HAL archives-ouvertes.fr

Similarity Evaluation Based on Intuitionistic Fuzzy Set for Service Cluster Selection as Cloud Service Candidate

Jorick Lartigau, Xiaofei Xu, Lanshun Nie, Dechen Zhan

▶ **To cite this version:**

Jorick Lartigau, Xiaofei Xu, Lanshun Nie, Dechen Zhan. Similarity Evaluation Based on Intuitionistic Fuzzy Set for Service Cluster Selection as Cloud Service Candidate. 5th International Working Conference on Enterprise Interoperability (IWEI), Mar 2013, Enschede, Netherlands. pp.36-49, 10.1007/978-3-642-36796-0_5 . hal-01474216

HAL Id: hal-01474216
https://hal.inria.fr/hal-01474216

Submitted on 22 Feb 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Distributed under a Creative Commons Attribution| 4.0 International License

Similarity evaluation based on intuitionistic fuzzy set for service cluster selection as cloud service candidate

Jorick Lartigau, Xiaofei Xu, Lanshun Nie, Dechen Zhan.

School of Computer Science and Technology, Harbin Institute of Technology
92 Xi Dazhi Str. Nangang District, Harbin 150001 China
Email: jorick@hotmail.fr, {xiaofei, nls, dechen}@hit.edu.cn

Abstract Cloud manufacturing (CMfg) provides new opportunities toward the servitization, and embeds a set of functional features to enhance the collaboration among various service providers and their resources. The main target is to compose dedicated manufacturing cloud, by encompassing a set of cloud services, to manufacture a requested service. CMfg is a recent concept, but already widely spread in the academic and industrial researches in China. The paper firstly focuses on the manufacturing environment background to understand its purpose. Thus as an introduction, the concept of CMfg is discussed. Finally, we present a method based on intuitionistic fuzzy set for the similarity evaluation between cloud services and service clusters. The objective is to match the best service cluster to provide composite resource services as cloud service candidates. Our method is ABC (Artificial Bee Colony) optimized, and its performance are discussed through experiments.

Keywords. Cloud manufacturing (CMfg), Service cluster, Cloud service, Intuitionistic fuzzy set (IFS), Artificial bee colony (ABC)

1 Background

Modern manufacturing industries are facing a major change in their organization, conducted by an unpredictable competition on a worldwide scale [1], the emergence of new information technologies, and cloud technology. Indeed, IoT (Internet of Things) / IoS (Internet of Services), Future Internet, Cloud computing and Virtualization techniques offers many new possibilities to remodel the manufacturing environment significantly. Meanwhile, during the past two decades, many advanced manufacturing models and technologies have been proposed in order to realize the aim of TQCSEFK (i.e. faster time-to-market, higher quality, lower cost, better service, better environment, greater flexibility, and higher knowledge) for manufacturing enterprises. Typical examples include computer integrated manufacturing (CIM), lean manufacturing (LM), digital manufacturing, agile manufacturing (AM), networked manufacturing (NM), virtual manufacturing (VM), application service provider (ASP), collaborative manufacturing network, industrial product-service system (IPS), manufacturing grid (MGrid), crowd sourcing and supply chain [2]. But the modern manufacturing faces new challenges, especially toward the survival of the SMEs (Small-Medium Enterprise).

1.1 The servitization and the needs of innovation

Servitization is a change process where manufacturing companies embrace service orientation and/or develop more and better services, with the aim to satisfy customer's needs, achieve competitive advantages and enhance firm performance [3]. The servitization tends to a high number of implied resource service providers collaborating and inter-connected for the value creation. According to [4], 58% of US manufacturers had servitized in 2007 and less than 20% of Chinese manufacturers had servitized in 2011. For actual major industries, servitization is a valuable source of expenditures. For instance, services represented the main part of IBM capital expenditures since 2010 [5].

The goal of servitization is to create a product-service shift. It implied a circular relationship where products create service opportunities. This relationship improves business opportunities and also has a fundamental impact on the product leading to transformation and innovation. It becomes a major change agent and driver of product innovation. In a world of competition and global market, the innovation is the constraint driver for expending its business, and maintaining its impact. The innovation is extended by the interplay of various service providers and demanders in a high collaborative level environment within many time zones, distances, or enterprise organizations. The need of intermediary core platform among the service providers, their related service centers and demander, to manage and orchestrate the operations is a key for innovation. In our days, from a business perspective, manufacturing companies sense that the core competitiveness of their product gravitates around all the service package offered as additional services (e.g. machinery maintenance, human resources training) (Fig. 1.).

