Adv. Manuf. (2021) 9:81–103
https://doi.org/10.1007/s40436-020-00331-0

Intelligent modular design with holonic fuzzy agents

Egon Ostrosi[1] · Alain-Jérôme Fougères[2] · Zai-Fang Zhang[3] · Josip Stjepandić[4]

Received: 14 September 2019 / Revised: 2 May 2020 / Accepted: 13 November 2020 / Published online: 25 January 2021
© Shanghai University and Springer-Verlag GmbH Germany, part of Springer Nature 2021

Abstract Presently, modular designs use various technologies accompanied by multiple models. Although no integral solution is known, a plethora of approaches is used to resolve this trans disciplinary challenge, often by using local intelligence. However, the effective utilization of multiple models requires proper integration for them to work together as a cohesive system. This requirement calls for the development of intelligent models and tools that can be used for the development of intelligent modular products. Modular design based on these intelligent models and tools is called intelligent modular design. Intelligent modular design requires to be considered both dynamically and holistically by combining customer requirements, product functions, solutions, service specifications, and their fuzziness in order to structure a product into intelligent modules. This paper proposes the use of holonic fuzzy agents to fulfill both the properties of intelligent models and the requirements of intelligent modular design. The set of fuzzy function agents and their corresponding fuzzy solution agents are found from customization of the product-service system in the fuzzy function agent-fuzzy solution agent sub-network. On the basis of attractor agent recognition, the fuzzy function and fuzzy solution agents interact to form the holonic fuzzy module agents. Self-embedding of holonic fuzzy module agents, which is the fundamental property of the holonic structure, is also characterized by vertical and horizontal communication. The flexibility and agility of the software agent make the holonic structure of intelligent modules adaptable. An application illustrates the proposed intelligent modular design.

Keywords Modularity · Modular design · Intelligent design · Holonic agents · Product platform · Product variety

1 Introduction

Modularity must be an integral part of the development of a sustainable and intelligent design. The use of modularity in product design includes its technical and business aspects from both quantitative and qualitative angles. Human cognitive abilities are considered to be the roots of modularity [1].

A product consists of several components that are connected to work as a whole unit. A product is deemed modular when its components and sub-modules can be combined seamlessly [2]. At the same time, each module is also required to work independently of the other modules, and the lack of connectivity between its components defines the modularity [3]. Similar interconnected physical components are encapsulated to form the modules. The flow of energy, material and information between components of modules allows them to meet functional requirements. The modules are characterized by the minimal interaction with external components and the maximum interaction between components within the module. The

---

✉ Egon Ostrosi
egon.ostrosi@utbm.fr

[1] University of Technology of Belfort Montbéliard, ERCOS/ELLIADD EA4661, University Burgundy - Franche-Comté, 90010 Belfort, France

[2] School of Engineering, ECAM Rennes, Bruz CS 29128, 35091 Rennes, France

[3] School of Mechatronic Engineering and Automation, Shanghai University, Shanghai 200240, People's Republic of China

[4] PROSTEP AG, Dolivostr.11, 64293 Darmstadt, Germany

functionality of the module is limited to predefined scenarios that control its functionality using interface and connections, and can be controlled in three ways: the interface between various components within the module, the interconnection with nonadjacent components within the product, and the behavior of components as a bridge between other components via interfaces.

With regard to product architecture, there are principally two types of design: integral design and modular design. Integral design defines a product structure where the functions of the product are fulfilled by a range of individual parts with a one-to-one arrangement. A consequence of this is the lack of stability of the structure: any change to parts tends to affect most parts around them. This has implications for assembly, maintenance, servicing, and reusability. In contrast to integral design, the challenge of modular design lies in the definition of a product architecture that consists of several modules where the functions of the product are distributed among singular modules that provide their specific function (e.g., power unit) and are reusable in different products. Compatibility, interfaces (where modules interact), and integration are the most important impact factors in creating a successful modular design. Because there is no known universal solution, this multidisciplinary task needs to be solved for each modular product.

Modular product design can be broken down into three different and potentially complementary activities: design of modules, identification of modules and design with modules (see Fig. 1) [4]. The design of modules comprises the definition of functional carriers and interfaces that build a module (e.g., a drive unit). Identifying modules represents an "ex post approach" to clustering existing functional carriers into modules afterwards. Design with modules or "construction kit design" comprises the design of a product using existing modules.

Product architecture consists of some hierarchical levels, adding a useful dimension to modularity analysis. Designing product architecture that maximizes its overall modularity over these levels of the product structure is one solution [5].

Intelligent products aim to provide flexibility in design to the architects and system designers. The methodology used in the design of intelligent products is important. The handling of modularity and its management requires intelligent multidisciplinary collaboration and distributed platforms. This challenge can be the best met by using the multiagent paradigm and providing innovative technologies that can handle the dynamic environment.

A holon, consisting of interrelated semi-autonomous modules, is an advantageous way of modeling modular intelligent systems. Each interrelated semi-autonomous module in a holon has a hierarchical structure ending with elementary modules. The concept of the attractor specific to a holon is used to deal with the form and stability of the module [6]. An attractor is a stable element or group of elements toward which a module formation tends to evolve. This study uses product functions or physical product solutions as attractors. Thus, the functions and solutions of a product are identified. This means that functions and solutions are designed as entities capable of exhibiting intelligent behavior. The function agents interact with the solution agents to form holarchic function-solution sub-communities called holonic modules with the holonic structure. Intelligent holonic modules are emerged sub networks of holonic fuzzy agents.

This paper is organized as follows: Sect. 2 analyzes the state-of-the-art of modular design from different perspectives; Sect. 3 presents the formal model for holonic modules; Sect. 4 presents the formation of holonic modules with holonic fuzzy agents; Sect. 5 demonstrates the application of the proposed model; and Sect. 6 summarizes this research study's conclusions and perspectives.

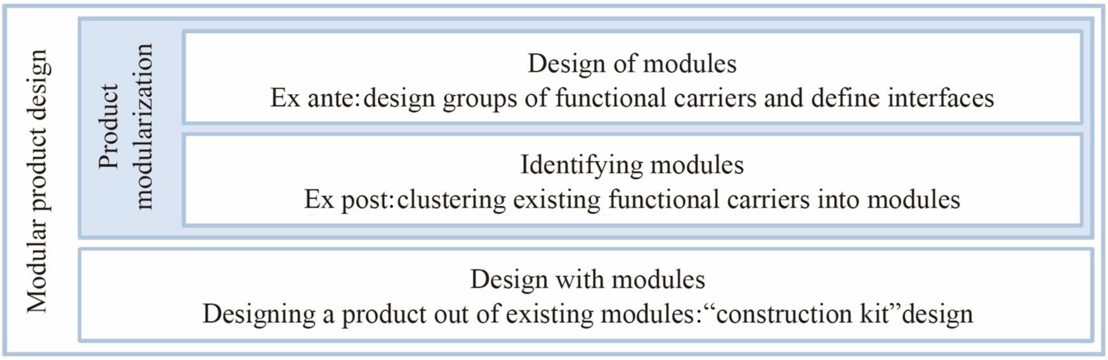


Fig. 1 Modular product design approaches [4]

2 Modular design: the state-of-the-art

In this section, we present recent trends and shortcomings in modular design research by examining four perspectives: technologies, product-service system, artificial intelligence and industrial applications.

2.1 Modular design and technologies

The modular design of a product specifies the functions, properties and interfaces of its constituents. The use of a standard interface allows parts to be interchangeable, thereby reducing the overall cost of combining the modular constituents.

The information is partitioned into three categories to achieve modularity [7]: architecture, interfaces and standards. The modules' architecture provides module functions and specifications, and their interaction is defined by the interface. The modules' conformity to design specifications is ensured by conducting standard tests. The rule-based scheme is used for the generation of modular product architecture by defining functions corresponding to the physical components of the module. The module interfaces are crucial for proper utilization of modules. Modular product attributes include commonality of modules, combinability of modules, function binding, interface standardization and loose coupling of components [2, 8].

The processes mentioned below are used for modular design [8]: identification of product architecture and reusable components from existing products, agglomeration and adaptation of singular building blocks into modules to derive a new design, and assessment of product performance and cost.

Researchers have suggested various modular design strategies such as functional modeling [9], axiomatic design [10], design structure matrix [11], and modular function deployment [12]. Variant mode and effects analysis (VMEA) [13] can also be used, along with the architecture development process [14]. Comparison of modular design methods in application areas such as product generation, product variety, and product lifecycle reveals that selection of the method to be used requires taking a variety of issues into account.

The integration of the holistic lifecycle characteristics of a product's architecture is a critical issue [15]. The product architecture depends on its configuration and modularity [16]. The lifecycle design must take into account the product architecture. However, because the industry has considered it as intellectual knowledge instead of a scientific-engineering issue, the theoretical foundation required for developing modular product architecture is missing. The modularity concept is also followed while designing a family of products and platform-based product development [17]. Use of modular design in the platform is of special interest in an industrial environment. The platform provides the engineered base for the development of new products using standardized subsystems, modules, and components. To enable the development of a unique end product, the platform includes the basic architecture and interfaces of the optional items. A platform requires criteria on and identification of new platform elements, which are termed design assets. These are introduced as a means to enable diverse types of resources to be reused by a company and provide a pragmatic way to bridge the gap between the physical products and the knowledge, tools, and methods needed to realize them [18]. The holistic view of modular design necessitates the inclusion of both back-end and front-end issues, including product family, product portfolio, manufacturing and production, platform-based product family design, and supply chain management [19].

The current trend is to use tools like product configurators for the development of the modular design [20]. Using interaction and mutual integrations, these tools provide optimal functionality. The product configurator, a multifunctional commercial tool, also interfaces with delivery and sales in an enterprise environment. The configurator tool ensures compliance with the specifications defined for the model and, using the configurator, the product logic, based on rules and limitations, is implemented. The customer provides his/her requirement through the user interface, and then, the product that meets his requirement is selected. After analysis of the validity and cost of the chosen models, the bill of materials (BOM) is drawn up. The integration of product configuration with other IT systems, such as computer-aided technologies (CAT), product life cycle management (PLM), and enterprise resource planning (ERP), is required. However, the management and synchronization of configuration data with the above-mentioned IT applications require proper attention to be paid to the deployment of integrated product configuration.

In recent product data management (PDM) systems, the modular product's structure is mapped onto a common structure [17, 21]. The PDM systems' database manages optional items, such as other items in the master records and their attributes. The management of an article depends on the bill of quantities required. Thereafter, using order neutral BOMs along with varying and optional positions, the PDM systems handle changes in product structure. This methodology allows for the creation of the explicit BOMs required for production and also benefits product development. However, managing large variants of the products makes data management complicated and requires more risk tolerance. These issues are taken care of by using the variant manager module in modern PDM systems. The base module manages data such as parts, processes, structures, and explicit variants are managed using configuration and clone modules. The report generation module is used for generation of reports as and when they are required.

2.2 Modularity and product-service system

From a management perspective, the modularity can be taken as a strategy for the efficient design of complex products, procedures, and services because it brings rationalization into the system. The modularity may be taken as a product development methodology that is used for several product designs [22, 23].

For products with very distinctive features, modularity is not required because creativity is necessary to produce specialized haptics, stylings, or colors. However, because the number of products requiring distinctive features is small, the modular design is not cost-effective. By contrast, productivity gains by the implementation of product modularity at a bus manufacturer are significant: an increase of 18.40% in the efficiency indexes of the company's products engineering and a 63.50% increase in the total factor productivity index (TFPI) [24]. Only a holistic controlling approach for the integration of product variants can lead to monetary benefits. Using a modularity-balanced-score-card (M-BSC), the holistic approach allows assessment and holistic management of modular product families. The modular product development also requires taking into account perspectives for the integration of development, production, marketing sales, and services. The costing of modular products can be done by looking at the cost of generic modules used for product development.

The mass customization uses modular design to create complex product and service offerings on-demand to meet specific customer requirements [25]. Mass customization is an amalgamation of mass production and customization. The two approaches used for mass customization are mass and craft (single piece) production. On the one hand, mass production uses standardization and scales of economies to create cost-effective products. Craft production, on the other hand, uses a high level of customization to meet specific customer requirements.

The creation of customized products calls for using specifically created components along with standardized and configurable modules. This requires identifying a fixed and variable area of the product structure. Then, the variable area is used for customization. Generally, configuration systems are used for product customization. Mass customization of product design requires the use of generic conceptual procedures [26]. These generic conceptual procedures require business process analysis and redesign, company product portfolio analysis and modeling, configuration software selection, software programming, and the implementation as well as further configuration of system development toward product-service system (PSS).

2.3 Modular design and artificial intelligence

Artificial intelligence allows the creation of efficient computing techniques. Evolutionary algorithms are used to identify separable modules and, simultaneously, to optimize the number of modules [27]. At the same time, they minimize the variation in complexity allocation to individual modules [28], product configuration based on modules [29], and consideration of the product lifecycle [30].

The use of a multi-objective grouping genetic algorithm can create several alternative product modularizations [31]. Likewise, it is shown that modules could be identified using neural network algorithms [32]. The agent paradigm can be used to solve complex uncertain problems with inherently distributed global knowledge shared among several agents [33]. Owing to its distribution and decentralization, the flexibility of the agent-based system allows it to be used in an industrial environment [34]. Multiagent systems have been found to be useful in modeling and simulation of adaptive or self-adaptive systems such as emergence and self-organization [35]. The use of reactive and autonomous agents allows for updating and use of information from other agents and their environment.

Product design configuration uses fuzziness in the early-stage of the product lifecycle design [36, 37]. Fuzzy agents optimize and model the fuzziness information model, fuzziness knowledge, and fuzziness interaction in distributed and collaborative design [38]. In order to be of interest to each other, the fuzzy agents process fuzziness information after receiving it and then interact within a multiagent system. The use of configuration grammars allows the formalization of configuration structural problems [39] and implementation is done in a grammar-based, multiagent platform [40]. The decision required in modular product collaborative design can also be supported by agent-based systems with a large knowledge base [41].

2.4 Modularity in industrial applications

Dilemmas, doubts, and uncertainties emerge during modular design; for example, which of the following is the right method to apply for modularization: function-based modularization, component-based modularization, or function-component modularization. These choices present dilemmas for engineers in product modularization [42].

An analysis of engine blocks modular design shows that it derives from functional knowledge and its proper restructuring by the designer [42]. The modularization approaches for conventional products versus large-scale products are dealt with differently. However, the large-scale modules can be subdivided into small scale modular design so that these modules can be used for both small scale and large-scale products [43]. The ideal module size required for a particular product can be determined using the following four steps: design, technical feasibility, economic viability and tool development. Small module size facilitates manufacturing and logistics. In the case of an airplane, for example, the increase in the number of modules leads to arise in fuel consumption because a higher number of interfaces is required between the modules. The predicted lifecycle costs provide a proper measure for determining the ideal module size while the minimization of total lifecycle costs leads to global optimization. Eco-modular product architecture also contributes to enhancing product recovery processes through recycling and reusing modules without full disassembly at the component or material levels. This leads to less consumption of natural resources and less landfill damage to the environment [44].

