee business processes orchestration and choreography in smart manufacturing).

Also, the lower level is concerned with the decision-making to select process optimal behavior under fuzzy-Markovian uncertainty through the control engine.

The remarkable point is that each business process has orchestration and choreography engines to describe its features (such as priority and conditions). The operations of these engines have been presented using the collaboration concept as activities set ($A$)

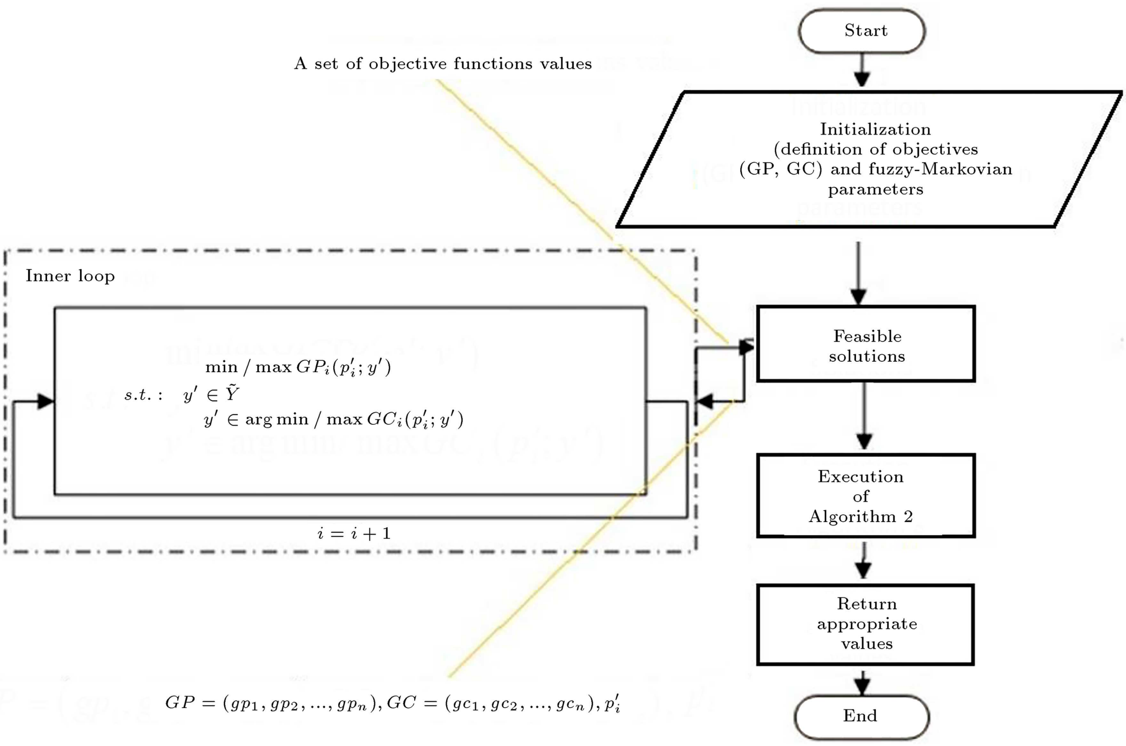


**Figure 4.** The upper-level algorithm of the proposed optimization approach.

that are made by business process inward smart manufacturing. The collaboration concept is based on the activity sequence and relationships between them. In general, there are two ways to construct business processes: orchestration and choreography. The orchestration concept defines a single centralized executable BP as the orchestrator that can synchronize the interaction between different services. Also, service choreography is an interaction description of the involved services based on exchanged messages and their rules, as shown in Algorithm 2. In general, the activity sets of multiple processes involved in collaboration are mutually disjoint (i.e., $a_i, a_j \in A \wedge i \neq j \Rightarrow a_i \cap a_j = \phi$) [26].

In this case, the proposed architecture presents the orchestration and choreography of the BP with fuzzy-Markovian uncertainty using a directed graph data structure as $G = (uBP, A, E, R_{ch}, R_o, C_{ch}, C_o, a_o)$ where:

- uBP represents the business process in fuzzy-Markovian uncertainty;
- $A$, as graph nodes set that shows process activities;
- $E$, as graph edges set that shows activities collaboration;
- $R_{\mathrm{ch}}$, as choreography rules set;
- $R_{\mathrm{o}}$ as orchestration rules set;
- $C_{\mathrm{ch}}$, as choreography constraints set;
- $C_{\mathrm{o}}$, as orchestration constraints set;
- $a_o$ is the initial activity, as discussed earlier.

■ $\Psi : (R_{ch} \cup R_o) \rightarrow A$ is a function for assigning appropriate rules to activities or set of activities;

■ $Z : (C_{ch} \cup C_o) \rightarrow A$ is a function for assigning appropriate constraints to activities or set of activities;

■ $T : A \rightarrow T_s$ is a function assigning each activity to a time stamp. Thus, the activities are ordered by their time stamp.

The orchestration engine has a model for identifying which collaborations are currently enabled. Thus, the proposed data structure can orchestrate the process by adding and removing graph nodes based on process features and initial activities. These initial activities are created through a choreography engine (Figure 5). As illustrated in Figure 5, the algorithm is intended to orchestrate the interaction between BP activities with fuzzy-Markovian uncertainty. In this case, the control engine as a central controller can manage activities and all the interactions between them through orchestration and choreography approach for each uncertain process based on upper-level results.

Figure 5 shows that the orchestration of the business process in fuzzy-Markovian uncertainty can be made by adding or removing nodes. In this case, the collaboration of new states of uncertain process has been updated. The remarkable point of the proposed uBPMS is reaching scalability, agility, and flexibility features of uncertain process inside the choreography approach through the following algorithm steps (Algorithm 2):

■ Create a list of business process activities with fuzzy-Markovian uncertainty;

■ Create a graph;

■ Generate different interactions between processes activities through graph traversal;

■ Select the best collaboration based on rules and constraints;

■ Update graph.