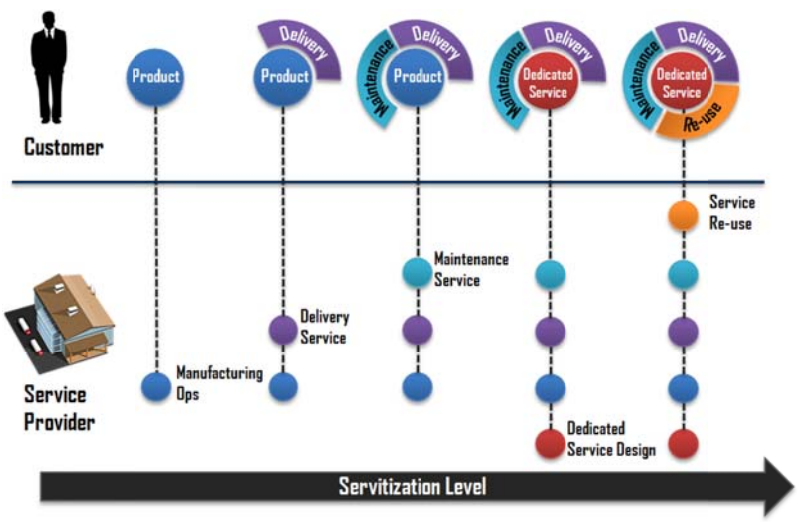

Fig. 1. Service provider / Customer relationship on a servitization scale

1.2 High-performance and precision equipment

In a world of competition with a wide range of offers, a service demander may give priority to the "*best of breed*". This phenomenon necessarily stimulates the enterprises to invest in better equipment and manufacturing resources, to enhance the service or product quality. Such investment can be unreachable for SMEs, especially with a large number of machineries and resources implied. As a result, collaboration between SMEs becomes an undeniable fact for their survival and expansion. On another hand, enterprises willing to invest might face under utilization of high-performance and precision equipment. From a business view, an interesting fact will be to offer the use of these equipments on demand, enabling a full sharing and open new business opportunities.

1.3 Enhance the QoS (Quality of Service), with interoperability, collaboration and standardization

As mentioned above, collaboration appears to be one of the main factors to succeed in modern manufacturing. Along the collaboration setups, come the interoperability and standardization challenges. The required manufacturing resource is transmitted among the enterprises, which employs different standardization strategies [6]. Considering a third-party platform, the core service has to insure the interoperability and standardization coverage of the shared resources among the service providers. The objective is to enhance the quality of the requested service, by evolving the best resources and equipments, while maximizing the collaboration between the service providers and their occupancy.

1.4 The emergence of Cloud Computing

Cloud computing is a concept consisting to dispatch computer programs to distant servers rather than local server or customer computer. The users are not anymore the host or the manager of the computer services, but can access from anywhere to online services without the need of managing the infrastructure model, often very complex. Cloud computing is changing the way industries and enterprises do their businesses in the meaning that dynamically scalable and virtualized resources are provided as a service over the Internet. Enterprises currently employ Cloud services in order to improve the scalability of their services and to deal with resource demands [7]. The concept of Cloud computing can be extended to the manufacturing field, providing on-demand service from remote resource service providers. Cloud computing mainly emerged to satisfy the needs of long time follow-up and service quality [8].

2 Cloud Manufacturing

Cloud Manufacturing (CMfg) service-oriented manufacturing model is been developed in order to satisfy the new paradigms and orientation of the manufacturing industries and market globalization. CMfg combines around a service-oriented architecture, new technologies and theory concepts [9]. CMfg realize the full sharing and circulation, high utilization, and on-demand use of various manufacturing resources and capabilities by providing safe and reliable, high quality, cheap and on-demand used manufacturing services for the whole lifecycle of manufacturing [2].

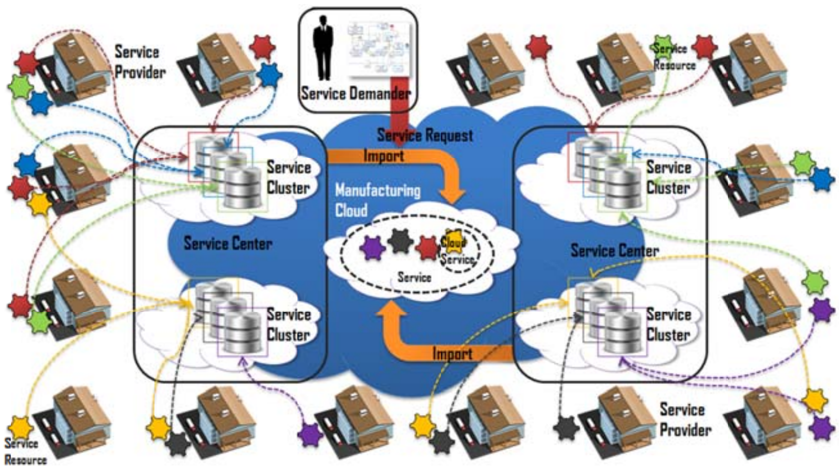

Fig. 2. CMfg concept model

The figure 2 presents the concept model of CMfg, based on resource service transaction. From the virtualization perspective, the physical service resources from various service providers are classified and characterized, to enable their encapsulation into service clusters. The service clusters are monitored and governed within one or several service centers. Thus, through a composition and evaluation process by an agent broker; here denoted as Cloud Platform, the system is able to build a set of cloud services to fulfill the functional and non-functional demander's service requirements. The four main parameters are defined as follow:

(a) *Resource Services* encompass a large set of resources, e.g. material resources, human resources, computational resources, equipment resources; which can interplay in collaboration to build existing and new services.