Modular design uses knowledge-based technologies integrated with numerical technologies. PDM and 3D-CAD systems are used for designing and documenting plant designs when machines have more than 10 000 parts [20]. The design is customized according to customer and market requirements, the technical feasibility of the product, business requirements, and manufacturing ability. In other words, the design should not reduce the offering or have unnecessary complexity. A web-based product configurator can be used for the proper product, whereas knowledge-based engineering (KBE) and CAD models are used for product presentation. The KBE and CAD models are highly adaptable, and the product configurator is used to provide the product variant required by the user. The configurator communicates directly with various sections, including internal systems (e.g., ERP, customer relationship management (CRM) and PDM) without involving the CAD system. This methodology enables the generation of bottom-up relationship knowledge and use of the ERP object list for setting up assembly plans. For example, CNC machine design [45] or components' design in the automobile sector [46] use product configuration with KBE. Original equipment manufacturers (OEMs) in the automotive sector have been using configuration technologies and tools for a long time. However, both PLM- and ERP-based standard applications do not provide the functionality required for product configuration. The ERP system provides tools for business operations, whereas the PLM system provides tools for product development. By supporting both designing and off-the-shelf solutions, the design platform (DP) approach provides a coherent environment for heterogeneous and transdisciplinary design resources to be used in product development. For example, it can be used at an automotive supplier to support the development of customized solutions when traditional modularity or platform scalability do not suffice [47]. A computer tool has been developed to support the creation and visualization of the DP. The support tool has a connection to a PDM database to link the platform model to the various kinds of engineering assets needed or intended to support variant creation.

2.5 Modularity and further development

Modular product design removes system-based interdependency hurdles, providing a rational and coherent design process that takes care of constraints and technical requirements at the beginning of the process. Because judging the relationship between effectiveness and level of modularity is an open-ended issue, modularity assessment provides ways to increase the effectiveness of modularity.

A flexible product that has good adaptability requires extra effort in its design and manufacturing. The design and development of intelligent systems require modules to have adaptability and suitability that can enable integration with different types of large systems. An intelligent product maximizes the design space for system designers and architects. As modular products are upgradable they facilitate the approaches of systems engineering [48].

The current trend in modular design uses technologies, such as product configurators, advanced CAD systems, PDM systems and agent-based systems, in an integrated form. The intelligent models, tools, and products require holistic and intelligent engineering approaches for cooperation and communication with each other. Self-sustainable models and products can be designed using these approaches.

Thus, the design of intelligent products necessitates the use of intelligent modules. The intelligent modules should overcome the distinction of the continuous-discontinuous traditional modular formation problem. In order to do that, the development of intelligent modular products requires intelligent models and intelligent tools. The modular design based on these intelligent models and tools is the intelligent modular design. The modularity and variant management can be handled nicely using intelligent collaborative and distributed platforms.

The holonic and multiagent model has demonstrated the potential to take up this challenge and provide innovative technologies that can tackle the challenges involved in modular product development. Furthermore, intelligent modular design dynamically and holistically considers: customer requirements, product functions, solutions, process or service specifications, their fuzziness, and structures the product into intelligent modules. This study uses holonic fuzzy agents to satisfy the properties of intelligent models and the requirements of modular design.

3 Formal model for holonic modules

Mathematically, the search for the holonic module is transformed into the search for the holonic decomposition of networks. We assume that a module is a holon in an uncertain environment. Based on this assumption, equations for holonic modules and holonic fuzzy agent modeling are presented.

3.1 Equations for holonic modules

A product is described by engineers using its decompositions. The decompositions of a product can be defined in many ways and depend on the information that requires highlighting [49]. The structural or functional properties of a product are part of its description and allow the decomposition of its properties. A new set of decomposition can be created by adding new engineering properties.

The decomposition of product properties defines the module. Thus, a module can be considered as a whole that is a part of a vaster module, which, at the same time, contains sub-modules of which it is composed and which provides its structural and functional meaning [6]. The module's decomposition ability, together with the holon definition, outlines its complete property. The double heading indicates the inclusion of the holons in the holarchy, which is a typical vertical arrangement. The modules in nested hierarchical order and progressive accumulation can be represented by a holarchy. Owing to their being self-completeness, holons are considered as a unit, but they can also be part of a larger system.

Many applications, including holonic systems design [50], holonic manufacturing [51–53], manufacturing control and scheduling [54, 55], holonic assembly [56, 57], and product design [58], use the concept of the holon.

Three cases of holonic modularization can be distinguished: physical solution (or component) based modularization, function-based modularization, and function-physical solution-based modularization.

The first case concerns the equation governing the fuzzy set of physical solutions and their relationships. Given the fuzzy set of physical solutions $\boldsymbol{S}$, the fuzzy relationship between the elements of $\boldsymbol{S}$ can be characterized by different degrees of affinity. This dependency can be written as

$$\tilde{\boldsymbol{S}}^{k=0}=\tilde{\boldsymbol{C}}^{k=0}\tilde{\boldsymbol{S}}^{k=0}, \tag{1}$$

where $\tilde{\boldsymbol{S}}^{k=0}$ is the initial vector $(k=0)$ representing the fuzzy physical solutions $\tilde{\boldsymbol{S}}^{k=0}=\{\tilde{s}_1^0,\tilde{s}_2^0,\cdots,\tilde{s}_n^0\}$ and $\tilde{\boldsymbol{C}}^{k=0}$ is the initial matrix representing the fuzzy affinity relationship between physical solutions.

To describe the affinity relationship between physical solutions, a fuzzy relationship between physical solutions $\tilde{\mathscr{R}}_1\left(\tilde{s}_j^{k=0},\tilde{s}_{j'}^{k=0}\right)$ can be defined. The fuzzy relationship $\tilde{\mathscr{R}}_1\left(\tilde{s}_j^{k=0},\tilde{s}_{j'}^{k=0}\right)$ represented by the matrix $\tilde{\boldsymbol{C}}^{k=0}$ is characterized by the membership function $\mu_{\tilde{\mathscr{R}}_1}\left(\tilde{s}_j^{k=0},\tilde{s}_{j'}^{k=0}\right)$, which takes values between 0 and 1. In practice, the affinity relationship between solutions may be either apparent or unclear. Hence, the designer could use an intermediate degree between 1 and 0 to quantify it. The network, called a fuzzy structural network, represents the affinity relationship $\tilde{\mathscr{R}}_1\left(\tilde{s}_j^{k=0},\tilde{s}_{j'}^{k=0}\right)$ and is characterized by the fuzziness. The holonic decomposition of the fuzzy structural network, corresponding to the holonic decomposition of the matrix $\tilde{\boldsymbol{C}}^{k=0}$ into sub-matrixes, will yield the holonic decomposition of the product into fuzzy structural module holons

$$\tilde{\boldsymbol{S}}_j^k=\tilde{\boldsymbol{C}}_j^k\tilde{\boldsymbol{S}}_j^k, \tag{2}$$

where $\tilde{\boldsymbol{S}}_j^k$ is the fuzzy solution holon j corresponding to the level $k>0$ of the holarchy of fuzzy holon solutions; $\tilde{\boldsymbol{S}}_j^k=\left\{\tilde{S}_1^{k-1},\tilde{S}_2^{k-1},\cdots,\tilde{S}_p^{k-1}\right\}$; $\tilde{\boldsymbol{M}}_j^k=\left\langle\tilde{S}_j^k,\tilde{S}_j^k,\tilde{C}_j^k\right\rangle$ is a fuzzy structural holon module with $\tilde{\boldsymbol{C}}_j^k:\tilde{\boldsymbol{S}}_j^k\leftrightarrow\tilde{\boldsymbol{S}}_j^k$.

The second case concerns the equation governing the fuzzy set of functions and their relationships. Functions and their inter-relationships that involve decomposition and dependency can be represented by the functional structure of a product. This dependency can be written as

$$\tilde{\boldsymbol{F}}^{k=0}=\tilde{\boldsymbol{B}}^{k=0}\tilde{\boldsymbol{F}}^{k=0}, \tag{3}$$

where $\tilde{\boldsymbol{F}}^{k=0}$ is the initial vector representing the fuzzy functions $\tilde{\boldsymbol{F}}^{k=0}=\{\tilde{f}_1^0,\tilde{f}_2^0,\cdots,\tilde{f}_m^0\}$ of the level $k>0$ of the holarchy of fuzzy holon functions; $\tilde{\boldsymbol{B}}^{k=0}$ is the matrix representing the dependency relationship between fuzzy functions.

The degree of interaction characterizes the product's fuzzy functions set $\tilde{\boldsymbol{F}}^{k=0}$. These interactions can be defined by a fuzzy relationship $\tilde{\mathscr{R}}_2\left(\tilde{f}_j^{k=0},\tilde{f}_{j'}^{k=0}\right)$ represented by the matrix $\tilde{\boldsymbol{B}}^{k=0}$. In this instance, the membership function $\mu_{\tilde{\mathscr{R}}_2}\left(\tilde{f}_j^{k=0},\tilde{f}_{j'}^{k=0}\right)$ can have a value between 0 and 1. The value of 1 for membership confirms identification of the relationship between product functions by the designer, whereas a value of 0 indicates lack of identification. When a functional network takes characteristics of fuzziness, it is termed a fuzzy functional network. The fuzzy functional network's holonic decomposition is a holonic decomposition of the matrix $\tilde{\boldsymbol{B}}^{k=0}$ into sub-matrixes. This decomposition provides holonic decomposition of the product into fuzzy functional module holons

$$\tilde{\boldsymbol{F}}_j^k = \tilde{\boldsymbol{B}}_j^k \tilde{\boldsymbol{F}}_j^k, \tag{4}$$

where $\tilde{\boldsymbol{F}}_j^k$ is the fuzzy function holon j corresponding to the level $k > 0$ of the holarchy of fuzzy holon functions; $\tilde{\boldsymbol{F}}_j^k = \left\{\tilde{F}_1^{k-1}, \tilde{F}_2^{k-1}, \cdots, \tilde{F}_p^{k-1}\right\}$; and $\boldsymbol{M}_j^k = \langle \tilde{\boldsymbol{F}}_j^k, \tilde{\boldsymbol{F}}_j^k, \tilde{\boldsymbol{B}}_j^k \rangle$ is a fuzzy functional holon module with $\tilde{\boldsymbol{B}}_j^k : \tilde{\boldsymbol{F}}_j^k \leftrightarrow \tilde{\boldsymbol{F}}_j^k$.

The third case is related to equations defining fuzzy functions and their relationship with fuzzy physical solutions. Mapping is done to move to the physical domain from the functional domain. The mapping between the functional domain and the physical solution is given by the expression

$$\tilde{\boldsymbol{F}}^{k=0} = \tilde{\boldsymbol{A}}^{k=0} \tilde{\boldsymbol{S}}^{k=0}, \tag{5}$$

where $\tilde{\boldsymbol{F}}^{k=0}$ is the initial vector representing the fuzzy functions $\tilde{\boldsymbol{F}}^{k=0} = \{\tilde{f}_1^0, \tilde{f}_2^0, \cdots, \tilde{f}_m^0\}$ of the level $k = 0$ of the holarchy of fuzzy holon functions; $\tilde{\boldsymbol{S}}^{k=0}$ is the initial vector representing the fuzzy physical solutions $\tilde{\boldsymbol{S}}^{k=0} = \{\tilde{s}_1^0, \tilde{s}_2^0, \cdots, \tilde{s}_n^0\}$ and $\tilde{\boldsymbol{A}}^{k=0}$ is the initial design matrix representing the mapping between fuzzy functions and fuzzy physical solutions.

The set of functions of each product can be taken care of by using various physical solution sets or, alternatively, can be partially taken care of by using physical solution sets. This shows that the mapping between the fuzzy set of alternative physical solutions and the fuzzy set of functions is also fuzzy in nature. The fuzzy matrix $\tilde{\boldsymbol{A}}^{k=0}$ represents a fuzzy relationship $\tilde{\mathscr{R}}_3\left(\tilde{f}_i^{k=0}, \tilde{s}_j^{k=0}\right)$. The fuzzy relationship provides relations between the fuzzy set of functions $\tilde{\boldsymbol{F}}^{k=0} = \{\tilde{f}_1^0, \tilde{f}_2^0, \cdots, \tilde{f}_m^0\}$ and the fuzzy set of physical solutions $\tilde{\boldsymbol{S}}^{k=0} = \{\tilde{s}_1^0, \tilde{s}_2^0, \cdots, \tilde{s}_n^0\}$. The corresponding membership function $\mu_{\tilde{\mathscr{R}}_3}\left(\tilde{f}_i^{k=0}, \tilde{s}_j^{k=0}\right)$ defined in [0, 1] provides the extent to which a product's function can be met using a set of alternative physical solutions. The membership function $\mu_{\tilde{\mathscr{R}}_3}\left(\tilde{f}_i^{k=0}, \tilde{s}_j^{k=0}\right)$ provides the designer's view of available product functions in relation with a set of alternative physical solutions. In this manner, the fuzzy relationship between product functions and related physical solutions provides the designer's perspective. The fuzzy functional-structural network is a function-solution network developed using fuzziness.

The fuzzy functional-structural network's holonic decomposition generates fuzzy functional-structural module holons

$$\tilde{\boldsymbol{F}}_j^k = \tilde{\boldsymbol{A}}_j^k \tilde{\boldsymbol{S}}_j^k, \tag{6}$$

where $\tilde{\boldsymbol{F}}_j^k$ is the fuzzy function holon j corresponding to the level $k > 0$ of the holarchy of fuzzy function holon; $\tilde{\boldsymbol{F}}_j^k = \left\{\tilde{\boldsymbol{F}}_1^{k-1}, \cdots, \tilde{\boldsymbol{F}}_p^{k-1}\right\}$; $\tilde{\boldsymbol{S}}_j^k$ is the fuzzy solution holon j corresponding to the level $k > 0$ of the holarchy of fuzzy holon solutions $\tilde{\boldsymbol{S}}_j^k = \left\{\tilde{\boldsymbol{S}}_1^{k-1}, \cdots, \tilde{\boldsymbol{S}}_p^{k-1}\right\}$; and $\tilde{M}_j^k = \langle \tilde{\boldsymbol{F}}_j^k, \tilde{\boldsymbol{S}}_j^k, \tilde{\boldsymbol{A}}_j^k \rangle$ is a fuzzy functional-structural module holon with $\tilde{\boldsymbol{A}}_j^k : \tilde{\boldsymbol{F}}_j^k \leftrightarrow \tilde{\boldsymbol{S}}_j^k$.