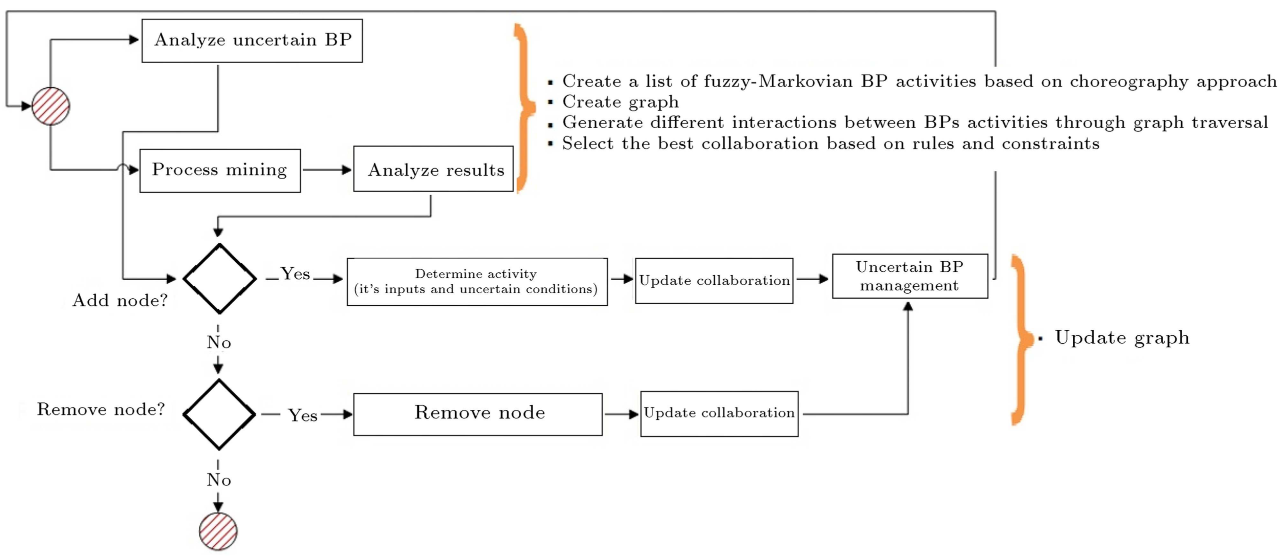


**Figure 5.** The lower-level approach of the proposed optimization approach.

The other required definition in the bi-level uncertain architecture approach is formulated through the following definitions (Eq. (4)) [22,23,25]:

$$\begin{aligned}&\min/\max GP_i\left(p'_i;y'\right), \forall i=1,2,...,n\\&\text{s.t.:} y'\in\tilde{Y}, y'\in\arg\min/\max GC_i\left(p'_i;y'\right),\\&\forall i=1,2,...,n,\end{aligned} \tag{4}$$

where:

$$GP=\left(gp_1,gp_2,...,gp_n\right), GC=\left(gc_1,gc_2,...,gc_n\right),$$

$$\forall n\geq 2.$$

In this case, GP is upper-level objectives, and GC is lower-level objectives. Also, $p'$ defines the fuzzy-Markovian parameters for each objective function and $\tilde{Y}$ is a probable solution set (Eq. (5)):

$$GP_i\left(p'_i;y'\right)=GC_i\left(p'_i;y'\right)$$

$$=\sum_{i=1}^{N}\sum_{j=1}^{\min(n,l)}\left(a_{ij}\right)\left(\frac{P_i\mu_{\tilde{s}_i}(i)}{P\left(\tilde{s}_i\right)}\right), \tag{5}$$

where:

$C=C_{ch}\cup C_o=\left(c_1,c_2,...,c_l\right)$ is a vector of constraints,

$D=\left\{c_1(d),c_2(d),...,c_l(d)\right\}$ represents the decision type based on selected constraints,

$d=C\cap(GP\cup GC)$ and $a_{id}$ is the value of state $i$ and decision $d$,

$y'^{\circ}\in\tilde{Y}$ is a solution if there exists no other $y'\in\tilde{Y}$ which $gp_i\left(p'_i;y'\right)\leq gp_i\left(p'_i;y'^{\circ}\right)$,

$P'$ as an uncertain variable could be calculated from an uncertainty space $(\alpha,\beta,\delta)\rightarrow\mathbb{R}$ Chien et al. [23].

The uncertainty distribution $\phi$ of $P'$ and expected value of it have been defined by [23] (Eq. (6)):

$$\phi(x)=g\left\{P'\leq x\right\},$$

$$E\left[P'\right]=\int_0^{+\infty}g\{P'\geq x\}dx-\int_{-\infty}^{0}g\{P'\leq x\}dx. \tag{6}$$

The model of programming based on uncertain objective function $GP_i\left(p'_i;y'\right), \forall i=1,2,...,n$, uncertain constraints $GC_i\left(p'_i;y'\right), \forall i=1,2,...,n$, and confidence level $Z_j$ is (Eq. (7)):

$$\min/\max\Big(E\left[GP_1\left(p'_1;y'\right)\right], E\left[GP_2\left(p'_2;y'\right)\right],...,$$

$$E\left[GP_n\left(p'_n;y'\right)\right]\Big), \tag{7}$$

subject to:

$$g\left\{GC_i\left(p'_i;y'\right)\right\}\geq Z_j, j=1,2,...,p.$$

$y'$ as a Pareto solution for programming model is (Eq. (8)):

$$E\left[GP_i\left(p'_i;y'\right)\right]\leq E\left[GP_i\left(p'_i;y'^{\circ}\right)\right]. \tag{8}$$

## 4. Case study: Simulated model of Rousselot smart factory

In this paper, the simulated model of the Rousselot company is used. Rousselot is a smart factory in Belgium that is a worldwide producer of gelatin and collagen peptides. These products are manufactured with high quality, safety standards, and sustainability. This factory is among the famous factory in Industry 4.0. Also, this factory makes up the ordering, production, settings, and energy consumption validation through digitalized steps. In this case, the simulated model describes processes from ordering and production processes to delivery processes based on Rousselot company characteristics.

### *4.1. Implementation details of proposed method execution*

The performance evaluation of the proposed method was assessed by applying Java Script/HTML/XML/Python languages programming, MATLAB 2013a, and Java SE11 on a computer with a corei7 processor, 12GB RAM, and 1TB hard disk space (300GB SSD).

The simulation contains different smart shop floors, and their characteristics are presented as follows:

✓ Shop1 contains 4 machines (machine1, machine2, machine3, machine4 with operation times Norm(3,0.1), Norm(3,0.1), Tria(2,3,5), and Tria(2,3,5) respectively),

✓ Shop2 contains 2 machines (machine5, machine6 with operation times Norm(4,0.1), and Tria(3,5,6) respectively),

✓ Shop3 contains 3 machines (machine7, machine8, machine9 with operation times Tria(2,4,5), Tria(2,3,5), and Norm(3,0.1) respectively),

Also, each shop is equipped with machine sensors (for enabling machines to monitor internal processes), remote monitoring (for monitoring materials), and predictive maintenance (for decreasing maintenance costs).