(b) *Service Clusters* gather the logical resource according to their functional parameters but also their non-functional QoS parameters, within preset ranges. Logical resources are the result of resource virtualization, transforming physical resources into logical. Service clusters embed a set of data associated to the functional definition of the resources and non-functional parameters denoted as QoSs, e.g. cost, reliability, and flexibility. QoS parameters are provided by the related service provider to the resource services, recorded and monitored from previous service operations.

(c) *Service centers* are responsible for the governance of the service clusters, and the agent services gravitating through the architecture layers to provide and share the needed information along the service processing.

(d) *Cloud services* are the representation on the cloud layer of the service tasks to perform. A service to provide is a chain of several Cloud services which can be the combination of several models (e.g. sequential, parallel, selective and cycle). A cloud service can be fulfilled by a single resource service within the same functional properties and non-functional QoSs minimum requirements.

3 Similarity evaluation between eligible service clusters and cloud services based on Intuitionistic Fuzzy Set (IFS)

3.1 Problem statement

One of the main features of CMfg is to enable the interplay of several composite resource services through a cloud service association, to model a new manufacturing service. Thus, the driven consideration is to set up an evaluation and composition strategy concerning the cloud service candidates. Cloud service candidates are fulfilled by composite resources encompassed in service clusters with the same functional characteristics and non-functional QoSs satisfying the service demander's minimum requirements. The composition process can be very exhaustive in term of computational time for cloud service chains involving a high number of resource services. Therefore, the problematic is to compose the best selection of resource services, without browsing all the possible solutions.

The advantage of a CMfg organization is to enable the resource virtualization and encapsulation into service clusters. Taking in account this specificity, an efficient approach is to pre-select eligible service clusters to provide the best resource services must be envisaged.

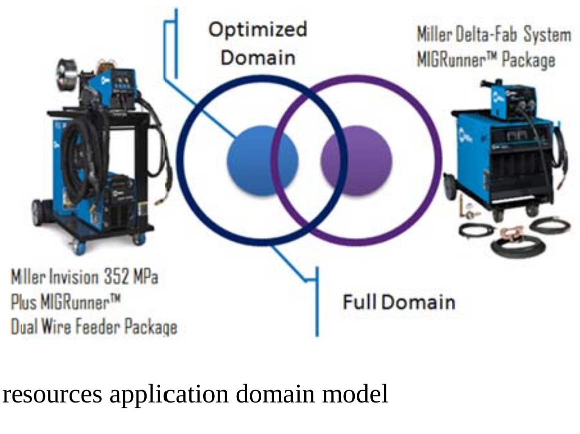

Fig. 3. Welding resources application domain model

To launch the composition and evaluation process, the CMfg system has to match the service clusters which can provide the right composite resource service. Along with the composition process, we have to keep in mind the diversity of Mfg resource services and their possible scalability toward several inputs and outputs, which might complicate the notion of candidate selection for a given cloud service. Indeed, a manufacturing machine is designed and optimized for a set of given inputs to generate a set of outputs, but can often manufacture services out of these bounds (Fig. 3.). However this process is not recommended since the machine is not optimized out of the bounds, and needs probably new calibrations and maintenance operations. Therefore, the membership definition of a manufacturing resource to a given set of inputs generating related outputs remains very fuzzy and cannot be expressed in full logical consideration.

In this paper, we propose a method based on IFS (intuitionistic fuzzy sets) to analyze the similarity degree between a requested set of cloud services and the available service clusters, within the same application domain (e.g. power supply, journal bearing) with optimized computational time, without going through all the possible solutions. The objective of our method is to select for each cloud service the best match among the service clusters. To illustrate the problematic, we model the whole composition process (Fig. 4.).

In the frame of manufacturing, the similarity evaluation between cloud services and service clusters enable to overview the input and outputs, to insure the correlation among them, for an optimal service clusters selection.

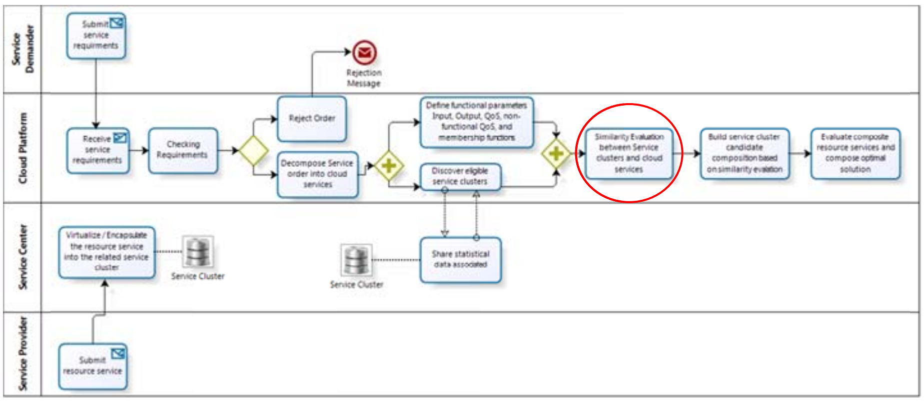

Fig. 4. The whole cloud service composition BPMN representation

The similarity evaluation can also be extended, for the consideration of interoperability (e.g. standards), and the non-functional QoSs requirements (e.g. reliability).