3.2 Holonic fuzzy agent modeling for holonic modules

Holonic fuzzy agents are modeled to integrate the behavior of both fuzzy agents and holonic agents. The holonic fuzzy agent modeling task for holonic modules involves answering three questions. The first question is "What is the model of a fuzzy agent?" Once a model of a fuzzy agent is proposed, we can ask the second question. How can a model of the fuzzy agent integrate the properties of holonic modules represented by Eqs. (2), (4) and (6)? A model of a holonic fuzzy agent is proposed to answer this question. Finally, the third question is "How can the model of a holonic fuzzy agent be applied for holonic modules?". Models of agentification and holonization are used to answer this question.

To respond to the first question, a model of fuzzy agent is adapted from Ref. [33]. The definition of a fuzzy agent-based system $\tilde{M}_\alpha$ is given as

$$\tilde{M}_\alpha = \left\langle \tilde{A}, \tilde{I}, \tilde{P}, \tilde{O}, \Phi_{\tilde{A}} \right\rangle, \tag{7}$$

where $\tilde{A}, \tilde{I}, \tilde{P}, \tilde{O},$ and $\Phi_{\tilde{A}}$ are a set of fuzzy agents, a set of fuzzy interactions between fuzzy agents, a set of fuzzy roles that fuzzy agents can perform, a set of fuzzy organizations defined for communities of fuzzy agents, and a set of functions of fuzzy agents' generation, respectively.

The definition of a fuzzy agent $\tilde{\alpha}_i$, whose behavior is of feedback loop type <perceive, decide, act> in a fuzzy context, is given as

$$\tilde{\alpha}_i = \left\langle \Phi_{\Pi(\tilde{\alpha}_i)}, \Phi_{\Delta(\tilde{\alpha}_i)}, \Phi_{\Gamma(\tilde{\alpha}_i)}, K_{\tilde{\alpha}_i} \right\rangle, \tag{8}$$

where $\Phi_{\Pi(\tilde{\alpha}_i)}, \Phi_{\Delta(\tilde{\alpha}_i)}$ and $\Phi_{\Gamma(\tilde{\alpha}_i)}$ are function of observation, function of decision, function of action, and knowledge of a fuzzy agent $\tilde{\alpha}_i$, respectively. This knowledge includes decision rules, objects, characteristics of the domain (for instance, the geometric and topological attributes), acquaintances (for instance, a set of fuzzy agents from the same holonic level, as we describe below), and dynamic knowledge (observed events, internal states)

$$\Phi_{\Pi(\tilde{\alpha}_i)} : (E_{\tilde{\alpha}_i} \cup I_{\tilde{\alpha}_i}) \times \Sigma_{\tilde{\alpha}_i} \rightarrow \Pi(\tilde{\alpha}_i), \tag{9}$$

where $E_{\tilde{\alpha}_i}$, $I_{\tilde{\alpha}_i}$, $\Sigma_{\tilde{\alpha}_i}$, $\Pi(\tilde{\alpha}_i)$ are the finite sets of observed events, interactions, states, and perceptions of a fuzzy agent $\tilde{\alpha}_i$, respectively.

$$\Phi_{\Delta(\tilde{\alpha}_i)} : \Pi_{\tilde{\alpha}_i} \times \Sigma_{\tilde{\alpha}_i} \rightarrow \Delta_{\tilde{\alpha}_i}, \tag{10}$$

where $\Pi_{\tilde{\alpha}_i}$, $\Sigma_{\tilde{\alpha}_i}$, $\Delta_{\tilde{\alpha}_i}$ are the finite sets of perceptions, states, and decisions of a fuzzy agent $\tilde{\alpha}_i$, respectively.

$$\Phi_{\Gamma(\tilde{\alpha}_i)} : \Delta_{\tilde{\alpha}_i} \times \Sigma_{\tilde{\alpha}_i} \rightarrow \Gamma_{\tilde{\alpha}_i}, \tag{11}$$

where $\Delta_{\tilde{\alpha}_i}$, $\Sigma_{\tilde{\alpha}_i}$, $\Gamma_{\tilde{\alpha}_i}$ are the finite sets of decisions, states, and actions of a fuzzy agent, respectively.

The definition of a fuzzy interaction $\tilde{\iota}_1 \in \tilde{I}$ between two fuzzy agents is given as

$$\tilde{\iota}_1 = \langle \tilde{\alpha}_\mathrm{s}, \tilde{\alpha}_\mathrm{r}, \tilde{\gamma}_\mathrm{c} \rangle, \tag{12}$$

where $\tilde{\alpha}_\mathrm{s}$ is the fuzzy agent source of the fuzzy interaction; $\tilde{\alpha}_\mathrm{r}$ is the fuzzy agent destination, and $\tilde{\gamma}_\mathrm{c}$ is a fuzzy act of communication ($\tilde{\gamma}_\mathrm{c} \in \tilde{\Gamma}$ and $\tilde{\Gamma}$={inform, diffuse, ask, reply, confirm}).

To respond to the second question, we define a model of a holonic fuzzy agent. A holonic fuzzy agent-based system should satisfy the holonic decomposition of the product into fuzzy structural module holons, fuzzy functional module holons, and fuzzy functional-structural module holons. Thus, a set $\tilde{H}$ of holonic fuzzy agents, forming a holonic fuzzy agent-based system, which respects Eqs. (2), (4), (6) and (7), is defined recursively as

(i) Each fuzzy agent $\tilde{\alpha}_i \in \tilde{A}$, becomes an atomic holonic fuzzy agent $\tilde{h}_i$ of level zero, also noted $\tilde{H}_i^0$, where $\tilde{H}_i^0 = (\{\tilde{\alpha}_i\}, \{\tilde{\alpha}_i\}, \emptyset) \in \tilde{H}$;

(ii) $\tilde{H}_i^k = \left(\tilde{H}_0, \tilde{H}', \tilde{R}\right) \in \tilde{H}$ is the holonic fuzzy agent i of level k, where $\tilde{H}' \subseteq \tilde{H}$ is the set of holonic fuzzy agents that participate in $\tilde{H}_i^k$, $\tilde{H}_0 \subseteq \tilde{H}'$ is the non-empty set of holonic fuzzy agents that represent $\tilde{H}_i^k$ to the environment and is responsible for coordinating the actions inside $\tilde{H}_i^k$. $\tilde{R}$ defines a fuzzy relationship inside $\tilde{H}_i^k$ (i.e., $\tilde{R}$ specifies the holonic fuzzy agent organization). Each $\tilde{H}_i^k$ respects Eq. (8), where $K_{\tilde{h}_i} = K_{\tilde{\alpha}_i} \cup \left(\tilde{H}_0, \tilde{H}', \tilde{R}\right)$.

This new model, integrating fuzzy agent and holonic concept, respects the following holonic properties.

(i) Self-organization, self-adaptation, transcendence, self-efficiency, dynamics, and emergence, which are intrinsic properties of the multiagent paradigm;

(ii) Self-similarity, self-preservation, self-expansion, and robustness, which are intrinsic properties of a holonic agent paradigm;

(iii) Utility, self-well-being, and flexibility, which are intrinsic properties of a fuzzy agent paradigm.

To respond to the third question, the model of a holonic fuzzy agent is applied to the problems represented by Eqs. (2), (4), (6) for intelligent holonic modular formation. In this case, fuzzy functions and fuzzy solutions are fuzzy agentified

$$\text{Agentification} : \tilde{F}, \tilde{S}, \tilde{M} \rightarrow \tilde{A}_F, \tilde{A}_S, \tilde{A}_M, \tag{13}$$

where each fuzzy function of the set $\tilde{F}$ is transformed into a fuzzy agent $\tilde{f}_{[1,2,\cdots,m]} \in \tilde{A}_F$; each fuzzy solution of the set $\tilde{S}$ is transformed into a fuzzy agent $\tilde{s}_{[1,2,\cdots,n]} \in \tilde{A}_S$, and each generated module of the set $\tilde{M}$ is defined as a fuzzy agent $\tilde{m}_{[1,2,\cdots,q]} \in \tilde{A}_M$.

Then, from these three sets of fuzzy agents and considering Eqs. (2), (4), (6), holonic fuzzy agents are generated

$$\text{Holonisation} : \tilde{A}_F, \tilde{A}_S, \tilde{A}_M \rightarrow \tilde{H}_F, \tilde{H}_S, \tilde{H}_M, \tag{14}$$

where $\tilde{F}_j^k \in \tilde{H}_F$, with $\tilde{F}_j^0 \in \tilde{A}_P$ and $\tilde{F}_j^{k>1} = \left\{\tilde{F}_1^{k-1}, \cdots, \tilde{F}_p^{k-1}\right\}$; $\tilde{F}_j^k \in \tilde{H}_F$ with $\tilde{S}_j^0 \in \tilde{A}_S$ and $\tilde{S}_j^{k>1} = \left\{\tilde{S}_1^{k-1}, \cdots, \tilde{S}_p^{k-1}\right\}$; $\tilde{M}_j^0 \in \tilde{A}_M$, $\tilde{M}_j^i = \langle \tilde{F}_j^k, \tilde{S}_j^k, \tilde{R}_j^k \rangle$ with $\tilde{F}_j^k \subset \tilde{H}_F, \tilde{S}_j^k \subset \tilde{H}_S$, and $\tilde{R}_j^k : \tilde{F}_j^k \leftrightarrow \tilde{S}_j^k$.

4 Formation of holonic modules with holonic fuzzy agents

The proposed approach for the formation of holonic modules occurs in three phases (see Fig. 2).

Phase 1: Modeling of fuzzy agents. In this phase, all the communities of fuzzy agents and the intra- and inter-community interactions necessary for the holonic modularity are built. A requirement-function-solution-service network of fuzzy agents emerges. Clearly, this network is a fuzzy one.

Phase 2: Customization of the product-service system. In this phase, the requirement-function-solution-service network is customized to respond to specific customer requirements and specifications of different services in the product lifecycle.

Phase 3: Formation of holonic modules. In this phase, from the attractor agent recognition, the fuzzy function and fuzzy solution agents interact to form the holonic fuzzy module agents.

4.1 Modeling of fuzzy agents

All elements of the model (e.g., functions, solutions, and modules) are fuzzy agentified and have specific knowledge.

Knowledge of each fuzzy function agent $\tilde{f}_i \in \tilde{A}_F$ includes the following.

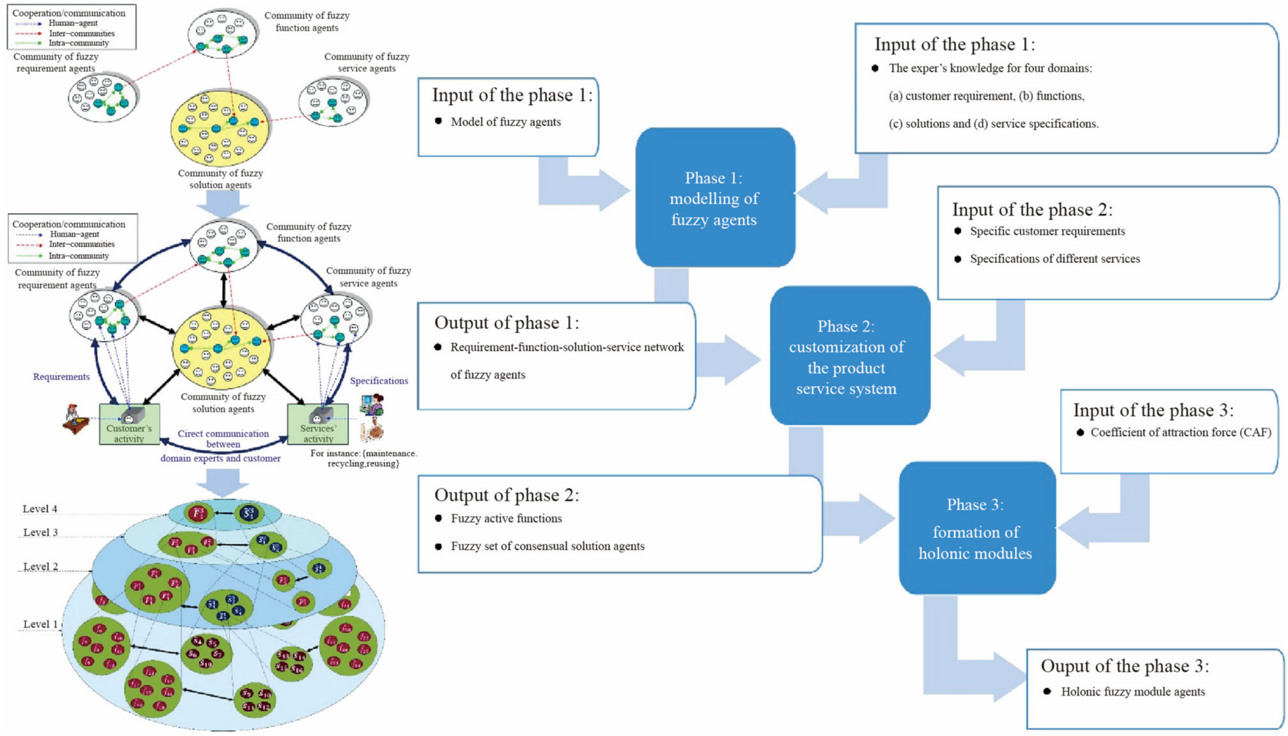


Fig. 2 Workflow for modular design through holonic fuzzy agents

(i) A set of fuzzy solution agents $\tilde{A}'_S \subset \tilde{A}_S$ that have the capability to satisfy the function agent $\tilde{f}_i$;
(ii) A set of fuzzy agents function-specific decision rules $\Delta_{\tilde{f}_i}$;
(iii) Ability to generate a holonic fuzzy function agent $\varphi\left(\tilde{f}_i, \tilde{h}_j \in \tilde{H}_F\right)$.

Knowledge of each fuzzy solution agent $\tilde{s}_i \in \tilde{A}_S$ includes the following.

(i) A set of fuzzy function agents $\tilde{A}'_f \subset \tilde{A}_f$ that the solution agent $\tilde{s}_i$ can satisfy;
(ii) A set of fuzzy agents solution-specific decision rules $\Delta_{\tilde{s}_i}$;
(iii) Ability to generate a holonic fuzzy solution agent: $\varphi\left(\tilde{s}_i, \tilde{h}_j \in \tilde{H}_S\right)$.

Knowledge of each fuzzy module agent $\tilde{m}_i \in \tilde{A}_M$ includes the following.

(i) A set $\tilde{A}'_F \subset \tilde{A}_F$ of fuzzy function agents include the fuzzy module agent $\tilde{m}_i$;
(ii) A set $\tilde{A}'_S \subset \tilde{A}_S$ of fuzzy solution agents included in the fuzzy module agent $\tilde{m}_i$;
(iii) A set $\tilde{R}$ of the relationship between fuzzy function agents and fuzzy solution agents ($\tilde{R} : \tilde{A}'_F \to \tilde{A}'_S$);
(iv) A set of fuzzy agents module-specific decision rules $\Delta_{\tilde{m}_i}$;
(v) Ability to generate a holonic fuzzy module agent: $\varphi\left(\tilde{m}_i, \tilde{h}_j \in \tilde{H}_S\right)$.