As illustrated earlier, four processes (ordering, production, delivery, settings, and energy consumption validation) in one scenario are considered in this simulation. The time between entering these processes is as follows:

✓ Ordering as proc1: $EXPO(\lambda)$,

✓ Production as proc2: $NORM(\mu,\sigma^2)$,

**Table 2.** Other settings details of proposed method implementation.

| Parameters | Values |
|---|---|
| Number of assigned resources to each shop | shop1(10), shop2(8), shop3(15) |
| Number of rules for each process | Proc1(15), proc2(10), proc3(20), proc4(10), proc5(9) |
| Number of uncertain constraints | 5(Inventory availability, order tracking, order placement, order routing, shipping and delivery) |
| Number of scenarios | 1 with four processes (ordering, production, delivery, settings, and energy consumption validation) |
| Number of orders | 500 |
| $\alpha, \beta, \gamma, \mu, \sigma^2, \lambda$ | Based on Ref. [26] formulations (which are described later in this section) |
| Simulation time | 10 hours |

✓ Delivery as proc3: $TRIA(\alpha, \beta, \gamma)$,

✓ Settings as proc4: $EXPO(\lambda)$,

✓ Energy consumption validation as proc5: $NORM$ $(\mu, \sigma^2)$.

The production sequences of gelatin(pro1) and collagen peptides(pro2) are (op3, op4) and (op1, op2, op4, op5) respectively too. In addition, the mentioned factory has five main objectives as following:

- Minimizing production resources;
- Minimizing energy consumption;
- Reduce downtime;
- Minimizing capacity shortage;
- Minimizing capacity oversupply.

Other settings of simulation parameters and programming environment are present in Table 2.

Orchestration of all processes involved in the mentioned factory can help upgrade the quality of products. The proposed bi-level approach can support these objectives and orchestrate all processes as presented later preciously. Figures 6 and 7 define a diagram of the proposed method application in the simulated model and the deployment platform of the executable version of the proposed approach in a smart factory. The performance evaluation of the proposed method for different objectives using the following scenario has been applied (Figure 8):

✓ Process a customer order inside different business rules,

✓ Evaluation of customer order, production configuration, and validation of energy consumption,

✓ Pick the requested items of customer order and pack them,

✓ Ship the order to the customer, control its quality, and delivery,

✓ Update the inventory capacity and order status.

In this case, as the XML code in Figure 9 shows, the interface has been created for each uncertain process and its involved activities in the considered scenario inward executable prototype of the proposed method.

Also, the orchestration and choreography concept of processes with different fuzzy-Markovian uncertainty variables (such as demand and capacity) through different service compositions have been shown in Figure 10. These orchestration and choreography description notations are based on the Web Services-Business Process Execution Language (WS-BPEL).

In this case, the lower level can define the orchestration function in uBPMS architecture. Thus, the aim is the correct, simple, and fast correlations between entire end-to-end processes and their resources in a centralized location. Also, this function provides powerful