3.2 Introduction to IFS

Fuzzy set theory was introduced by L.A. Zadeh [10], by extension to set theories, where a given element is not anymore characterized by a binary (0 or 1) assessment to define its membership or not to a given set, but by a gradual assessment. The gradual assessment is the result of a membership function in the interval [0, 1] to characterize the strength of its membership to this set. The fuzzy set theory can be used in a wide range of domains in which information is incomplete or imprecise [11].

Let $X = \{x_1, x_2, \dots, x_i, \dots x_n\}$ be a fixed set of cardinality *n*. A fuzzy set *A* is expressed as:

$$A = \{\langle x, \mu_A(x)\rangle | x \in X\} \tag{1}$$

where $\mu_A: X \to [0,1]$ is the membership function of *A* and $\mu_A(x) \in [0,1]$ is the membership of $x \in X$ in *A*.

The notion of Intuitionistic Fuzzy Set (IFS) was introduced as generalization of the notion of fuzzy set [12].

Let $X = \{x_1, x_2, \dots, x_i, \dots x_n\}$ be a fixed set of cardinality *n*. An IFS *A* is expressed as:

$$A = \{\langle x, \mu_A(x), \upsilon_A(x)\rangle | x \in X\} \tag{2}$$

where respectively $\mu_A: X \to [0,1]$ and $\upsilon_A: X \to [0,1]$ are the membership degree and the non-membership degree of *A*, as $\mu_A(x) \in [0,1]$ is the membership degree of $x \in X$ in *A* and $\upsilon_A(x) \in [0,1]$ is the non-membership degree of $x \in X$ in *A*.

Naturally is introduced $\pi_A(x)$ the degree of indeterminacy [13] of x to *A*, determined as:

$$\pi_A(x) = 1 - \mu_A(x) - \upsilon_A(x) \tag{3}$$

If $\pi_A(x) = 0$, for all $x \in X$, then the IFS *A* is reduced to a fuzzy set, else $\pi_A(x) > 0$, thus an indeterminacy occurs for the element x.

The similarity between two IFSs *A* and *B* is defines as [14]:

$$S(A, B) = 1 - \frac{|\mu_A(x) - \mu_B(x)| + |\upsilon_A(x) - \upsilon_B(x)|}{2} \tag{4}$$

with $S(A, B) \to [0,1]$ expressing the similarity degree.

3.3 Environment Definition

A service to manufacture S is a set of cloud services as $S = \{CS_1, CS_2, \dots, CS_i, \dots CS_N\}$. Then, let's consider $In_i = \{in_1^i, in_2^i, \dots, in_j^i, \dots in_{J_i}^i\}$, the set of input relative to the cloud service CS_i, where $i = 1,2, \dots, N$ and $j = 1,2, \dots, J_i$, with $N, J_i \in \mathbb{N}$. As well, $Out_i = \{out_1^i, out_2^i, \dots, out_k^i, \dots out_{K_i}^i\}$ is the set of output relative to CS_i, with $k = 1,2, \dots, K_i$ and $K_i \in \mathbb{N}$. Both input and output can globalize functional and non-functional parameters. The proposed approach is then, fully customizable and scalable to any type of cloud services.

Thus, an IFS $CSFin_i$ is introduced to characterize the degree of membership of the elements from the set In_i within CS_i based on eq.(2).

$$CSFin_i = \{\langle x, \mu_{CSFin_i}(x), \upsilon_{CSFin_i}(x)\rangle | x \in In_i\} \tag{5.a}$$

And a second IFS to characterize the membership of the elements from the set Out_i to CS_i.

$$CSFout_i = \{\langle x, \mu_{CSFout_i}(x), \upsilon_{CSFout_i}(x)\rangle | x \in Out_i\} \quad (5.b)$$

The objective is to evaluate the similarity for each element between a given CS_i and a set of service cluster eligible, within the same domain (e.g. power supply, journal bearing).