The building of fuzzy multiagent systems takes into consideration the following.

(i) Community-building using fuzzy agents. There are four communities of fuzzy agents: requirement, function, solution, and service.
(ii) Intra and inter-community interactions between fuzzy agents: defining interactions between fuzzy function agents and the fuzzy requirement agents, defining interactions among fuzzy function agents, defining interactions between fuzzy solution agents and fuzzy function agents, defining interactions between fuzzy solution agents, defining interactions between fuzzy solution agents and fuzzy constraint agents.

The requirement-function-solution-service network of fuzzy agents emerges from intra- and inter- actions among fuzzy requirement agents, fuzzy function agents, fuzzy solution agents, and fuzzy service agents. As a consequence, this network (which is the output of Phase 1) is a fuzzy one (see Fig. 2).

The expert's knowledge allows for specification of interactions between fuzzy agents in four considered domains: customer requirements, functions, solutions, and service specifications.

4.2 Customization of the product-service system

This phase leads to the emergence of a complete customized network of fuzzy agents (built in Phase 1) to respond to specific customer requirements and specifications of different services in the product lifecycle (see Fig. 2).

Firstly, fuzzy requirement agents use their model to work out specific customer requirements and, after that, the latter is modeled into fuzzy values.

Once the fuzzy set of requirements are available, including the fuzzy relationship between the fuzzy function agents and fuzzy requirement agents, the fuzzy set of function agents, also called fuzzy active functions, appears dynamically. Then, the fuzzy active functions, a subset of fuzzy function agents, interact between them throughout the design process.

The interaction between the fuzzy set of solution agents and fuzzy set of active function agents leads to the emergence of fuzzy active solutions. Soon after the fuzzy set of active function agents are defined, the fuzzy solution agents are activated.

Secondly, the fuzzy service agents use their model to get the specifications of different services. Then, the service preferences are transformed into fuzzy values. When the fuzzy agent solutions interact with the fuzzy service agents, a new fuzzy set of active solution agents will emerge that will satisfy considered services' specifications.

Finally, the fuzzy set of consensual solution agents emerges as a fuzzy set of active solution agents meeting the customer requirements and the specifications of the services. The fuzzy set of active function agents and the fuzzy set of consensual solution agents (which are the outputs of Phase 2) will be used for the formation of holonic modules (in Phase 3) (see Fig. 2).

4.3 Formation of holonic modules

In holon architecture, self-similarity can be observed in "horizontal" and "vertical" spaces. Horizontal self-similarity defines the existence of self-similarity between various specializations at the fixed level aggregation. Vertical self-similarity defines the existence of self-similarity between various levels of aggregation so that there is work similarity between higher and lower levels of holons.

Firstly, our proposal postulates that the intelligent module follows the holon's structure, and this requires the creation of intelligent modules within an intelligent module. Secondly, our proposal postulates that an intelligent module holon is created around defined physical elements such as functions or solutions called attractors.

The attractors are structurally stable elements in the complete customized network of requirement-function-solution-service fuzzy agents. The attractors (fuzzy function agents or fuzzy solution agents) can be found in the fuzzy set of active function agents and in the fuzzy set of consensual solution agents (which are the inputs of Phase 3)(see Fig. 2). Indeed, the fuzzy set of consensual solution agents are considered stable because they emerge as a fuzzy set of active solution agents meeting the customer requirements and the specifications of the services. Similarly, the fuzzy active functions emerge as a fuzzy set of function agents meeting the customer requirements.

Properties relating to holonic fuzzy module formation are shown in Table 1. The fuzzy active function or fuzzy consensual solution can be weighted w_i as per the

Table 1 Properties of fuzzy module formation

Properties	Meaning
Holon	A module is a fuzzy holon
Coupling	A holonic module is defined through two basic coupled elements: a fuzzy holonic function agent and a fuzzy holonic solution agent
Attractor	An attractor is a structurally stable agent, fuzzy function, or fuzzy solution toward which a module tends to evolve dynamically
Force of attraction	An attractor exerts a force of attraction on either of the other fuzzy agents
Condition of attraction	The contribution of a fuzzy agent (fuzzy holonic function or fuzzy holonic solution) to a module must be greater than or equal to an acceptable value of the force of attraction, called Coefficient of Attraction Force (CAF). The CAF varies between 0 and 1
Competition of fuzzy design modules and condition of transfer	The contribution of a fuzzy agent (fuzzy holonic function or fuzzy holonic solution) to its own module must be greater than its contribution to all the other modules
Equivalent modules	The construction of equivalent modules reposes on the possibility that a fuzzy agent (fuzzy holonic function or fuzzy holonic solution) should be able to contribute in an equal way to several modules
Preferences of designer	This property stipulates that a designer can build a preferential relationship in the fuzzy set of function agents and the fuzzy set of solution agents

Table 2 Holonicmodule formation based on holonic function attractors

```
Algorithm                                                                              Meaning

(i) Given a set of $n$ fuzzy function holons $\{\tilde{F}_1^0,\cdots,\tilde{F}_n^0\}\in\tilde{H}_F$;     // initial conditions
    Given a set of $m$ fuzzy solution holons $\{\tilde{S}_1^0,\cdots,\tilde{S}_m^0\}\in\tilde{H}_S$;   // initial conditions
    Given a set of $p$ fuzzy module holons $\{\tilde{M}_1^0,\cdots,\tilde{M}_p^0\}\in\tilde{H}_M$;     // initial conditions

   Each fuzzy function holon agent $\tilde{F}_i^k$, do {
(ii) CAF $\leftarrow$ CAFencours                                      // current CAF = CAFencours, for instance, CAF=1.0
     $nbX_i^k \leftarrow 0$                                             // number of fuzzy solution holons that $\tilde{F}_i^k$ satisfies = 0
     $\_h$ ←false                                                       // $\tilde{F}_i^k$ is not included in a fuzzy function holon of higher level
(iii) While ($\_h$ == false) {                                          // while $\tilde{F}_i^k$is not included in a fuzzy holon of higher level
         For $j$ in $1,\cdots,$ Card$(S^k)$                              // loop to know how much fuzzy solution holons satisfy $\tilde{F}_i^k$
             $nbX_i^k \leftarrow nbX_i^k +$ Exchange_1 $(\tilde{F}_i^k,\tilde{S}_j^k)$   // Exchange to determine $nbX_i^k$
(iv)     IF $\left(\max\left(\text{Card}(\tilde{X}_i^k), nbX_i^k\right)\right)$ then   // $\tilde{F}_i^k$ is satisfied by the greatest number of fuzzy solution holons
             $\rho(\tilde{F}_i^k$ , attractor)                          // $\tilde{F}_i^k$ is an attractor of fuzzy function holons; $\tilde{F}_i^k$ has the role of attractor
(v)          generate $(\tilde{F}_i^{k+1})$                             // generation of a new fuzzy function holon $\tilde{F}_i^{k+1} \leftarrow \tilde{F}_i^k$
             generate $(\tilde{S}_i^{k+1})$                             // generation of a new fuzzy solution holon $\tilde{S}_i^{k+1} \leftarrow \tilde{S}_i^k$
(vi)         calculate (CAF$_i$,$\tilde{F}_i^k$,$\tilde{F}_i^{k+1}$)    // calculation of CAF$_i$ $\leftarrow$ Card$(\tilde{F}_i^{k+1})$/Card$(\tilde{F}_i^k)$
             For $i$ in $1,\cdots,$ Card$(\tilde{S}_i^k)$               // loop to get the list of fuzzy function holons satisfied by the
                 $\tilde{F}_m$ ←Exchange_2 $(\tilde{F}_i^k,\tilde{S}_i^k)$   // fuzzy solution holons that satisfy $\tilde{F}_i^k$
(vii)        For $l$ in $1,\cdots,$ Card$(F_m)$                         // loop for each fuzzy function holon satisfied by fuzzy solution holons of $\tilde{S}_i^x$
                 CAF$_i$.$[\tilde{F}_l^k]$ $\leftarrow$ Exchange_3$(\tilde{F}_i^k,\tilde{F}_l^k,\tilde{S}_i^{k+1})$   // exchange to get the CAF of $\tilde{F}_l^k$
                 IF (CAF$_i$. $[\tilde{F}_l^k] \geq$ CAFencours) then   // $\tilde{F}_i^k$ is a stable attractor for the fuzzy function holon $\tilde{F}_l^k$
                     $\tilde{F}_i^{k+1} \leftarrow \tilde{F}_i^{k+1} \cup \tilde{F}_l^k$   // $\tilde{F}_l^k$ is included in the new fuzzy function holon $\tilde{F}_i^{k+1}$
                     inform $(\tilde{F}_i^k,\tilde{F}_l^k,\tilde{F}_i^{k+1})$   // $\tilde{F}_l^k$ is informed of inclusion in the fuzzy function holon $\tilde{F}_i^{k+1}$
                 Else inform $(\tilde{F}_i^k,\tilde{F}_l^k,!\,\tilde{F}_i^{k+1})$   // $\tilde{F}_l^k$ is informed of not inclusion in the fuzzy function holon $\tilde{F}_i^{k+1}$
(viii) generate $(\tilde{M}_i^{k+1},\tilde{F}_i^k,\tilde{F}_i^{k+1},\tilde{S}_i^{k+1})$   // generation of the new fuzzy module holon $\tilde{M}_i^{k+1}$
         Else                                                           // $\tilde{F}_i^k$ is not satisfied by the greatest number of fuzzy solution holons
             wait (inform $(\tilde{F}_j^k,\tilde{F}_i^k,a)$)            // $\tilde{F}_i^k$ is waiting for information on possible attraction
             IF $(a == \tilde{F}_i^{k+1})$ then                         // $\tilde{F}_i^k$ is attracted by the fuzzy function holon $\tilde{F}_y^k$
                 $\_h$ ←true                                            // $\tilde{F}_i^k$ is included in a fuzzy function holon of higher level
(ix)             $\rho(\tilde{F}_j^k$ , attractor)                      // attractor of $\tilde{F}_i^k$ is the fuzzy function holon $\tilde{F}_j^k$; $\tilde{F}_j^k$ has the role of attractor
           Else $\_h \leftarrow$ false                                  // $\tilde{F}_i^k$ is never included in a fuzzy holon of higher level
(x) lower (CAF)                                                         // lower the current CAF (CAF = CAF−$\varepsilon$, $\varepsilon$=[0,CAF]
(xi) } End while
        } End do
```

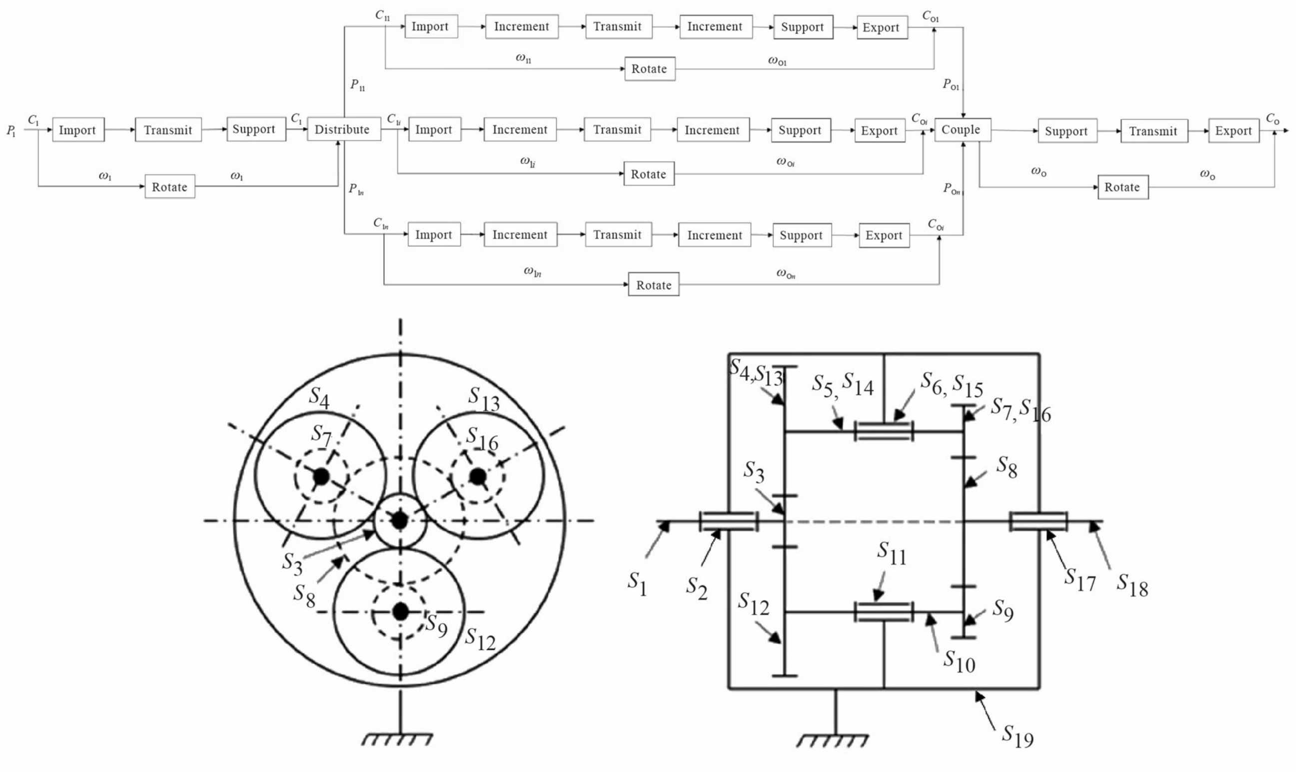


Fig. 3 Functional architecture and conceptual solution (n=3)

designer's preference. As such, engineers know the acceptable value of variables from various sources, including their own experience. The acceptable intervals are those where the designer has a positive preference. While specifying the designer's preference for the fuzzy set agents, the α-levels (acceptable levels), which are equal to or more than the designer's preferred threshold value, can be used. For the formation of holonic modules, the contribution of a fuzzy agent (fuzzy function or fuzzy solution) to a fuzzy module must be greater than or equal to an acceptable value of the force of attraction called the coefficient of attraction force (CAF) (see Tables 1 and 2).

The distributed algorithm of holonic module formation executed by each fuzzy holonic function agent $\tilde{F}_i^x$ is presented in Table 2. The main steps of this algorithm are identified by indices as follows.

(i) Initial conditions;
(ii) Coefficient of attraction force;
(iii) Vertical "self-similarity";
(iv) Fuzzy attractor identification;
(v) Attraction of fuzzy solution holons;
(vi) Attraction of fuzzy function holons;
(vii) Equivalent fuzzy module holons and transfer;
(viii) New fuzzy module holons;
(ix) Fuzzy modules within a fuzzy module; and
(x) Evaluation of the end of fuzzy module holons formation.