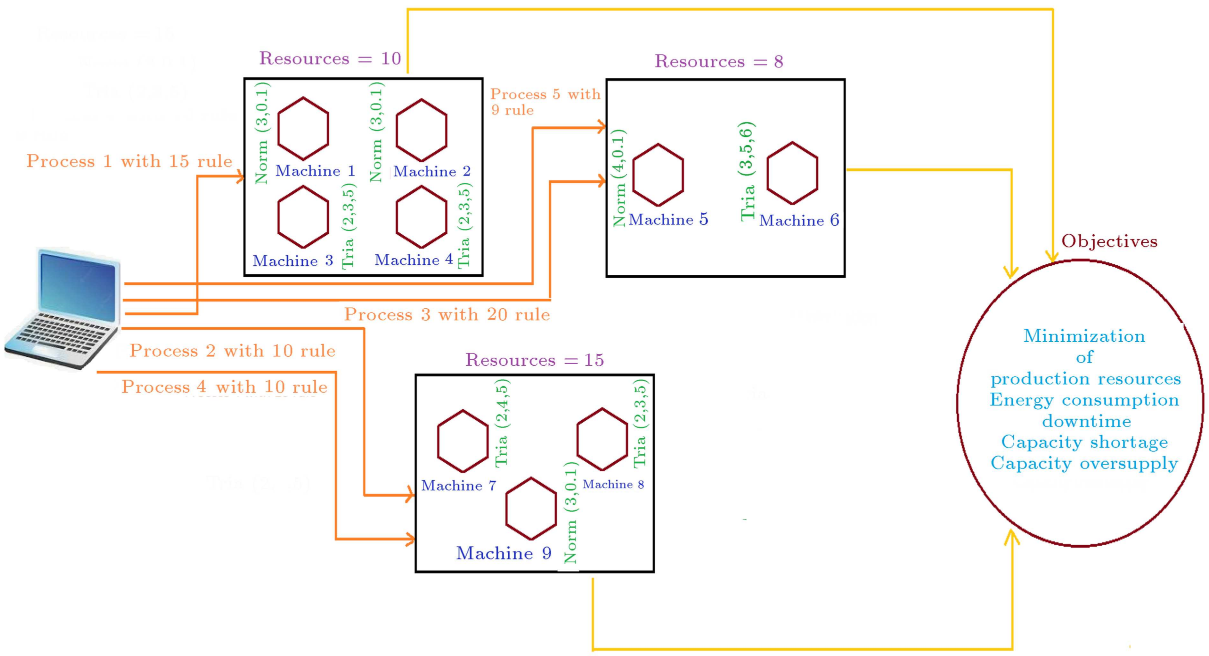


**Figure 6.** Schematic diagram of proposed method application in the simulated model.

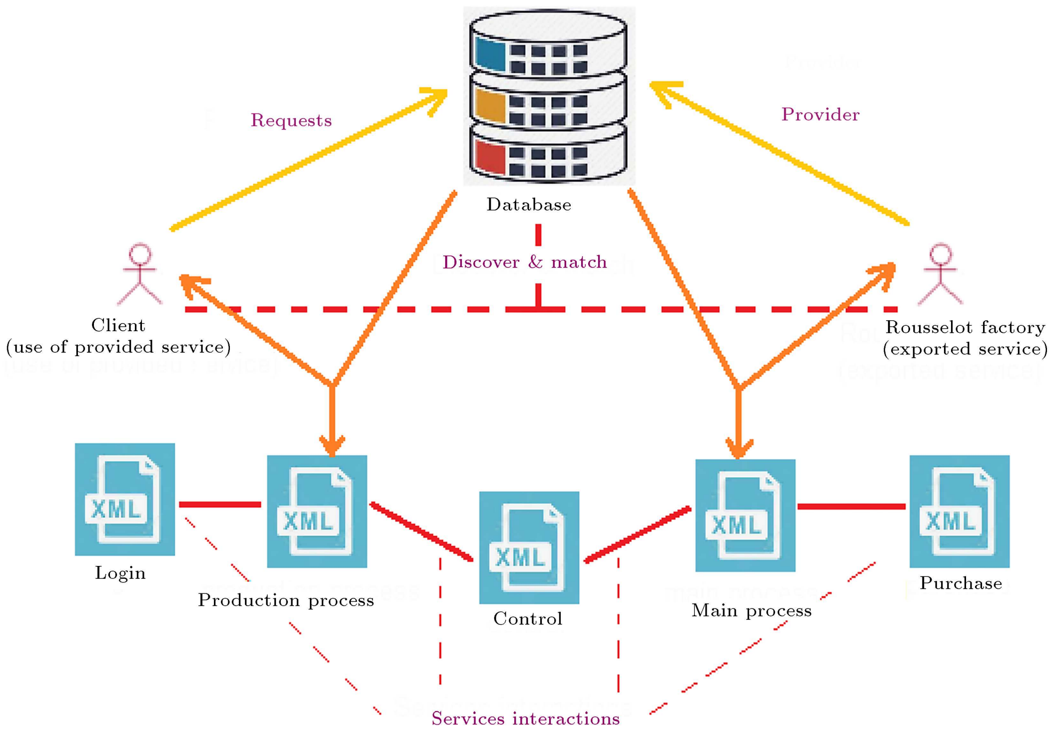


**Figure 7.** Deployment platform.

insights into operational gaps with increased complexity and resiliency for covering at the upper level.

The upper-level mathematical model of the proposed approach is as follows:

***Indexes, parameters, and decision variables:***

$t$ Index of time,

$k$ Index of supplier,

$j$ Index of factory,

$n$ Index of product,

$D_{nt}$ Predicted demand for product type $n$ at time $t$,

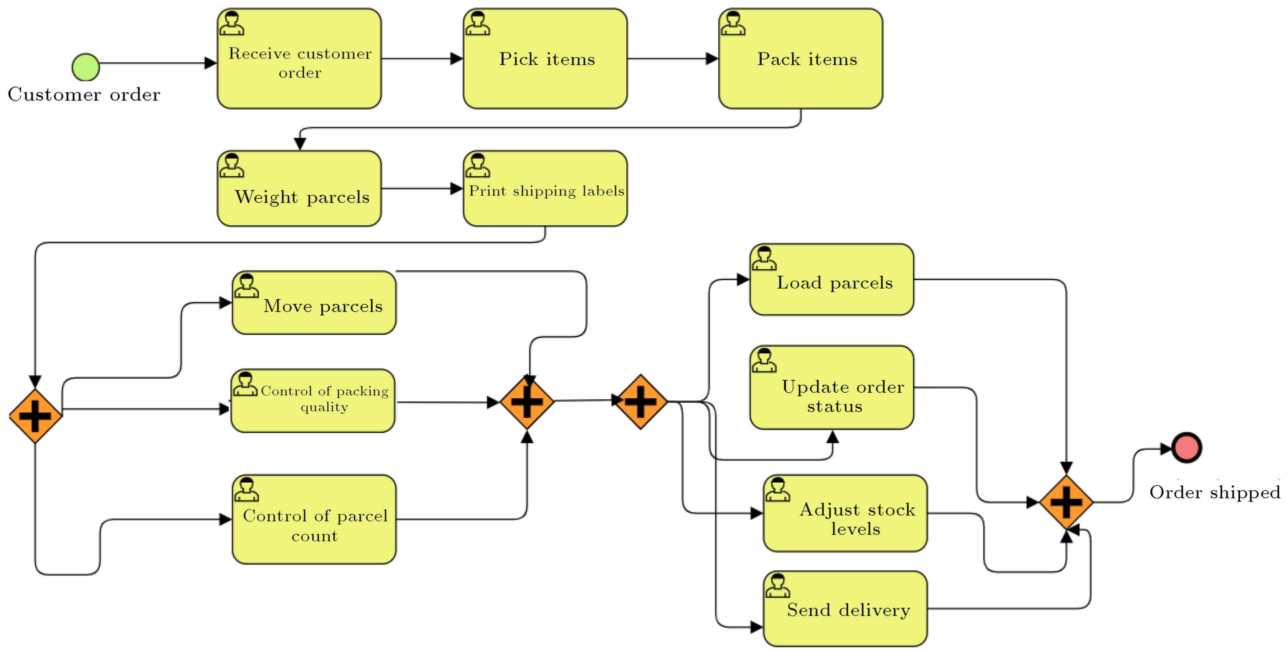


**Figure 8.** Diagram of the considered scenario for evaluating the proposed method performance.

```
<?xml version="1.0" encoding="UTF-8"?>
<process name="production" instantiation="message">
  <sequence>
     <action name="production resources" role="Agent"
operation="tns: consumption">
     <correlate correlation="tns: energy consumption" />
     <correlate correlation="tns: capacity shortage" />.
     <correlate correlation="tns: capacity oversupply" />
     \call process="tns: production" />
    </action>
    <action name="downtime" role="Agent" operation="tnc
:dcwn">
    </action>

  </sequence>
</process>
```

**Figure 9.** Interface code for creating production process as a sample.

$RD_{nt}$ Real demand for product type $n$ at time $t$,

$LT_{kj}$ Defined time for transporting raw material from source (supplier $k$) to destination (factory $j$),

$\mathrm{Cap}_j$ Storage capacity in factory $j$,

$P\max_{n}$ Maximum production capacity of product type $n$,

$\max_{t}$ Maximum total production capacity at time $t$,

$Pcast_{nj}$ Production cost of product type $n$ at factory $j$,

$icost_{nj}$ Cost of inventory holding for product type $n$ at factory $j$,

$w_{jt}$ Total number of workers at factory $j$ in a time $t$,

$aw_{jt}$ Total number of available workers at factor $j$ in a time $t$,

$v_{kjt}$ The total number of vehicles required for transporting between the source (supplier $k$) and destination (factory $j$) in $a$ time $t$,