Therefore, we consider $SC_i = \{sc_1^i, sc_2^i, \dots, sc_m^i, \dots sc_{M_i}^i\}$ the set of service clusters associated to the same domain than CS_i. The two IFSs to characterize the elements x membership from the two sets In_i and Out_i within a given sc_m^i are defined as:

$$scFin_m^i = \{\langle y, \mu_{scFin_m^i}(x), \upsilon_{scFin_m^i}(x)\rangle \, | x \in In_i\} \quad (6.a)$$

and

$$scFout_m^i = \{\langle y, \mu_{scFout_m^i}(x), \upsilon_{scFout_m^i}(x)\rangle \, | x \in Out_i\} \quad (6.b)$$

3.4 Membership, Non-membership and Indeterminacy functions generation

The membership functions $\mu_{scFin_m^i}(x)$, $\mu_{scFout_m^i}(x)$ and the non-membership $\upsilon_{scFin_m^i}(x)$, $\upsilon_{scFout_m^i}(x)$ can be obtained through different methods; e.g. consult specialists, use predefined membership functions, sort of the membership functions automatically [15]. In our case they will express the degree of optimization and ownership of a given set of inputs or outputs to a given manufacturing resources. The advantage is to define the domain of capabilities and optimization for the whole set of service cluster candidate according to the cloud service definition.

However, for the sake of simplicity, we consider the intervention of several specialists to evaluate the membership of an element $x \in In_i$. That's why we setup the following matrix for membership and non-membership function generation process for a given sc_m^i and the set of input In_i and the set of output Out_i as:

$$InGen_m^i = \begin{bmatrix} ePos_{m,1}^i & eNeg_{m,1}^i & eInd_{m,1}^i \\ ePos_{m,2}^i & eNeg_{m,2}^i & eInd_{m,2}^i \\ . & . & . \\ ePos_{m,j}^i & eNeg_{m,j}^i & eInd_{m,j}^i \\ . & . & . \\ ePos_{m,J_i}^i & eNeg_{m,J_i}^i & eInd_{m,J_i}^i \end{bmatrix}_{3\times J_i} \quad (7.a)$$

$$OutGen_m^i = \begin{bmatrix} ePos_{m,1}^i & eNeg_{m,1}^i & eInd_{m,1}^i \\ ePos_{m,2}^i & eNeg_{m,2}^i & eInd_{m,2}^i \\ . & . & . \\ ePos_{m,k}^i & eNeg_{m,k}^i & eInd_{m,k}^i \\ . & . & . \\ ePos_{m,K_i}^i & eNeg_{m,K_i}^i & eInd_{m,K_i}^i \end{bmatrix}_{3\times K_i} \quad (7.b)$$

where $ePos_{m,j}^i$ is the number of positive evaluation for the membership of the element in_j^i in the IFS $scFin_m^i$ associated to the service cluster sc_m^i, $eNeg_{m,j}^i$ the number of negative evaluation (non-membership), and $eInt^{i}_{m,j}$ the number of indeterminacy; e.g. specialist who did not evaluate the parameter membership of in^{i}_{j}.

Therefore we can setup the functions $\mu_{scFin^{i}_{m}}(x)$, $\upsilon_{scFin^{i}_{m}}(x)$ and $\pi_{scFin^{i}_{m}}(x)$, $\forall x \in In_i$. Let's consider the case $x = in^{i}_{j}$, with $j = 1,2, \ldots ,J_i$:

$$\mu_{scFin^{i}_{m}}(in^{i}_{j}) = \frac{ePos^{i}_{m,j}}{(ePos^{i}_{m,j} + eNeg^{i}_{m,j} + eInd^{i}_{m,j})} \tag{8.a}$$

$$\upsilon_{scFin^{i}_{m}}(in^{i}_{j}) = \frac{eNeg^{i}_{m,j}}{(ePos^{i}_{m,j} + eNeg^{i}_{m,j} + eInd^{i}_{m,j})} \tag{8.b}$$

and

$$\pi_{scFin^{i}_{m}}\left(in^{i}_{j}\right) = \frac{eInd^{i}_{m,j}}{\left(ePos^{i}_{m,j} + eNeg^{i}_{m,j} + eInd^{i}_{m,j}\right)} = 1 - \mu_{scFin^{i}_{m}}(in^{i}_{j}) - \upsilon_{scFin^{i}_{m}}(in^{i}_{j}) \tag{8.c}$$

By analogy, we setup the functions $\mu_{scFout^{i}_{m}}(x)$, $\upsilon_{scFout^{i}_{m}}(x)$ and $\pi_{scFout^{i}_{m}}(x)$ using $OutGen^{i}_{m}$.

3.5 Similarity evaluation between CS_i and sc^{i}_{m}

We propose the following framework (Fig. 5.) to illustrate the similarity evaluation process between a given CS_i and sc^{i}_{m}. For the sake of readability, we only consider the set In_i. Our approach is to evaluate the similarity separately (heuristic approach) between all the elements from $\mu CSin_i$, $\upsilon CSin_i$ and $\mu scin^{i}_{m}$, $\upsilon scin^{i}_{m}$ respectively the set of membership and non-membership functions from the elements In_i in CS_i, and the set of membership and non-membership functions from the elements In_i in sc^{i}_{m}. Thus, the similarity evaluation for the element x_j is defined using eq.(4) as:

$$S\left(CSFin_i(\, x_j), scFin^{i}_{m}(\, x_j)\right) = 1 - \frac{\left|\mu_{CSFin_i}(x_j) - \mu_{scFin^{i}_{m}}(x_j)\right| + \left|\upsilon_{CSFout_i}(x_j) - \upsilon_{scFout^{i}_{m}}(x_j)\right|}{2} \tag{9}$$

Finally, the overall similarity between the IFS $CSFin_i$ and $scFin^{i}_{m}$ is linearized and computed as:

$$S^{i}_{m} = \sum_{j=0}^{J_i} [S\left(CSFin_i(\, x_j), scFin^{i}_{m}(\, x_j)\right) \times \omega_j] \tag{10}$$

with ω_j the weight associated to the importance of the element x_j as $\omega_j \in \mathbb{R}^{+}_{0}$ and $\sum_{j=1}^{J_i} \omega_j = 1$.