5 Application

The gearbox is a mechanism which increases the engine torque (I = Input) in order to rotate a receiving member under the effect of a new torque (O = Output) such that

$$C_O = \eta_{IO} \frac{I}{k_{IO}} C_I, \tag{15}$$

$$\text{where } k_{IO} = \frac{\omega_I}{\omega_O}, \tag{16}$$

η_{IO} the mechanical efficiency of the gearbox.

Most gearboxes are reversible; therefore, in this case, the input function can be assigned to shaft O, and the output is assigned to shaft I. Then, the mechanism is called a velocity multiplier. The output torque is $C_I > C_O$.

The reduction of the overall bulk is an important requirement of the gearbox. By splitting the input power P_1 into n paths, this requirement is satisfied. It can be written as

Fuzzy function		Fuzzy solution																		
		Shaft $\tilde{s}_1$	Pivot $\tilde{s}_2$	Gears $\tilde{s}_3$	Gears $\tilde{s}_4$	Shaft $\tilde{s}_5$	Pivot $\tilde{s}_6$	Gears $\tilde{s}_7$	Gears $\tilde{s}_8$	Gears $\tilde{s}_9$	Shaft $\tilde{s}_{10}$	Pivot $\tilde{s}_{11}$	Gears $\tilde{s}_{12}$	Gears $\tilde{s}_{13}$	Shaft $\tilde{s}_{14}$	Pivot $\tilde{s}_{15}$	Gears $\tilde{s}_{16}$	Pivot $\tilde{s}_{17}$	Shaft $\tilde{s}_{18}$	Bati $\tilde{s}_{19}$
Import C_{I}	$\tilde{f}_1$	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Transmit C_{I}	$\tilde{f}_2$	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Support ω_{I}	$\tilde{f}_3$	0.9	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8
Rotate ω_{I}	$\tilde{f}_4$	0.9	0.9	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8
Distribute P_{I}	$\tilde{f}_5$	0.4	0	0.9	0.9	0	0	0	0	0	0	0	0.9	0.9	0	0	0	0	0	0
Import C_{I1}	$\tilde{f}_6$	0	0	0	0.9	0.8	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Increment $C_{\mathrm{I1}} \times k$	$\tilde{f}_7$	0	0	0	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
Transmit $C_{\mathrm{I1}} \times k$	$\tilde{f}_8$	0	0	0	0.9	0.9	0	0.8	0	0	0	0	0	0	0	0	0	0	0	0
Increment $C_{\mathrm{I1}} \times k \times k$	$\tilde{f}_9$	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0	0	0	0	0	0
Support $C_{\mathrm{I1}} - C_{\mathrm{O1}}$	$\tilde{f}_{10}$	0	0	0	0.8	0.8	0.8	0.8	0	0	0	0	0	0	0	0	0	0	0	0.8
Export C_{O1}	$\tilde{f}_{11}$	0	0	0	0	0.8	0	0.9	0	0	0	0	0	0	0	0	0	0	0	0
Rotate ω_{O1}	$\tilde{f}_{12}$	0	0	0	0.8	0.9	0.9	0.8	0	0	0	0	0	0	0	0	0	0	0	0.8
Import C_{I2}	$\tilde{f}_{13}$	0	0	0	0	0	0	0	0	0	0.8	0	0.9	0	0	0	0	0	0	0
Increment $C_{\mathrm{I2}} \times k$	$\tilde{f}_{14}$	0	0	0	0	0	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0
Transmit $C_{\mathrm{I2}} \times k$	$\tilde{f}_{15}$	0	0	0	0	0	0	0	0	0.8	0.9	0	0.9	0	0	0	0	0	0	0
Increment $C_{\mathrm{I2}} \times k \times k$	$\tilde{f}_{16}$	0	0	0	0	0	0	0	0	0.9	0	0	0	0	0	0	0	0	0	0
Support $C_{\mathrm{I2}} - C_{\mathrm{O2}}$	$\tilde{f}_{17}$	0	0	0	0	0	0	0	0	0.8	0.8	0.8	0.8	0	0	0	0	0	0	0.8
Export C_{O2}	$\tilde{f}_{18}$	0	0	0	0	0	0	0	0	0.9	0.8	0	0	0	0	0	0	0	0	0
Rotate ω_{O2}	$\tilde{f}_{19}$	0	0	0	0	0	0	0	0	0.8	0.9	0.9	0.8	0	0	0	0	0	0	0.8
Import C_{I3}	$\tilde{f}_{20}$	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0.8	0	0	0	0	0
Increment $C_{\mathrm{I3}} \times k$	$\tilde{f}_{21}$	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0	0	0	0	0	0
Transmit $C_{\mathrm{I3}} \times k$	$\tilde{f}_{22}$	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0.9	0	0.8	0	0	0
Increment $C_{\mathrm{I3}} \times k \times k$	$\tilde{f}_{23}$	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0.8	0.9	0	0	0
Support $C_{\mathrm{I3}} - C_{\mathrm{O3}}$	$\tilde{f}_{24}$	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0	0.8	0	0	0.8
Export C_{O3}	$\tilde{f}_{25}$	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0	0.9	0	0	0
Rotate ω_{O3}	$\tilde{f}_{26}$	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0.9	0.9	0.8	0	0	0.8
Couple $C_{\mathrm{O}i}$	$\tilde{f}_{27}$	0	0	0	0	0	0	0.9	0.9	0.9	0	0	0	0	0	0	0.9	0	0.4	0
Support C_{O}	$\tilde{f}_{28}$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.8	0.9	0.8
Transmit C_{O}	$\tilde{f}_{29}$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0
Export C_{O}	$\tilde{f}_{30}$	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.9	0
Rotate ω_{O}	$\tilde{f}_{31}$	0	0	0	0	0	0	0	0.8	0	0	0	0	0	0	0	0	0.9	0.9	0.8

Fig. 4 Fuzzy relationship between fuzzy function agents and fuzzy solution agents

$$P_{\mathrm{O}} = \eta_1 \frac{P_{\mathrm{I}}}{n} + \cdots + \eta_i \frac{P_{\mathrm{I}}}{n} + \cdots + \eta_n \frac{P_{\mathrm{I}}}{n}, \tag{17}$$

where P_{I} is the input power of the gearbox, P_{O} output power of the gearbox, η_i the mechanical efficiency of each division of the gearbox.

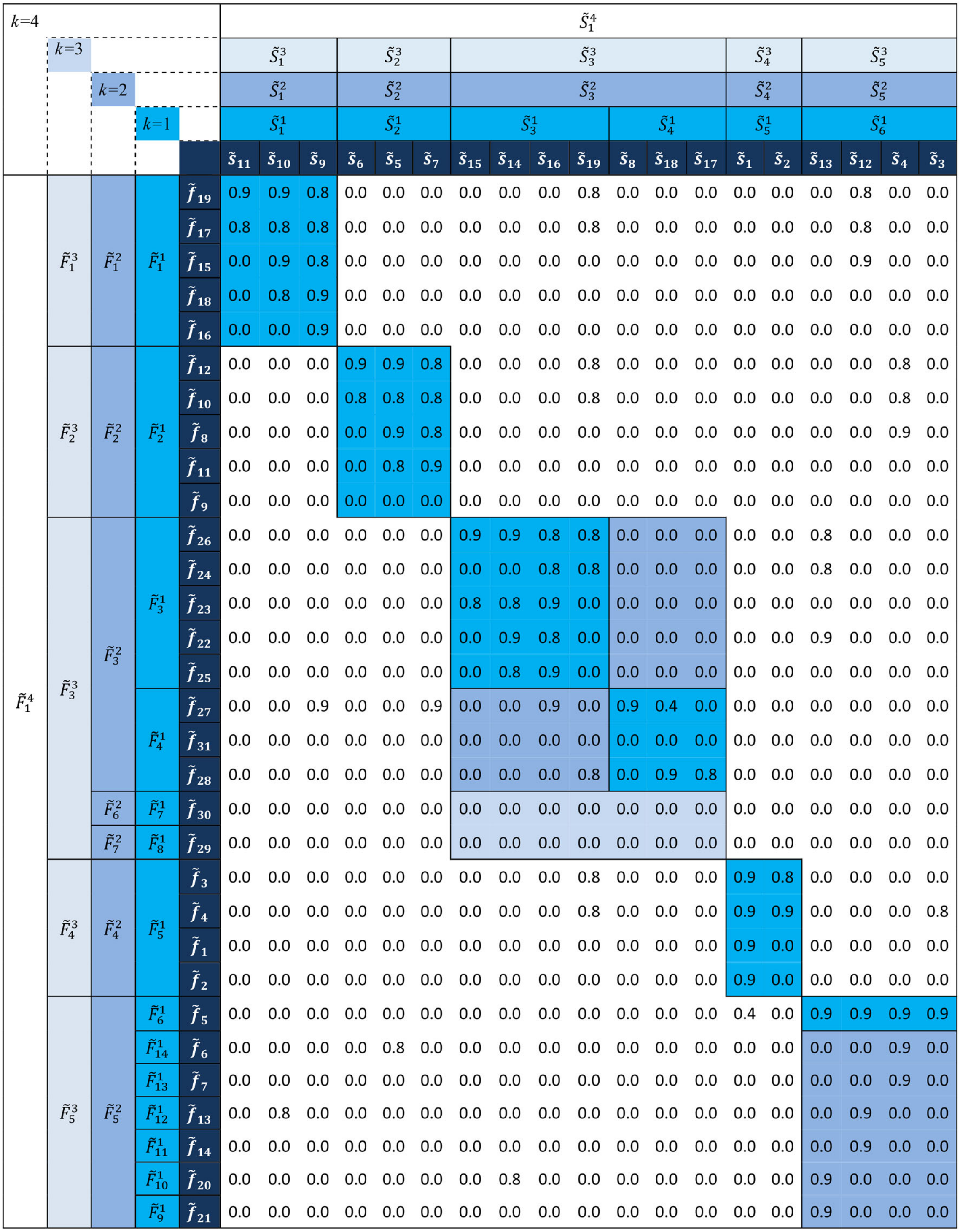
$$P_{\mathrm{O}} = \eta_{\mathrm{IO}} P_{\mathrm{I}}, \tag{18}$$

$$\text{with } \eta_{\mathrm{IO}} = \frac{1}{n}(\eta_1 + \cdots + \eta_i + \cdots + \eta_n). \tag{19}$$

The functional architecture of a parallel gearbox, with the input of power *I* (Input) and the output of power *O* (Output), is represented in Fig. 3. It shows a parallel gearbox responding to the functional architecture for n=3. The input is represented by shaft S_1, and the output is represented by shaft S_{18}. The gears S_3, S_4, S_7, S_8, S_9, S_{12}, S_{13}, and S_{16} are used to transmit power from the input shaft S_1 to the output shaft S_{18}. The gears are assembled on shafts S_1, S_5, S_{10}, S_{14} and S_{18} while the revolute joints S_2, S_6, S_{11}, S_{15} and S_{17} between shafts and frame S_{19} guide the rotation of the shafts.

The set of fuzzy active function agents and their corresponding fuzzy consensual solution agents are established from the customization of the PSS in the subnetwork of the fuzzy function agent-fuzzy solution agent in the platform FAPIC [59]. Figure 4 shows the fuzzy relationship between these fuzzy function and fuzzy solution agents.

Thereafter, from the attractor agent recognition, these fuzzy function and fuzzy solution agents interact to form the holonic fuzzy module agents (see Fig. 5). The different steps of holonic fuzzy module agents are shown in Table 3.



k=4					$\tilde{S}_1^4$																		
	k=3				$\tilde{S}_1^3$			$\tilde{S}_2^3$			$\tilde{S}_3^3$							$\tilde{S}_4^3$		$\tilde{S}_5^3$			
		k=2			$\tilde{S}_1^2$			$\tilde{S}_2^2$			$\tilde{S}_3^2$							$\tilde{S}_4^2$		$\tilde{S}_5^2$			
			k=1		$\tilde{S}_1^1$			$\tilde{S}_2^1$			$\tilde{S}_3^1$				$\tilde{S}_4^1$			$\tilde{S}_5^1$		$\tilde{S}_6^1$			
					$\tilde{s}_{11}$	$\tilde{s}_{10}$	$\tilde{s}_9$	$\tilde{s}_6$	$\tilde{s}_5$	$\tilde{s}_7$	$\tilde{s}_{15}$	$\tilde{s}_{14}$	$\tilde{s}_{16}$	$\tilde{s}_{19}$	$\tilde{s}_8$	$\tilde{s}_{18}$	$\tilde{s}_{17}$	$\tilde{s}_1$	$\tilde{s}_2$	$\tilde{s}_{13}$	$\tilde{s}_{12}$	$\tilde{s}_4$	$\tilde{s}_3$
$\tilde{F}_1^4$	$\tilde{F}_1^3$	$\tilde{F}_1^2$	$\tilde{F}_1^1$	$\tilde{f}_{19}$	0.9	0.9	0.8	0.0	0.0	0.0	0.0	0.0	0.0	0.8	0.0	0.0	0.0	0.0	0.0	0.0	0.8	0.0	0.0
				$\tilde{f}_{17}$	0.8	0.8	0.8	0.0	0.0	0.0	0.0	0.0	0.0	0.8	0.0	0.0	0.0	0.0	0.0	0.0	0.8	0.0	0.0
				$\tilde{f}_{15}$	0.0	0.9	0.8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.9	0.0	0.0
				$\tilde{f}_{18}$	0.0	0.8	0.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
				$\tilde{f}_{16}$	0.0	0.0	0.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	$\tilde{F}_2^3$	$\tilde{F}_2^2$	$\tilde{F}_2^1$	$\tilde{f}_{12}$	0.0	0.0	0.0	0.9	0.9	0.8	0.0	0.0	0.0	0.8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.8	0.0
				$\tilde{f}_{10}$	0.0	0.0	0.0	0.8	0.8	0.8	0.0	0.0	0.0	0.8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.8	0.0
				$\tilde{f}_8$	0.0	0.0	0.0	0.0	0.9	0.8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.9	0.0
				$\tilde{f}_{11}$	0.0	0.0	0.0	0.0	0.8	0.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
				$\tilde{f}_9$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	$\tilde{F}_3^3$	$\tilde{F}_3^2$	$\tilde{F}_3^1$	$\tilde{f}_{26}$	0.0	0.0	0.0	0.0	0.0	0.0	0.9	0.9	0.8	0.8	0.0	0.0	0.0	0.0	0.0	0.8	0.0	0.0	0.0
				$\tilde{f}_{24}$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.8	0.8	0.0	0.0	0.0	0.0	0.0	0.8	0.0	0.0	0.0
				$\tilde{f}_{23}$	0.0	0.0	0.0	0.0	0.0	0.0	0.8	0.8	0.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
				$\tilde{f}_{22}$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.9	0.8	0.0	0.0	0.0	0.0	0.0	0.0	0.9	0.0	0.0	0.0
				$\tilde{f}_{25}$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.8	0.9	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
			$\tilde{F}_4^1$	$\tilde{f}_{27}$	0.0	0.0	0.9	0.0	0.0	0.9	0.0	0.0	0.9	0.0	0.9	0.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0
				$\tilde{f}_{31}$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
				$\tilde{f}_{28}$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.8	0.0	0.9	0.8	0.0	0.0	0.0	0.0	0.0	0.0
		$\tilde{F}_6^2$	$\tilde{F}_7^1$	$\tilde{f}_{30}$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
		$\tilde{F}_7^2$	$\tilde{F}_8^1$	$\tilde{f}_{29}$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	$\tilde{F}_4^3$	$\tilde{F}_4^2$	$\tilde{F}_5^1$	$\tilde{f}_3$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.8	0.0	0.0	0.0	0.9	0.8	0.0	0.0	0.0	0.0
				$\tilde{f}_4$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.8	0.0	0.0	0.0	0.9	0.9	0.0	0.0	0.0	0.8
				$\tilde{f}_1$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.9	0.0	0.0	0.0	0.0	0.0
				$\tilde{f}_2$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.9	0.0	0.0	0.0	0.0	0.0
	$\tilde{F}_5^3$	$\tilde{F}_5^2$	$\tilde{F}_6^1$	$\tilde{f}_5$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.4	0.0	0.9	0.9	0.9	0.9
			$\tilde{F}_{14}^1$	$\tilde{f}_6$	0.0	0.0	0.0	0.0	0.8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.9	0.0
			$\tilde{F}_{13}^1$	$\tilde{f}_7$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.9	0.0
			$\tilde{F}_{12}^1$	$\tilde{f}_{13}$	0.0	0.8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.9	0.0	0.0
			$\tilde{F}_{11}^1$	$\tilde{f}_{14}$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.9	0.0	0.0
			$\tilde{F}_{10}^1$	$\tilde{f}_{20}$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.9	0.0	0.0	0.0
			$\tilde{F}_9^1$	$\tilde{f}_{21}$	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.9	0.0	0.0	0.0