$dp_j$ Daily production rate of existing machine $j$,

$Mcap$ Machine capacity,

$inven$ Amount of inventory,

$em$ Number of existing machines,

$\exp an_{nt}$ Expansion capacity of product $n$ at time $t$,

$cs_{ntj}$ Capacity shortage of product $n$ at factory $j$ at time $t$,

$co_{ntj}$ Capacity oversupply of product $n$ at factory $j$ at time $t$,

$up_{nt_1t_2}$ Upper bound of capacity for product $n$ during period $t_1$ to $t_2$,

$lp_{nt_1t_2}$ Lower bound of capacity for product $n$ during period $t_1$ to $t_2$,

$c_1$ Cost of capacity shortage,

$c_2$ Cost of capacity oversupply,

$ud_{nt}$ Distribution of predicted demand as an uncertain variable for product type $n$ at time $t$,

$ud_{nt}^{-1}$ Inverse distribution of $ud_{nt}$.

```
Service main process
Operation
 import login[int ID,string username]:bool
 import demand[int productnumber]:void
 import capacity[int value,int productnumber]:int
 import daily-production-rate[int productnumber,long int rate]:long int
 import logout [int ID]:void
Process
    login!;((demand!+capacity!)*;daily-production-rate!))*;logout!

Service login
Operation
 export login[int ID,string username]:bool
 export logout[int ID]:void
Process
    (login?+logout?)*

Service purchase
Operation
 export available [int ID]:bool
Process
    (available?)*

Service production process
Operation
 export capacity[int value,int productnumber]:int
 import available[int ID]:bool
 import quantity-vehicles[int num]:int
 import number-workers[int numm]:int
 import inventory-cost[long int cost]:long int
Process
    (available!;capacity?;((quantity-vehicles!+number-workers!)*;inventory-cost!))

Service control
Operation
 import production-resources[string resources]:string
 import energy-consumption[long int energy]:long int
 import down-time[Time time]:Time
 import capacity-shortage[int productnumber,int ID]:int
 import capacity-oversupply[int productnumber,int ID]:int
Process
    capacity-shortage!;(production-resources!+energy consumption!+down-time!)*;capacity-oversupply!
```

Orchestration:
Each process implements many services.

Choreography:
Each invocation results in an interaction-message exchange.

**Figure 10.** Orchestration and choreography concept of the simulated case through service composition as a sample.

***Objectives:***

$$Z_1 = \min\left(E\left[\left(\sum_j (em * mcap_j) * dp_j\right)\right]\right), \tag{9}$$

$$Z_2 = \min\left(E\left[\left(\sum_n\sum_t\sum_j\left((\text{pmax}_{nt} - D_{nt}) * p\cos t_{nj}\right) + \sum_n\sum_j (inven * i\cos t_{nj})\right)\right]\right), \tag{10}$$

$$Z_3 = \min\left(E\left[\left(\sum_k\sum_j\sum_t (v_{kjt} * LT_{kj}) + \sum_j\sum_t\left(\frac{aw_{jt}}{em}\right) + \sum_j\sum_n\left(\frac{cap_j}{inven}\right) * i\cos t_{nj}\mathbb{F}\right)\right]\right), \tag{11}$$

$$Z_4 = \min\left(E\left[\sum_n\sum_t\sum_j cs_{ntj}\right] = E\left[\sum_n\sum_{t_1}\sum_{t_2}(d_{nt} - up_{nt_1t_2})\right]\right), \tag{12}$$

$$Z_5 = \min\left(E\left[\sum_n\sum_t\sum_j co_{ntj}\right] = E\left[\sum_n\sum_{t_1}\sum_{t_2}(lp_{nt_1t_2} - d_{nt})\right]\right). \tag{13}$$

The $Z_1$ and $Z_2$ objectives (Eq. (9) and Eq. (10)) are the production resources and energy consumption minimization. The $Z_3$, $Z_4$, and $Z_5$ objectives (Eq. (11) to Eq. (13)) are to minimize the downtime, capacity oversupply and capacity shortage.

***Constraints:***

$$\underset{n}{p\max} \leq cap_j, \qquad \forall n, j, \tag{14}$$

$$Mcap \leq \underset{n}{p\max}, \qquad \forall n, \tag{15}$$

$$aw_{jt} + v_{kjt} \leq D_{nt}, \qquad \forall k, j, t, n, \tag{16}$$

$$\exp an_{nt} \leq \underset{n}{p\max}, \qquad \forall n, t, \tag{17}$$

$$\sum_n \exp an_{nt} \leq \max_t, \qquad \forall t. \tag{18}$$

These constraints (Eq. (14) to Eq. (18)) declare that:

- The quantity of maximum production capacity is lower than that of storage capacity;
- The quantity of machine capacity is lower than the maximum production capacity;
- The total quantity of available workers at the factory and vehicles needed for transportation is lower than the predicted demand;
- The quantity of expansion capacity is lower than the quantity of maximum production capacity;
- The total quantity of expansion capacity is lower than the quantity of maximum total production capacity.

As mentioned earlier, the uncertain program model is resolved as follows.

The uncertainty distribution $ud_{nt}$ can be formulated as (Eq. (19)):

$$ud_{nt}(P') = \frac{P' - P'_{n,t}}{P'_{n,t+1} - P'_{n,t}}, P'_{n,t} < P' < P'_{n,t+1}. \tag{19}$$

The normalized prediction error for product $n$ is [23] (Eq. (20)):

$$error_{nt} = \frac{RD_{nt} - P'_{n,t}}{P'_{n,t}}. \tag{20}$$

The $\varepsilon$ parameter value as a mean and $\sigma$ parameter value as a standard deviation of the normalized prediction error is [23] (Eq. (21)):

$$\varepsilon_n = \frac{\sum_n \sum_t error_{nt}}{t},$$

$$\sigma_n = \sqrt{\frac{\sum_n \sum_t (error_{nt} - \varepsilon_{nt})^2}{t-1}}. \tag{21}$$

Thus, the uncertain distribution of product $n$ at $t$ is (Eq. (22)):

$$ud_{nt}(P') = P'_{n,t} + P'_{n,t}(\varepsilon_n, \sigma_n). \tag{22}$$

Also, the constraints should be rewritten as Eq. (23) ($\alpha$, $\beta$, $\delta_1$, $\delta_2$, $\delta_3$ define degrees of parameters):

$$\begin{cases} g\{p\max_n \leq cap_j\} \geq \alpha_n \\ g\{Mcap \leq p\max_n\} \geq \beta_n \\ g\{aw_{jt} + v_{kjt} \leq D_{nt}\} \geq \delta_{1n} \\ g\{\exp an_{nt} \leq p\max_n\} \geq \delta_{2n} \\ g\left\{\sum_n \exp an_{nt} \leq \max_t\right\} \geq \delta_{3n} \end{cases} \tag{23}$$

### *4.2. Simulation results*

Based on the illustrated implementation steps, the results of the proposed method execution are presented in Tables 3–5.