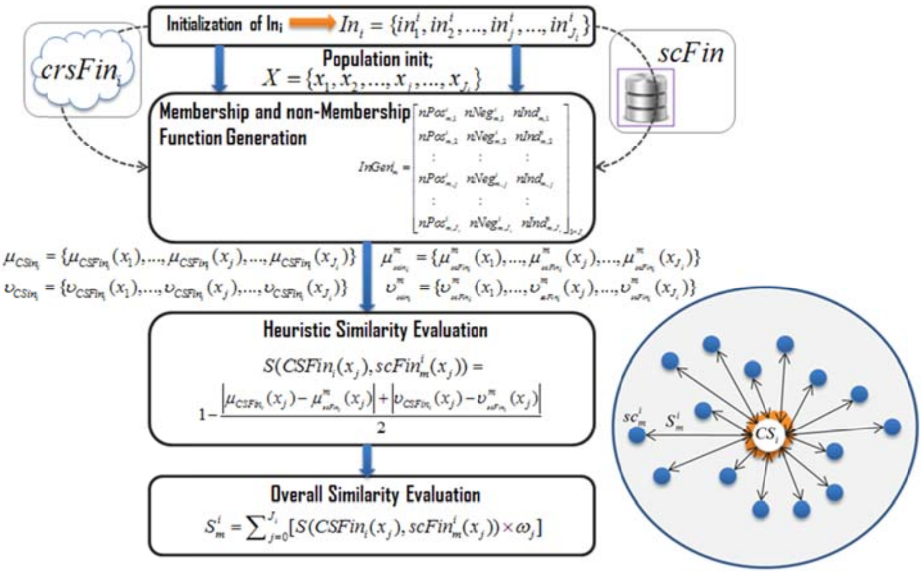

Fig. 5. Similarity evaluation Framework between Cloud service and Service cluster candidate based on IFS

4 Computational Experiments

4.1 Introduction to Artificial Bee Colony (ABC) optimization for similarity evaluation

Our first objective is to reduce the computational time for the similarity evaluation, and find the service cluster with the highest service cluster for each cloud service of the *N* cloud service chain. However, we deliberately avoid a full heuristic similarity evaluation. Instead our approach evaluates the whole chain of service clusters, allowing the integration of additional features (e.g. correlation analysis among service clusters). To realize this objective, we compute our method through Artificial Bee Colony (ABC) optimization.

For the best understanding, we only introduce the features of ABC, where a modification is needed to fit to our similarity evaluation between service clusters and cloud services.

ABC is one of the most recently introduced swarm-based algorithms, which present higher performances than ES (Evolution Strategies), GA (Genetic Algorithm), DE (Differential Evolution Algorithm) and PSO (Particle Swarm Optimization) [16].

ABC route execution is inspired by the behavior of honeybee swarm. The population of bees is divided into three categories:

(a) *Employed Bees*, who search for food sources and evaluate their nectar, in order to share their information in the hive with onlooker bees.

(b) *Onlooker Bees*, who position themselves on food sources presenting higher nectar amount.
(c) *Scout Bees*, who search to discover new food sources area.

The route of the algorithm is defined as [17]:

```
Initialization; set cycle
Repeat
    Place the employed bees on their food sources
    Place the onlooker bees on the food sources depending on
their nectar amount
    Send the scouts to search new areas for new food sources
    Memorize the best food source found so far
Until cycle=0
```

Here the food source represents the possible solution, which in our case is an eligible service cluster, and the nectar amount the fitness, which is related to the similarity of the service cluster to a given cloud service.

An onlooker bee selects its food source according to the probability value p_m^i associated with that food source as:

$$p_i = \frac{fit_i}{\sum_{n=1}^{SN} fit_n} \tag{11}$$

with SN the population number equal to the number of possibilities, and fit_i the fitness value of the solution *i*. In our case:

$$SN = \prod_{i=1}^{N} M_i \tag{12}$$

$$\forall m \in [1, M_i]; fit_i = \sum_{i=1}^{N} S_m^i \tag{13}$$

Thus, the objective is to minimize fit_i, $\forall i \in [1, N]$ and $\forall m \in [1, M_i]$.

In ABC, as the search approaches the optimal solution in the given population of service clusters, the research area is adaptively reduced.