Fig. 5 Fuzzy module holon agents where *k* represents the level, according to the holon model (see Sect. 3.2)

Three levels of holonic fuzzy module agents emerge during the formation of fuzzy holonic modules. For instance, in the first level, five complete holonic fuzzy module agents $\tilde{M}_1^1, \tilde{M}_2^1, \tilde{M}_3^1, \tilde{M}_4^1$ and $\tilde{M}_5^1$ are recognized. The first fuzzy holonic module agent $\tilde{M}_1^1$ is composed of the holonic function agent $\tilde{F}_1^1$ and the fuzzy holonic agent solution $\tilde{S}_1^1$, which are related by the fuzzy sub-relationship $\tilde{R}_1^1$. The emerged fuzzy function holon agents are shown in Fig. 6. Similarly, the emerged fuzzy solution holon agents are shown in Fig. 7.

Table 3 Different steps of module holon agent formation

Step	Holon agent formation
Fuzzy function and solution agents	$\mathrm{card}(\tilde{F}) = 31;\ \tilde{F} = \{\tilde{f}_1, \tilde{f}_2, \cdots, \tilde{f}_i, \cdots, \tilde{f}_{31}\}$
	$\mathrm{card}(\tilde{S}) = 19;\ \tilde{S} = \{\tilde{s}_1, \tilde{s}_2, \cdots, \tilde{s}_j, \cdots, \tilde{s}_{19}\}$
Fuzzy holonic modules of level 1	$\tilde{M}_1^1 = \{\tilde{F}_1^1 = (\tilde{f}_{19}, \{\tilde{f}_{19}, \tilde{f}_{17}, \tilde{f}_{15}, \tilde{f}_{18}, \tilde{f}_{16}\}),\ \tilde{S}_1^1 = (\tilde{s}_9, \{\tilde{s}_{11}, \tilde{s}_{10}, \tilde{s}_9\}), \tilde{R}_1^1\}$
	$\tilde{M}_2^1 = \{\tilde{F}_2^1 = (\tilde{f}_{12}, \{\tilde{f}_{12}, \tilde{f}_{10}, \tilde{f}_8, \tilde{f}_{11}, \tilde{f}_9\}),\ \tilde{S}_2^1 = (\tilde{s}_7, \{\tilde{s}_6, \tilde{s}_5, \tilde{s}_7\}), \tilde{R}_2^1\}$
	$\tilde{M}_3^1 = \{\tilde{F}_3^1 = (\tilde{f}_{26}, \{\tilde{f}_{26}, \tilde{f}_{24}, \tilde{f}_{23}, \tilde{f}_{22}, \tilde{f}_{25}\}),\ \tilde{S}_3^1 = (\tilde{s}_{16}, \{\tilde{s}_{15}, \tilde{s}_{14}, \tilde{s}_{16}, \tilde{s}_{19}\}), \tilde{R}_3^1\}$
	$\tilde{M}_4^1 = \{\tilde{F}_4^1 = (\tilde{f}_{31}, \{\tilde{f}_{27}, \tilde{f}_{31}, \tilde{f}_{28}\}),\ \tilde{S}_4^1 = (\tilde{s}_{18}, \{\tilde{s}_8, \tilde{s}_{18}, \tilde{s}_{17}\}), \tilde{R}_4^1\}$
	$\tilde{M}_5^1 = \{\tilde{F}_5^1 = (\tilde{f}_4, \{\tilde{f}_3, \tilde{f}_4, \tilde{f}_1, \tilde{f}_2\}),\ \tilde{S}_5^1 = (\tilde{s}_1, \{\tilde{s}_1, \tilde{s}_2\}), \tilde{R}_5^1\}$
	$\tilde{M}_6^1 = \{\tilde{F}_6^1 = (\tilde{f}_5, \{\tilde{f}_5\}),\ \tilde{S}_6^1 = (\tilde{s}_{13}, \{\tilde{s}_{13}, \tilde{s}_{12}, \tilde{s}_4, \tilde{s}_3\}), \tilde{R}_6^1\}$
	$\tilde{M}_7^1 = \{\tilde{F}_7^1 = (\tilde{f}_{30}, \{\tilde{f}_{30}\}),\ \tilde{S}_7^1 = (\emptyset, \{\emptyset\}), \tilde{R}_7^1\}$
	$\tilde{M}_8^1 = \{\tilde{F}_8^1 = (\tilde{f}_{29}, \{\tilde{f}_{29}\}),\ \tilde{S}_8^1 = (\emptyset, \{\emptyset\}), \tilde{R}_8^1\}$
	$\tilde{M}_9^1 = \{\tilde{F}_9^1 = (\tilde{f}_{21}, \{\tilde{f}_{21}\}),\ \tilde{S}_9^1 = (\emptyset, \{\emptyset\}), \tilde{R}_9^1\}$
	$\tilde{M}_{10}^1 = \{\tilde{F}_{10}^1 = (\tilde{f}_{20}, \{\tilde{f}_{20}\}),\ \tilde{S}_{10}^1 = (\emptyset, \{\emptyset\}), \tilde{R}_{10}^1\}$
	$\tilde{M}_{11}^1 = \{\tilde{F}_{11}^1 = (\tilde{f}_{14}, \{\tilde{f}_{14}\}),\ \tilde{S}_{11}^1 = (\emptyset, \{\emptyset\}), \tilde{R}_{11}^1\}$
	$\tilde{M}_{12}^1 = \{\tilde{F}_{12}^1 = (\tilde{f}_{13}, \{\tilde{f}_{13}\}),\ \tilde{S}_{12}^1 = (\emptyset, \{\emptyset\}), \tilde{R}_{12}^1\}$
	$\tilde{M}_{13}^1 = \{\tilde{F}_{13}^1 = (\tilde{f}_7, \{\tilde{f}_7\}),\ \tilde{S}_{13}^1 = (\emptyset, \{\emptyset\}), \tilde{R}_{13}^1\}$
	$\tilde{M}_{14}^1 = \{\tilde{F}_{14}^1 = (\tilde{f}_6, \{\tilde{f}_6\}),\ \tilde{S}_{14}^1 = (\emptyset, \{\emptyset\}), \tilde{R}_{14}^1\}$
Fuzzy holonic modules of level 2	$\tilde{M}_1^2 = \{\tilde{F}_1^2 = (\tilde{F}_1^1, \{\tilde{F}_1^1\}),\ \tilde{S}_1^2 = (\tilde{S}_1^1, \{\tilde{S}_1^1\}), \tilde{R}_1^2\}$
	$\tilde{M}_2^2 = \{\tilde{F}_2^2 = (\tilde{F}_2^1, \{\tilde{F}_2^1\}),\ \tilde{S}_2^2 = (\tilde{S}_2^1, \{\tilde{S}_2^1\}), \tilde{R}_2^2\}$
	$\tilde{M}_3^2 = \{\tilde{F}_3^2 = (\tilde{F}_3^1, \{\tilde{F}_3^1, \tilde{F}_4^1\}),\ \tilde{S}_3^2 = (\tilde{S}_3^1, \{\tilde{S}_3^1, \tilde{S}_4^1\}), \tilde{R}_3^2\}$
	$\tilde{M}_4^2 = \{\tilde{F}_4^2 = (\tilde{F}_5^1, \{\tilde{F}_5^1\}),\ \tilde{S}_4^2 = (\tilde{S}_5^1, \{\tilde{S}_5^1\}), \tilde{R}_4^2\}$
	$\tilde{M}_5^2 = \{\tilde{F}_5^2 = (\tilde{F}_6^1, \{\tilde{F}_6^1, \tilde{F}_{14}^1, \tilde{F}_{13}^1, \tilde{F}_{12}^1, \tilde{F}_{11}^1, \tilde{F}_{10}^1, \tilde{F}_9^1\}),\ \tilde{S}_5^2 = (\tilde{S}_6^1, \{\tilde{S}_6^1\}), \tilde{R}_5^2\}$
	$\tilde{M}_6^2 = \{\tilde{F}_6^2 = (\tilde{F}_7^1, \{\tilde{F}_7^1\}),\ \tilde{S}_6^2 = (\emptyset, \{\emptyset\}), \tilde{R}_6^2\}$
	$\tilde{M}_7^2 = \{\tilde{F}_7^2 = (\tilde{F}_8^1, \{\tilde{F}_8^1\}),\ \tilde{S}_7^2 = (\emptyset, \{\emptyset\}), \tilde{R}_7^2\}$
Fuzzy holonic modules of level 3	$\tilde{M}_1^3 = \{\tilde{F}_1^3 = (\tilde{F}_1^2, \{\tilde{F}_1^2\}),\ \tilde{S}_1^3 = (\tilde{S}_1^2, \{\tilde{S}_1^2\}), \tilde{R}_1^3\}$
	$\tilde{M}_2^3 = \{\tilde{F}_2^3 = (\tilde{F}_2^2, \{\tilde{F}_2^2\}),\ \tilde{S}_2^3 = (\tilde{S}_2^2, \{\tilde{S}_2^2\}), \tilde{R}_2^3\}$
	$\tilde{M}_3^3 = \{\tilde{F}_3^3 = (\tilde{F}_3^2, \{\tilde{F}_3^2, \tilde{F}_6^2, \tilde{F}_7^2\}),\ \tilde{S}_3^3 = (\tilde{S}_3^2, \{\tilde{S}_3^2\}), \tilde{R}_3^3\}$
	$\tilde{M}_4^3 = \{\tilde{F}_4^3 = (\tilde{F}_4^2, \{\tilde{F}_4^2\}),\ \tilde{S}_4^3 = (\tilde{S}_4^2, \{\tilde{S}_4^2\}), \tilde{R}_4^3\}$
	$\tilde{M}_5^3 = \{\tilde{F}_5^3 = (\tilde{F}_5^2, \{\tilde{F}_5^2\}),\ \tilde{S}_5^3 = (\tilde{S}_5^2, \{\tilde{S}_5^2\}), \tilde{R}_5^3\}$
Fuzzy holonic modules of level 4	$\tilde{M}_1^4 = \{\tilde{F}_1^4 = (\tilde{F}_3^3, \{\tilde{F}_1^3, \tilde{F}_2^3, \tilde{F}_3^3, \tilde{F}_4^3, \tilde{F}_5^3\}),\ \tilde{S}_1^4 = (\tilde{S}_3^3, \{\tilde{S}_1^3, \tilde{S}_2^3, \tilde{S}_3^3, \tilde{S}_4^3, \tilde{S}_5^3\}), \tilde{R}_1^4\}$
Fuzzy holonic modules of level 5	$\tilde{M}_1^5 = \{\tilde{F}_1^5 = (\tilde{F}_1^4, \{\tilde{F}_1^4\}),\ \tilde{S}_1^5 = (\tilde{S}_1^4, \{\tilde{S}_1^4\}), \tilde{R}_1^5\}$

The holarchy of a fuzzy holonic module network with intra- and inter- actions and final self-organization of fuzzy function holon agents, fuzzy solution holon agents, and fuzzy module holon agents is shown in Fig. 8. In this figure, the colored module indicates the progressive interactions of functions and solutions in holonic structures. The holonic structure depends strongly on the competitive attractors and their force of attraction. Thus, the dynamic of an intelligent module holon depends on the dynamics of conflict between agent attractors (see Appendix).