As resulted in Table 3, the execution of the upper-level algorithm resulted in feasible solutions that are nearest the desired value. As a sample in Table 3, the desired value of production resources is 5.2 units, and the algorithm of the upper level in the proposed architecture provides 4.52 units; the resulting value is better than the previous result (8.1). After generating feasible solutions at the upper level, each solution was examined using a lower-level algorithm to guarantee the orchestration of processes. Each value for objectives should be tested based on processes. In this case, an okay reply from lower-level means that feasible solutions are appropriate, but a not-okay reply from lower-level means that collaboration between processes should be updated for assurance of

**Table 3.** Results of the upper level in proposed method execution.

| Objective | Feasible solutions (after-this paper) | Result (before) | Desired value |
|---|---|---|---|
| Production resources (unit) | 4.52 | 8.1 | 5.2 |
| Energy consumption (J) | 0.38 | 1.5 | 0.5 |
| Downtime (day) | 1 | 2 | 1.5 |
| Cost of capacity shortage ($) | 1900 | 4300 | 2500 |
| Cost of capacity oversupply ($) | 1400 | 3400 | 1600 |

**Table 4.** Results of lower level in proposed method execution.

| No. | Update phase | | Objectives name=value | Algorithm 1 analyze rate | Algorithm 1 analyze result |
|---|---|---|---|---|---|
| 1 | Ψ | Done | Production resources=4.52, | 100% | Ok |
| | Z | Done | Energy consumption=0.8, | | |
| | T | Done | Downtime=1.8, Capacity shortage=1900, | | |
| | | | Capacity oversupply=1400 | | |
| 2 | Ψ | Done | Production resources=4.52, | 50% | Not Ok collaboration should be updated |
| | Z | Done | Energy consumption=0.8, | | |
| | T | Done | Downtime=1.8, Capacity shortage=1900, | | |
| | | | Capacity oversupply=1400 | | |
| 3 | Ψ | Done | Production resources=4.52, | 20% | Not Ok |
| | Z | Done | Energy consumption=0.8, | | collaboration |
| | T | Done | Downtime=1.8, Capacity shortage=1900, | | should be updated |
| | | | Capacity oversupply=1400 | | |

**Table 5.** Results of process orchestration application in factories.

| No. | Process parameters | Resulted of orchestration |
|---|---|---|
| 1 | Number of done tasks | 25 |
| | Productivity | 76% |
| | Accelerate process | High |
| 2 | Number of done tasks | 22 |
| | Productivity | 52% |
| | Accelerate process | Moderate |
| 3 | Number of done tasks | 28 |
| | Productivity | 82% |
| | Accelerate process | high |

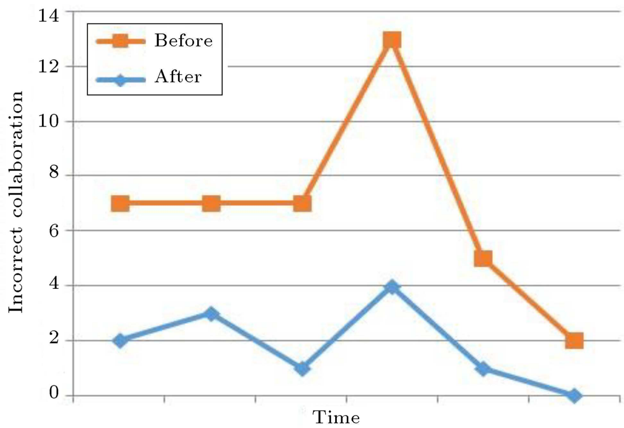


**Figure 11.** The robustness of the bi-level optimization approach.

process orchestration. As a sample in Table 4, row one shows the correct solutions, but other rows show the necessity of collaboration upgrade between processes. Thus, the proposed method can monitor the process behavior each time, which causes higher productivity and accelerates the process.

The results of the proposed method application could be affected by different parameters of the process, such as the number of tasks done, productivity, and accelerated process (Table 5). In general, the effect of orchestration assurance for higher task defines better productivity and accelerate the process.

To show the robustness of the proposed approach in smart manufacturing process management, two situations (process management based on the proposed method and process management without the proposed method) have been considered. Also, good performance means an incorrect collaboration presentation in this paper. Based on this definition, Figure 11 shows the robustness of bi-level optimization architecture to minimize incorrect collaboration presentation.

## 5. Discussion and managerial implications

In general, some supply chains, such as food supply chains in smart factories, involve new production processes. In this case, the definition of the new algorithms in order to control a product all over the supply chain [27] is necessary. Thus, this paper proposed a new architecture of uBPMS that can handle uncertain process management through bi-level optimization. This new method guarantees safe and high-quality products using process control and optimization.