There are three controlled parameters to setup the search and evaluation environment:

(a) The *limit* is the number of cycle, during which one, each bee will search for better food sources in its neighborhood. If the fitness is not improved by then; the food source is abandoned.
(b) The *NP*, the number of colony size (employed bees + onlooker bees).
(c) *MCN* (Maximum Cycle Number) set up the number of time the sequence of foraging will last.

These parameters are settled arbitrarily, and can influence the performances of the algorithm significantly. But as an advantage, ABC has only three [18].

4.1 Performance Evaluation

We evaluate the performances of our method optimized through ABC, and the same similarity evaluation method using LP (linear programming) through all the possible solutions, enabling us to identify the optimal solution. During these experimentations we study the performances by modifying the numbers M_i of service clusters per cloud services, while the number of parameters N, J_i of In_i and K_i of Out_i are set to 10. For all the run, the setup parameters of ABC are $NP = 2N$, *Limit=50*, and $MCN = NP \times \overline{M_\iota}$.

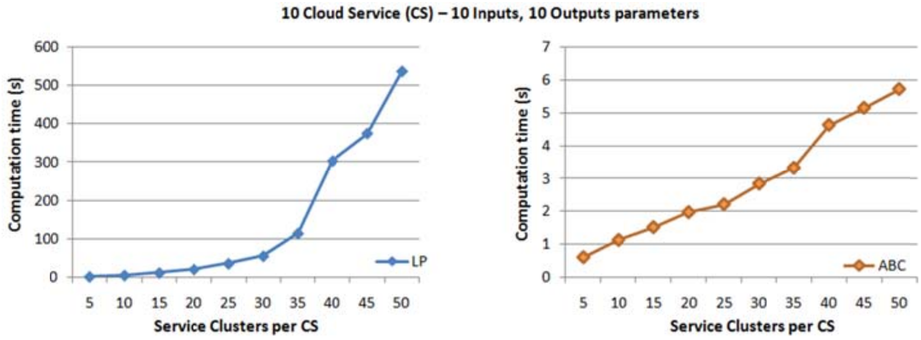

Fig. 6. Computational time comparison for Similarity evaluation using LP and ABC

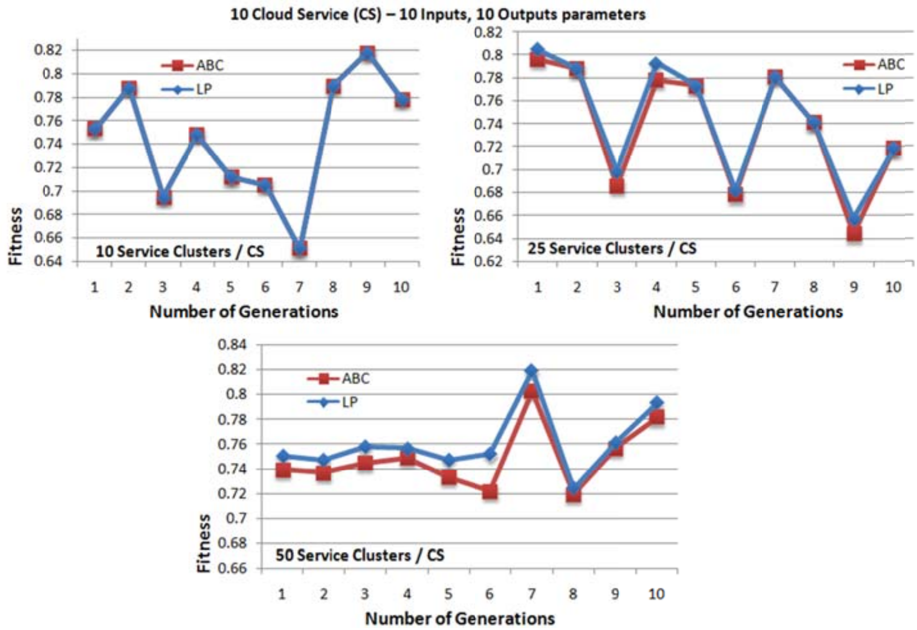

Fig. 7. Fitness comparison for Similarity evaluation using LP and ABC

We conducted these experiments on a computer equipped with an Intel Core i3-2100 3.10GHz processor and 4GB RAM. The machine is running under Windows 7 pro and Java 1.7.

We can easily remark that the LP method becomes unrealistic while the problem size is increasing (Fig. 6.). The computational time is growing exponentially toward the number of possible solutions, whereas the ABC optimization offers a faster computation, especially for large scale problem.

However, the LP method consisting to browse all the solutions to match the optimal one presents the best fitness possible. While the problem size is increasing our similarity evaluation ABC optimized shows a distance between its best fitness and the optimal fitness (Fig. 7.). The discrepancy is linked to the number of MCN. Since ABC is based on a probability selection process, it is impossible to define a sure value of MCN according the problem inputs. Nevertheless, the CMfg system can train a neural network aiming to define the best MCN. But of course, MCN has a strong influence on the computational time.