6 Discussion

The proposed model for intelligent modular design uses concepts such as a holon, an attractor, and uncertainty. The formation of multi-scale modules is facilitated by the holon and attractor. As shown in Fig. 8, this leads to the formation of a holarchy of modules, a nested hierarchical order, where a holarchy is a hierarchy of fuzzy module holons. Unlike traditional module formation, which is characterized principally by the decomposition of the product structure, this multi-scale formation is adaptable. The fuzzy

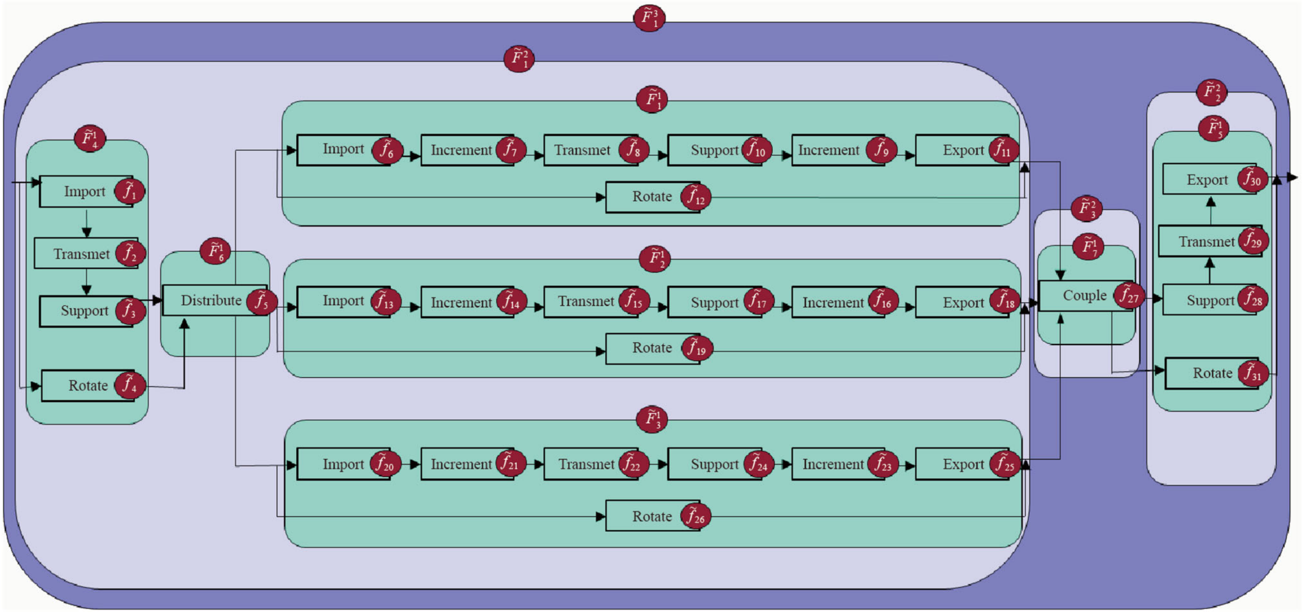


Fig. 6 Fuzzy function holon agents

module holon agents dynamically adjust their structure according to the customization of the PSSs in the sub-network of the fuzzy function agent-fuzzy solution agent and the strength of the attractor. If it is capable of building a fuzzy module holon, a fuzzy module holon agent is considered a strong attractor (see Appendix).

Formation of the communication network by the holonic fuzzy agents removes the distinction between continuous-discontinuous formations of traditional modules. The developed network is the intelligent module within an intelligent module. Indeed, as shown in Fig. 8, this holonic structure's fundamental property, self-embedding, is followed by communication. Thus, with the use of a holarchic model, the module formation problem is modernized. The embedding of a fuzzy module holon is influenced by the whole fuzzy module holons and also influences it. The fuzzy module holon, though complete in itself, is part of a whole fuzzy module holons and consists of many decomposed fuzzy module holons. Horizontal self-similarity defines the existence of self-similarity between various specializations at the fixed level aggregation (see Fig. 8). Vertical self-similarity defines the existence of self-similarity between various levels of aggregation so that there is work similarity between higher and lower levels of holons (see Fig. 8). Thus, from horizontal and vertical communication, holonic fuzzy module agents can reconfigure the systems, either partially or totally.

The suggested model of module holon formation is influenced by knowledge of product design and PSS. However, the module formation is not a straightforward problem as it is riddled with unpredictable changes and sudden transformations. The dynamic of an intelligent module holon depends on the dynamics of conflict between agent attractors. In our model, this is resolved by considering their strength: the capability of agent attractors to form attractors.

However, our approach also suggests that modules, as well as their models, should be designed holistically as adaptive objects. The agent-based digital twin for the intelligent modular design for smart product-service systems should be used to investigate this issue.

7 Conclusions and outlook

The current trend in modular design technologies is to use a combination of technologies, including product configurators, advanced CAD systems, PDM systems, and agent-based systems. Thus, intelligent models, intelligent tools, and intelligent modular products requiring design processes for product configuration are organized intelligently. Intelligent model-tool-product system development, therefore, demands the use of concurrent and holistic engineering approaches. Employing these approaches, the design of intelligent self-sustainable models and products is feasible.

This study is focused on modular design using the concepts of intelligent holons and the integration of fuzziness with the modeling of holonic fuzzy agents. It proposes a formal model for holonic modules. Based on the equations for holonic modules, the holonic fuzzy agents for holonic modules are modeled. After customization of the

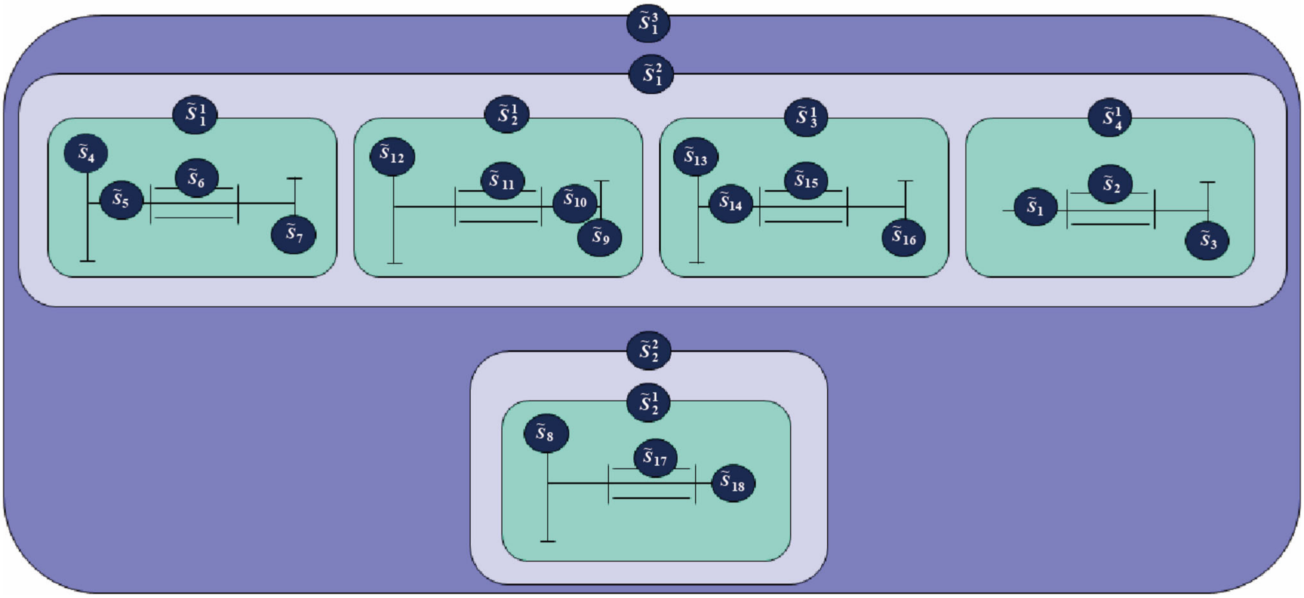


Fig. 7 Fuzzy solution holon agents

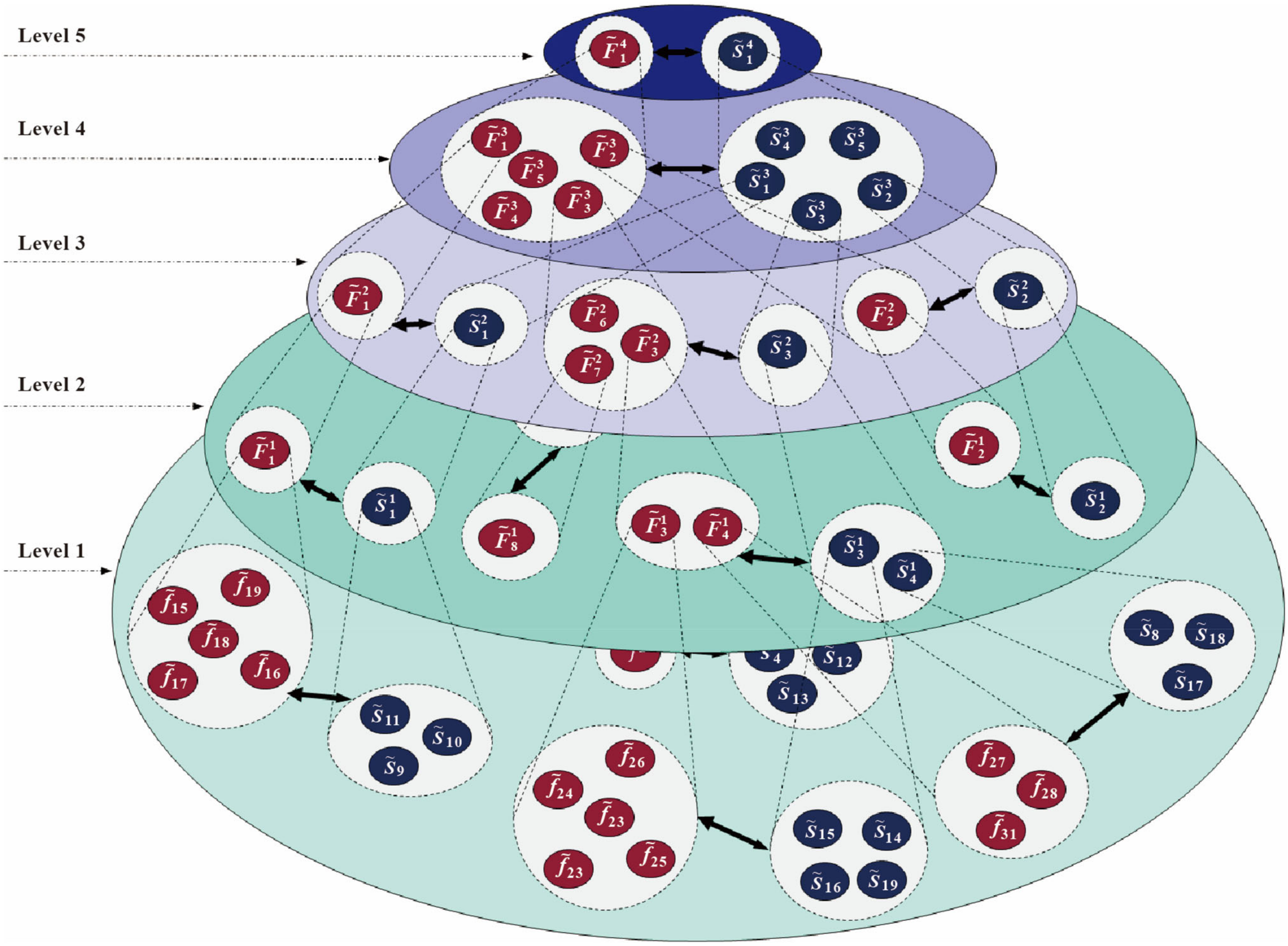


Fig. 8 Holarchy of fuzzy module holon agents

product-service system in the sub-network of fuzzy function agent- fuzzy solution agent, the holonic modular configuration emerges around the strong attractor.

Fuzzy holonic modules overcome the distinction of the continuous-discontinuous traditional modular formation problem. Horizontal and vertical communication of fuzzy holonic modules allows the partial or total reconfiguration of the modular products. The software agent paradigm brings its reactivity, dynamism, agility, and, consequently, its adaptability to the holonic structure.

As the proposed approach suggests, modules, as well as their models, should be designed holistically, as adaptive objects; further developments includes the agent-based digital twin for the intelligent modular design for smart product-service systems. The study of the dynamic behavior of intelligent modules in digitalized design network through the analysis of the creation or destruction of attractors identified and defined in holonic fuzzy module agents is another problem under investigation.

Acknowledgements The project was supported by the National Natural Science Foundation of China (Grant No. 51205242), and Shanghai Science and Technology Innovation Action Plan (Grant No. 16111106402).

Appendix: Fuzzy module holon based on fuzzy function holon

```
#ALPHA CUT= 0.300000
#CAF= 0.650000                       #In every Level the CAF is the same: ε=0

###################################
# Level of fuzzy module holon = 1 #
###################################

#Fuzzy Solution - Fuzzy Function Matrix Holon
0.90    0.90    0.90    0.90    0.40    0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.80    0.90    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.80    0.90    0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.90    0.90    0.90    0.90    0.00    0.80    0.00    0.80
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.00    0.80    0.00    0.90    0.00    0.80    0.80    0.90
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.80    0.00    0.90
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.80    0.90    0.80    0.90    0.80
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.90    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.90    0.00    0.00    0.00    0.80
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.80    0.90    0.80    0.90    0.80    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.90    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.80    0.00    0.90    0.00    0.80    0.80    0.90    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.80    0.00    0.90    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.90    0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.90    0.90    0.90    0.00    0.80    0.00    0.80    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.90    0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.90    0.90    0.90    0.00    0.80
0.00    0.80    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.80    0.00    0.90    0.80    0.00
0.80    0.90    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.80    0.00
0.00    0.90    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.80    0.90    0.80
0.90    0.80    0.90    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.80    0.00    0.00    0.90
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00    0.00
0.00    0.00    0.40    0.90    0.90    0.90    0.90
0.00    0.00    0.80    0.80    0.00    0.00    0.00    0.00    0.00    0.80    0.00    0.80
0.00    0.00    0.00    0.00    0.80    0.00    0.80    0.00    0.00    0.00    0.00    0.80
0.00    0.80    0.00    0.80    0.00    0.00    0.80

#Fuzzy Function Holons
F1.1 = {f19, {f19, f17, f15, f18, f16}}
F1.2 = {f12, {f12, f10, f8, f11, f9}}
F1.3 = {f26, {f26, f24, f23, f22, f25}}
F1.4 = {f31, {f27, f31, f28}}
F1.5 = {f4, {f3, f4, f1, f2}}
```

```
F1.6 = {f5, {f5}}
F1.7 = {f30, {f30}}
F1.8 = {f29, {f29}}
F1.9 = {f21, {f21}}
F1.10 = {f20, {f20}}
F1.11 = {f14, {f14}}
F1.12 = {f13, {f13}}
F1.13 = {f7, {f7}}
F1.14 = {f6, {f6}}

#Fuzzy Solution Holons
S1.1 = {s9, {s11, s10, s9}}
S1.2 = {s7, {s6, s5, s7}}
S1.3 = {s16, {s15, s14, s16, s19}}
S1.4 = {s18, {s8, s18, s17}}
S1.5 = {s1, {s1, s2}}
S1.6 = {s13, {s13, s12, s4, s3}}

#Fuzzy Module Holon / Strong Attractor
(M1.1 / f19)
(M1.2 / f12)
(M1.3 / f26)
(M1.4 / f31)
(M1.5 / f4)
(M1.6 / f5)
(M1.7 / f30)
(M1.8 / f29)
(M1.9 / f21)
(M1.10 / f20)
(M1.11 / f14)
(M1.12 / f13)
(M1.13 / f7)
(M1.14 / f6)


###################################
# Level of fuzzy module holon = 2 #
###################################

#Fuzzy Solution - Fuzzy Function Matrix Holon
9.3   0.0   0.0   0.9   0.0   0.0   0.0   0.0   0.8   0.0   0.0   0.0   0.0   0.0
0.0   9.3   0.0   0.9   0.0   0.0   0.8   0.0   0.0   0.0   0.0   0.0   0.0   0.0
1.6   1.6  10.9   2.5   1.6   0.0   0.0   0.0   0.0   0.0   0.8   0.0   0.0   0.0
0.0   0.0   0.0   5.6   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.9   0.9
0.0   0.0   0.0   0.0   5.3   0.4   0.0   0.0   0.0   0.0   0.0   0.0   0.0   0.0
2.5   2.5   2.5   0.0   0.8   3.6   0.9   0.9   0.9   0.9   0.9   0.9   0.0   0.0

#Fuzzy Function Holons
F2.1 = {F1.1, {F1.1}}
F2.2 = {F1.2, {F1.2}}
F2.3 = {F1.3, {F1.3, F1.4}}
F2.4 = {F1.5, {F1.5}}
F2.5 = {F1.6, {F1.6, F1.14, F1.13, F1.12, F1.11, F1.10, F1.9}}
F2.6 = {F1.7, {F1.7}}
F2.7 = {F1.8, {F1.8}}

#Fuzzy Solution Holons
S2.1 = {S1.1, {S1.1}}
S2.2 = {S1.2, {S1.2}}
S2.3 = {S1.3, {S1.3, S1.4}}
S2.4 = {S1.5 {S1.5}}
S2.5 = {S1.6, {S1.6}}

#Fuzzy Module Holon / Strong Attractor
(M2.1 / F1.1)
(M2.2 / F1.2)
(M2.3 / F1.3)
(M2.4 / F1.5)
(M2.5 / F1.6)
(M2.6 / F1.7)
(M2.7 / F1.8)


###################################
# Level of fuzzy module holon = 3 #
###################################
```

```
#Fuzzy Solution - Fuzzy Function Matrix Holon
9.3   0.0   0.9   0.0   0.0   0.0   0.8
0.0   9.3   0.9   0.0   0.0   0.0   0.8
1.6   1.6  19.0   0.9   0.9   1.6   0.8
0.0   0.0   0.0   0.0   0.0   5.3   0.4
2.5   2.5   2.5   0.0   0.0   0.8   9.0

#Fuzzy Function Holons
F3.1 = {F2.1, {F2.1}}
F3.2 = {F2.2, {F2.2}}
F3.3 = {F2.3, {F2.3, F2.6, F2.7}}
F3.4 = {F2.4, {F2.4}}
F3.5 = {F2.5, {F2.5}}

#Fuzzy Solution Holons
S3.1 = {S2.1, {S2.1}}
S3.2 = {S2.2, {S2.2}}
S3.3 = {S2.3, {S2.3}}
S3.4 = {S2.4, {S2.4}}
S3.5 = {S2.5, {S2.5}}

#Fuzzy Module Holon / Strong Attractor
(M3.1 / F2.1)
(M3.2 / F2.2)
(M3.3 / F2.3)
(M3.4 / F2.4)
(M3.5 / F2.5)


###################################
# Level of fuzzy module holon = 4 #
###################################

#Fuzzy Solution - Fuzzy Function Matrix Holon
9.3   0.0   0.9   0.0   0.8
0.0   9.3   0.9   0.0   0.8
1.6   1.6  20.8   1.6   0.8
0.0   0.0   0.0   5.3   0.4
2.5   2.5   2.5   0.8   9.0

#Fuzzy Function Holons
F4.1 = {F3.3, {F3.1, F3.2, F3.3, F3.4, F3.5}}

#Fuzzy Solution Holons
S4.1 = {S3.3, {S3.1, S3.2, S3.3, S3.4, S3.5}}

#Fuzzy Module Holon / Strong Attractor
(M4.1 / F3.3)


###################################
# Level of fuzzy module holon = 5 #
###################################

#Fuzzy Solution - Fuzzy Function Matrix Holon

71.40

#Fuzzy Function Holons
F5.1 = {F4.1, {4.1}}

#Fuzzy Solution Holons:
S5.1 = {S4.1, {S4.1}}

#Fuzzy Module Holon / Strong Attractor
(M5.1 / F4.1)
```