In the new architecture, the process control is de-

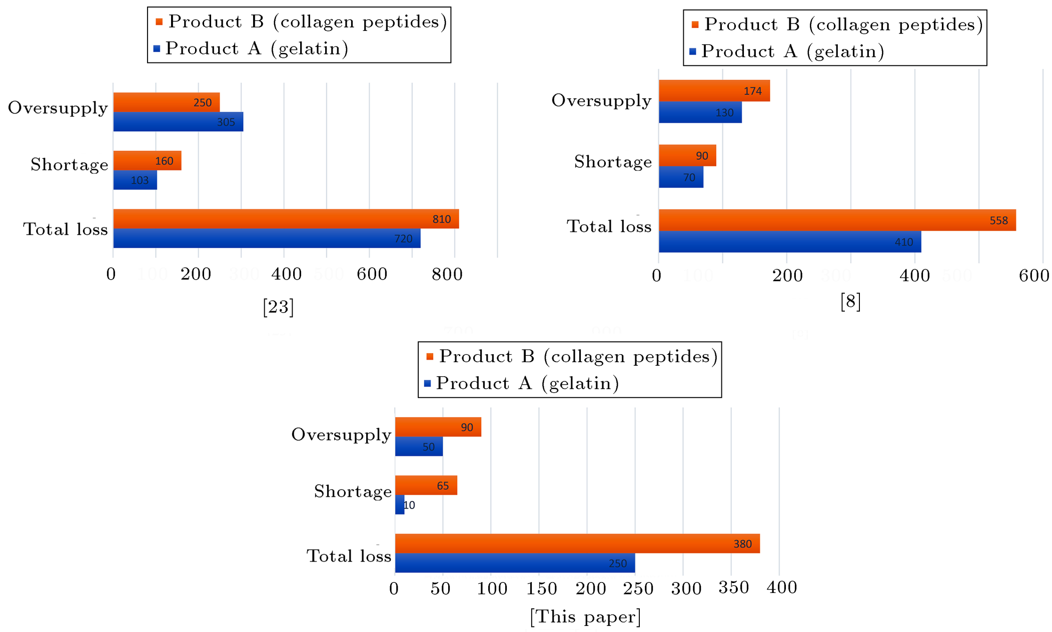


**Figure 12.** Proposed architecture comparison with [23], [8], and this paper.

fined based on orchestration. In this case, management of entire end-to-end uncertain processes is done based on the correct interactions between them. This type of control includes uBPMS and, based on the BPM cycle, provides different advantages as follows:

- ✓ Different events (such as errors) in processes are more detectable;
- ✓ Data in processes are more accessible. Thus, monitoring of processes with generated information can be upgraded;
- ✓ Transition between classic BPMS and uBPMS has a low risk;
- ✓ Resulted coordination is a necessary feature in businesses with hundreds of automations.

The new behavior of control operation in uBPMS causes automated activities with a center of attention on important process activities. Also, the required services and demands can decrease, providing lower costs, saving money, and higher productivity. The better traceability of uncertain processes is another advantage of process orchestration in centralized situations.

The results show that by playing an orchestration feature (for example, in the order process), factories have a good capacity to control their order processing. In this instance, a better position to avoid stockouts, reduction of delivery time, and minimization of shipping costs resulted in these factories. Furthermore, the orchestrated uncertain process is generated by considering the appropriate values of objective functions through control and optimization operations. These features are remarkable points for managing intelligent, uncertain marketing.

**Table 6.** Overhead result of proposed method in comparison with [23] and [8].

| References | Time complexity | Execution time |
|---|---|---|
| [23] | $O(n^2)$ | 48 minutes |
| [8] | $O(n)$ | 30 minutes |
| This paper | $O(log\ n)$ | 18 minutes |

As illustrated earlier, a simulated case has been evaluated based on received data from a Rousselot smart factory in Belgium to validate the proposed architecture. To compare the obtained results against other state-of-the-art approaches, the proposed method of this paper is compared with [23] and [8] (Figure 12). Also, the overhead of the proposed architecture in comparison with [23] and [8] has been calculated (Table 6).

In [23], authors reported that predicting demand and making capacity decisions for new advanced products is not easy. Also, in [8], the authors defined an autonomous architecture that can manage uncertain processes but cannot provide autonomous management in smart factories with smart manufacturing features. The investigation of their proposed architecture in smart manufacturing has also been defined as future work. As Figure 12 and Table 6 show, the proposed method is the most robust one compared to [23] and [8], with better time complexity and execution time because of lower operations inward new architecture. Also, the proposed architecture causes the cost minimization of capacity compared with [23] and [8]. For example, for product B (collagen peptides), this study on capacity shortage makes it save at least 300 units. Therefore, a 38% reduction in capacity loss will influence the product rate of the company.

In this case, customers have the ideal experience in their order process. As a result, the presented architecture could be applied as an extended version of the mentioned architecture in [23] and [8]. In addition, the number of operations is low in the bi-level architecture, and applied algorithms are simple, too. Therefore, the proposed method has a better overhead than [23] and [8] (Table 6).

## 6. Conclusions

Business Process Management System (BPMS) can be applied in different application scenarios, such as control and optimization of smart factories. Also, BPMS can make the end-to-end process visible, which can be useful for smart factories as distributed systems.

Smart manufacturing acts as an adaptive manufacturing system. In these systems, changes in production, supply chain, and customer requirements should be considered [27]. Thus, the uncertain version of BPMS should be applied to manage uncertainty.

This study proposed a new extension architecture of uncertain Business Process Management System (uBPMS) for acting in smart manufacturing management. This new extension is based on a bi-level architecture. A bi-level architecture applies to control and optimize smart manufacturing processes with fuzzy-Markovian uncertainty sources. This architecture progresses uBPMS by:

- ✓ Laying the groundwork for the bi-level optimization approach;
- ✓ Helping uncertain process control through orchestration and choreography approach in the lower level of the bi-level optimization approach.

Performance assessment of the proposed approach has been done using the simulated model of Rousselot company to identify the efficiency of the new architecture for minimizing the production resources, downtime, energy consumption, capacity oversupply, and capacity shortage over time. The new behavior of control operation in uBPMS causes automated activities with a center of attention on important process activities. Also, the required services and demands can decrease, providing lower costs, saving money, and higher productivity. The better traceability of uncertain processes is another advantage of process orchestration in centralized situations. The results show that by playing the orchestration feature, factories have a good capacity to control their order processing. In this instance, better positions to avoid stockouts, reduced delivery time, and minimized shipping costs resulted in these factories, too. Thus, this new method does not have significant capital risks of oversupply or shortage. Hence, this architecture can be presented as a smart decision-maker for managing smart manufacturing processes.