Therefore, the advantage of our similarity evaluation ABC optimized is to scale the computational time and the quality of the fitness evaluation, according to the equipment restriction and / or quality fitness requirements.

5 Conclusion & Future work

The research work presented in this paper proposes a method to evaluate the similarity between service clusters and cloud services, to match the service cluster with the highest similarity value, according to a set of definition (In_i and / or Out_i). The ABC optimization offers satisfying computational time for the proposed method.

However, this method has to be considered in the whole problematic of cloud service composition. As mentioned, this process represents the core value of the CMfg system, enabling the creation of services and innovations. Therefore, the CMfg system has to select the best matches toward the functional parameters of the cloud services to manufacture, and also the non-functional parameter as QoSs, to satisfy the demander's requirements. In this purpose, our method represents an entry point to the whole composition process, enabling to select the best service cluster. Thus, a strategy concerning the evaluation of composite resource service within the same service cluster must be established. Since our method is mainly designed to insure the coordination among the requested inputs and outputs for a given cloud service, a strategy more QoS-aware oriented will be a relevant point to ponder.

Acknowledgement This work has been partly funded by the MOST of China through the Project Key Technology of Service Platform for Cloud Manufacturing. The authors wish to acknowledge MOST for their support. We also wish to acknowledge our gratitude and appreciation to all the Project partners for their contribution during the development of various ideas and concepts presented in this paper.

References

1. Fei Tao, Ye Fa Hu, Zu De Zhou. *Study on manufacturing grid and its resource service optimal-selection system*. Int. J. Adv. Mfg Technology. (May 2008), 37: 1022-1041.
2. F.Tao, L.Zhang, V.C.Venkatesh, Y.Luo and Y.Cheng. *Cloud manufacturing: a computing and service-oriented manufacturing model*. Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, (August 2011),pp.1-8

3. Guanjie Ren and Mike Gregory. *Servitization in manufacturing Companies – Literature Review, Research Progress and Cambridge service Research*. Cranfield Product-Service Systems Seminar, (November 2007).
4. Andy Neely, Ornella Benedetinni and Ivanka Visnjic. *The servitization of manufacturing: Further evidence*. 18th Euro. Ops. Management Association Conf., Cambridge, (July 2011).
5. Mark Loughridge. *IBM Financial Model*. IBM Investor Briefing, (2010).
6. Fanghua Ning, Weizong Zhou, Fengying Zhang, Qian Yin, Xiajing Ni. *The Architecture of Cloud manufacturing and its key technologies research*. Proceedings of IEEE Cloud Computing and Intelligence Systems (CCIS), (September 2011), pp.259-263.
7. Buyya, Rajkumar; Chee Shin Yeo, Srikumar Venugopal. *Market-Oriented Cloud Computing: Vision, Hype, and reality for delivering it services as Computing Utilities.* University of Melbourne, Australia, (2008), 07:7-9
8. Xun Xu. *From Cloud computing to Cloud manufacturing.* Rob. and Comp. Mfg, (July 2011)
9. Tehani Mou, Lanshun Nie, Dechen Zhan, and Xiaofei Xu. *Task Scheduling and Assignment Methods for Cloud Enterprises*. Enterprise Interoperability V: (2012), pp.427-437
10. L. A. Zadeh. *Fuzzy sets*. Information and control. (1965), 8:338-553
11. Lily R. Liang, Shiyong Lu, Xuena Wang, Yi Lu, Vinay Mandal, Dorrelyn Patacsil, Deepak Kumar. *FM-test: A Fuzzy-Set-Theory-Based Approach to Differential Gene Expression Data Analysis*. BMC Bioinformatics, (2006) vol.7:(4)
12. K. T. Atanassov. "*Intuitionistic Fuzzy Sets*". Fuzzy sets and Systems, (1986), pp.87-96
13. Fei Tao, Dongming Zhao, Lin Zhang. *Resource service optimal-selection based on intuitionistic fuzzy set and non-functionality QoS in manufacturing grid system*. Knowledge and Information Systems, (2010), 25(01):185-208
14. Z. Xu. "*Some similarity evaluations of intuitionistic fuzzy sets and their applications to multiple attribute decision making*", Fuzzy Optim. Decision Making (2007), 6(2):109-121
15. Alan C. Genari, Denise Guliato. *Similarity evaluations based on fuzzy sets*. Federal university of Uberlandia, Brazil
16. Dervis Karaboga, Bahriye Akay. *A comparative study of Artificial Bee Colony algorithm*. Applied Mathematics and Computation, (August 2009), 214(1):108-132
17. Dervis Karaboga, Beyza Gorkemli, Celal Ozturk and Nurhan Karaboga. *A comprehensive survey: artificial bee colony (ABC) algorithm and applications*. Artificial Intelligence Review, (March 2012)
18. R. Anandhakumar, S. Subramanian, and S. Ganesan. *Modified ABC Algorithm for Generator Maintenance Scheduling*. Int. J. of Comp. and Elec. Engineering. (2011), 3(6):812-819