## References

1. Dumas, M., La Rosa, M., Mendling, J., et al. "Fundamentals of business process management", Springer (2013).
2. Waibel, M.W.W., Steenkamp, L.P., Moloko, N., et al. "Investigating the effects of smart production systems on sustainability elements", *Procedia Manufacturing*, **8**, pp. 731–737 (2017).
3. Wang, J., Ma, Y., Zhang, L., et al. "Deep learning for smart manufacturing: Methods and applications", *Journal of Manufacturing Systems*, **48**, pp. 144–156 (2018).
4. Bogle, I. "A perspective on smart process manufacturing research challenges for process systems engineers", *Engineering*, **3**, pp. 161–165 (2017).
5. Cheng, Y., Chen, K., Sun, H., et al. "Data and knowledge mining with big data towards smart production", *Journal of Industrial Information Integration*, **9**, pp. 1–13 (2017). DOI: 10.1016/j.jii.2017.08.001
6. Ochoa, W., Larrinaga, F., and Perez, A. "Architecture for managing AAS-based business processes", *Procedia Computer Science*, **217**, pp. 217–226 (2023).
7. Saraeian, Sh., Shirazi, B., and Motameni, H. "Towards an extended BPMS prototype: open challenges of BPM to flexible and robust orchestrate of uncertain processes", *Computer Standards and Interfaces*, **57**, pp. 1–19 (2018).
8. Saraeian, S., Shirazi, B., and Motameni, H. "Optimal autonomous architecture for uncertain processes management", *Information Sciences*, **51**, pp. 84–99 (2019).
9. Kai, D., Jingyunan, L., Felix, T.S.Ch., et al. "Hidden Markov model-based autonomous manufacturing task orchestration in smart shop floors", *Robotics and Computer-Integrated Manufacturing*, **61**, 101845 (2020).
10. Seiger, R., Malburg, L., Weber, B., et al. "Integration process management and event processing in smart factories: A systems architecture and use cases", *Journal of Manufacturing Systems*, **63**, pp. 575–592 (2022).
11. Czvetko, T., Kummer, A., Ruppert, T., et al. "Data-driven business process management-based development of Industry 4.0 solutions", **36**, pp. 117–132 (2022).
12. El Zant, C., Benfriha, Kh., Loubere, S., et al. "A design methodology for modular processes orchestration", *CIRP Journal of Manufacturing Science and Technology,* **35**, pp. 106–117 (2021).
13. Malburg, L., Rieder, M-P., Seiger, R., et al. "Object detection for smart factory processes by machine learning", *Procedia Computer Science,* **184**, pp. 581–588 (2021).

14. Ma, Sh., Ding, W., Liu, Y., et al. "Digital twin and big data-driven sustainable smart manufacturing based on information management systems for energy-intensive", *Applied Energy*, **326**, 119986 (2022).

15. Meidan, A., Garcia-Garcia, J.A., Escalona, M.J., et al. "A survey on business processes management suites", *Computer Standards & Interfaces Journal*, **51**, pp. 71–86 (2017). DOI: 10.1016/j.csi.2016.06.003

16. Vasilecas, O., Kalibatiene, D., and Lavbic, D. "Rule- and context-based dynamic business process modelling and simulation", *The Journal of Systems and Software*, **122**, pp. 1–15 (2016). DOI: 10.1016/j.jss.2016.08.048

17. Yousfi, A., Bauer, Ch., Saidi, R., et al. "uBPMN: A BPMN extension for modeling ubiquitous business processes", *Information and Software Technology*, **74**, pp. 55–68 (2016). DOI: 10.1016/j.infsof.2016.02.002

18. Delgado, A., Calegari, D., and Arrigoni, A. "Towards a generic BPMS user portal definition for the execution of business process", *Theoretical Computer Science*, **329**, pp. 39–59 (2016).

19. Yuan, Zh., Qin, W., and Zhao, J. "Smart manufacturing for the oil refining and petrochemical industry", *Engineering*, **3**, pp. 179–182 (2017).

20. Carstensen, J., Carstensen, T., Pabst, M., et al. "Condition Monitoring and cloud-based energy analysis for autonomous mobile manipulation: smart factory concept with LUHbots", **26**, pp. 560–569 (2016).

21. Clyde, M. and George, E.I. "Model uncertainty", *Institute of Mathematical Statistics*, **19**(1), pp. 81–94 (2004). DOI: 10.1214/088342304000000035

22. Pardo, M. and Fuente, D. "Fuzzy Markovian decision processes: application to queuing systems", *Computers and Mathematics with Applications*, **60**, pp. 2526–2535 (2010).

23. Chien, Ch-F., Dou, R., and Fu, W. "Strategic capacity planning for smart production: decision modeling under demand uncertainty", *Applied Soft Computing, Accepted-Manuscript*, **68**, pp. 900–909 (2017).

24. Hinkelmann, K., Gerber, A., Karagiannis, D., et al. "A new paradigm for the continuous alignment of business & IT: Combining enterprise architecture modelling & enterprise ontology", *Computers In Industry Journal*, **79**, pp. 77–86 (2015). DOI: 10.1016/j.compind.2015.07.009

25. Silva, R.C. and Yamakami, A. "The use of possibility theory in the definition of fuzzy Pareto-optimality", Springer Journal, **10**, pp. 11–30 (2011).

26. Zenkert, J., Weber, C., Dornhofer, M., et al. "Knowledge integration in smart factories", *Encyclopedia*, **1**(3), pp. 792–811 (2021).

27. Espinoza-Mejia, M., Saquicela, V., and Abril-Ulloa, V. "Ensuring Traceability and Orchestration in the food supply chain", *Artificial Intelligence, Computer and Software Engineering Advances*, **1**, pp. 135–149 (2021).

## Biographies

**Shideh Saraeian** is the Assistant Professor of Computer Engineering at Gorgan Branch, Islamic Azad University, Gorgan, Iran. She studied Computer Engineering at Najafabad Branch, Islamic Azad University, Esfahan, Iran, from 2001 to 2005. She graduated with an MSc in Information Technology Engineering from Yazd University, Yazd, Iran, from 2006 to 2008. She graduated with a PhD in Computer Engineering in Sari Branch, Islamic Azad University, Sari, Iran, from 2014 to 2017. Her research interests include uncertain Information Systems management, data/process mining, machine learning/intelligent algorithms, and network security/anomaly detection. She has published several research papers. Also, she authored two books in Persian.

**Babak Shirazi** is the Associate Professor of Industrial Engineering at Mazandaran University of Science and Technology, Babol, Iran. He studied Computer Engineering at Tehran University from 1996 to 2000. He graduated with an MSc and a PhD in industrial engineering from Mazandaran University of Science and Technology, Babol, Iran, in 2005 and 2010, respectively. His research interests include digital manufacturing, data science, system engineering, and Enterprise architecture. He has published over 80 research papers. He authored a book on enterprise resource planning in Persian.