References

1. Fodor JA (1983) The modularity of mind. MIT Press, Cambridge
2. Salvador F (2007) Toward a product system modularity construct: literature review and reconceptualization. IEEE Trans Eng Manag 54:219–240
3. Sosa ME, Eppinger SD, Rowles CM (2007) A network approach to define modularity of components in complex products. J Mech Des 129:1118–1129
4. Bonvoisin J, Halstenberg F, Buchert T et al (2016) A systematic literature review on modular product design. J Eng Des 27:488–514
5. Kashkoush M, El Maraghy H (2017) Designing modular product architecture for optimal overall product modularity. J Eng Des 28:293–316
6. Issa H, Ostrosi E, Lenczner M et al (2017) Fuzzy holons for intelligent multi-scale design in cloud-based design for configurations. J Intell Manuf 28:1219–1247
7. Ong SK, Xu QL, Nee AYC (2008) Design reuse in product development modeling, analysis and optimization. World Scientific Publishing, Singapore
8. Fogliatto FS, da Silveira GJ (eds) (2011) Mass customization: engineering and managing global operations. Springer, London
9. Stone RB, Wood KL (2000) Development of a functional basis for design. J Mech Des 122:359–370
10. Suh NP (2001) Axiomatic design: advances and applications. Oxford University Press, New York
11. Browning TR (2001) Applying the design structure matrix to system decomposition and integration problems: a review and new directions. IEEE Trans Eng Manag 48:292–306
12. Lange MW, Imsdahl A (2014) Modular function deployment: using module drivers to impart strategies to a product architecture. In: Siddique Z, Hölttä-Otto K, Simpson TW et al (eds) Advances in product family and product platform design. Springer, New York, pp 91–118
13. Fischer JO (2008) Cost-conscious design: proven methods and information systems for the design process. Springer-Verlag, Berlin
14. Schuh G, Arnoscht J, Aleksic S (2012) Systematic design of communalities in products and processes. ZWF Z Für Wirtsch Fabr 107:322–326
15. Ostrosi E, Stjepandić J, Fukuda S et al (2014) Modularity: new trends for product platform strategy support in concurrent engineering. Adv Transdiscipl Eng. https://doi.org/10.3233/978-1-61499-440-4-414
16. Zhuo L, Yoke SW, Kim SL (2008) Integrated approach to modularize the conceptual product family architecture. Int J Adv Manuf Technol 36:83–96
17. Baylis K, Zhang G, McAdams DA (2018) Product family platform selection using a Pareto front of maximum commonality and strategic modularity. Res Eng Des 29:547–563
18. Raudberget D, Elgh F, Stolt R et al (2019) Developing agile platform assets—exploring ways to reach beyond modularisation at five product development companies. Int J Agile Syst Manag 12:311–331
19. Jiao J, Simpson TW, Siddique Z (2007) Product family design and platform-based product development: a state-of-the-art review. J Intell Manuf 18:5–29
20. Stjepandić J, Ostrosi E, Fougères AJ et al (2015) Modularity and supporting tools and methods. In: Stjepandić J, Wognum N, Verhagen JCW (eds) Concurrent engineering in the 21st century. Springer International Publishing, Cham, pp 389–420
21. Bruun HPL, Mortensen NH, Harlou U et al (2015) PLM system support for modular product development. Comput Ind 67:97–111
22. Sun J, Chai N, Pi G et al (2017) Modularization of product service system based on functional requirement. Procedia CIRP 64:301–305
23. Tinnilä M, Lipponen M, Rajahonka M et al (2010) The concept of modularity: diffusion from manufacturing to service production. J Manuf Technol Manag 21:355–375
24. Jung M (2005) Controlling of modular product families in the automotive industry. Dissertation, Deutscher University, Wiesbaden
25. Piller FT, Tseng MM (2010) Handbook of research in mass customization and personalization. World Scientific, Singapore
26. Custódio DT, Roehe VGL, Nunes FL et al (2018) Variant product configuration of industrial air handling units in a MTO environment. Int J Adv Manuf Technol 95:1025–1037
27. Kreng VB, Lee TP (2004) Modular product design with grouping genetic algorithm—a case study. Comput Ind Eng 46:443–460
28. Sinha K, Suh ES (2018) Pareto-optimization of complex system architecture for structural complexity and modularity. Res Eng Des 29:123–141
29. Xu X, Zhang W, Ding X (2018) Modular design method for filament winding process equipment based on GGA and NSGA-II. Int J Adv Manuf Technol 94:2057–2076
30. Yu S, Yang Q, Tao J et al (2011) Product modular design incorporating life cycle issues - group genetic algorithm (GGA) based method. J Clean Prod 19:1016–1032
31. Lee MJ, Case K, Marshall R (2016) Product lifecycle optimisation of car climate controls using analytical hierarchical process (AHP) analysis and a multi-objective grouping genetic algorithm (MOGGA). J Eng Sci Technol 11:1–17
32. Pandremenos J, Chryssolouris G (2011) A neural network approach for the development of modular product architectures. Int J Comput Integr Manuf 24:879–887
33. Fougères AJ, Ostrosi E (2013) Fuzzy agent-based approach for consensual design synthesis in product configuration. Integr Comput-Aided Eng 20:259–274
34. Albrecht SV, Stone P (2018) Autonomous agents modelling other agents: a comprehensive survey and open problems. Artif Intell 258:66–95
35. Valckenaers P, Van Brussel H, Hadeli BO et al (2003) On the design of emergent systems: an investigation of integration and interoperability issues. Eng Appl Artif Intell 16:377–393
36. Deciu ER, Ostrosi E, Ferney M et al (2005) Configurable product design using multiple fuzzy models. J Eng Des 16:209–233
37. He B, Pan Q, Deng Z (2018) Product carbon footprint for product life cycle under uncertainty. J Clean Prod 187:459–472
38. Ostrosi E, Fougères AJ (2011) Optimization of product configuration assisted by fuzzy agents. Int J Interact Des Manuf 5:29–44
39. Ostrosi E, Haxhiaj L, Ferney M (2008) Configuration grammars: powerful tools for product modelling in CAD systems. In: Curran R, Chou SY, Trappey A (eds) Collaborative product and service life cycle management for a sustainable world. Springer, London, pp 469–481
40. Ostrosi E, Fougères AJ, Ferney M et al (2012) A fuzzy configuration multi-agent approach for product family modelling in conceptual design. J Intell Manuf 23:2565–2586
41. Huang CC (2004) A multi-agent approach to collaborative design of modular products. Concurr Eng 12:39–47
42. Issa H, Ostrosi E, Lenczner M et al (2013) Influence of functional knowledge structuring for modular design. Adv Mater Res 651:595–600
43. Overmeyer L, Bentlage A (2014) Small-scaled modular design for aircraft wings. In: Denkena B (ed) New production technologies in aerospace industry. Springer International Publishing, Berlin, pp 55–62

44. Kim S, Moon SK (2019) Eco-modular product architecture identification and assessment for product recovery. J Intell Manuf 30:383–403
45. Zhang Z, Chai N, Ostrosi E et al (2019) Extraction of association rules in the schematic design of product service system based on Pareto-MODGDFA. Comput Ind Eng 129:392–403
46. Elgh F (2014) Automated engineer-to-order systems—a task-oriented approach to enable traceability of design rationale. Int J Agile Syst Manag 7:324
47. André S, Elgh F (2018) Modeling of transdisciplinary engineering assets using the design platform approach for improved customization ability. Adv Eng Inform 38:277–290
48. Stjepandic J, Wognum N, Verhagen WJC (eds) (2019) Systems engineering in research and industrial practice: foundations, developments and challenges. Springer, Berlin. https://doi.org/10.1007/978-3-030-33312-6
49. Krstic D (2004) Computing with analyzed shapes. In: Design computing and cognition. Springer, pp 397–416
50. Valckenaers P, Van Brussel H (2005) Fundamental insights into holonic systems design. In: International conference on industrial applications of holonic and multi-agent systems. Springer, pp 11–22
51. Valckenaers P, Van Brussel H, Wyns J et al (1998) Designing holonic manufacturing systems. Robot Comput Integr Manuf 14:455–464
52. Van Brussel H, Bongaerts L, Wyns J et al (1999) A conceptual framework for holonic manufacturing: Identification of manufacturing holons. J Manuf Syst 18:35
53. Van Brussel H, Wyns J, Valckenaers P et al (1998) Reference architecture for holonic manufacturing systems: PROSA. Comput Ind 37:255–274
54. Leitão P, Restivo F (2008) A holonic approach to dynamic manufacturing scheduling. Robot Comput Integr Manuf 24:625–634
55. Leitão P, Restivo F (2006) ADACOR: a holonic architecture for agile and adaptive manufacturing control. Comput Ind 57:121–130
56. Arai T, Aiyama Y, Sugi M et al (2001) Holonic assembly system with plug and produce. Comput Ind 46:289–299
57. Hsieh FS (2008) Robustness analysis of holonic assembly/disassembly processes with Petri nets. Automatica 44:2538–2548
58. Hashemian M (2005) Design for adaptability. Dissertation, University of Saskatchewan, Saskatoon, Saskatchewan, Canada
59. Ostrosi E, Fougères AJ, Ferney M (2012) Fuzzy agents for product configuration in collaborative and distributed design process. Appl Soft Comput 12:2091–2105

Egon Ostrosi holds a PhD from the University of Strasbourg - France. He is currently Associate Professor at the Pôle Industrie 4.0 at University of Technology of Belfort – Montbéliard (UTBM) and conducts also his research at the Pôle ERgonomie et COnception des Systèmes (ERCOS) - ELLIADD (E.A.4661), Université de Bourgogne Franche-Comté. His current research concerns design and modelling, intelligent product design and development, cloud-based design and manufacturing and design theory. He has been actively involved with the development of methods for product design, CAD product modelling, product configuration, and collaborative and distributed design process.

Alain-Jérôme Fougères holds a PhD in Artificial Intelligence from the University of Technology of Compiègne - France. He is a Professor of Computer Science at ECAM Rennes and he conducts his research at Laboratoire Informatique et Télécommunications. His areas of interests and his scientific contributions concerns: (1) assistance to the redaction of formal specifications, where problems focused on the natural language processing, the knowledge representation, and the techniques of formal specifications; and (2) design of agent-based systems, in particular architecture, interactions, communication, and cooperation problems.

Zai-Fang Zhang holds a PhD in Industry Engineering from Shanghai Jiao Tong University - China. He is currently Associate Professor at the School of Mechatronic Engineering and Automation, Shanghai University, Shanghai, P.R. China. His current research concerns product service system design, decision-making methods, and configuration design. He has been actively involved with the development of methods for intelligent decision making in the collaborative and distributed design processes. His current research concerns Product Service System design, decision-making methods, knowledge engineering and product configuration. He has been actively involved with the development of methods for decision making in the collaborative and distributed design processes.

Josip Stjepandić is the Head of business unit 3D Product Creation at the PROSTEP AG, the leading product data integration company worldwide. He is in charge for consultancy and solution development in the areas design methodology, supplier integration, systems engineering, knowledge-based engineering, product data validation and visualisation, configuration management and CAD data exchange for many industries (automotive, aerospace, shipbuilding, and machinery). In the past, he gathered strong research experience in these areas including his current research on systems engineering, modular design, and digital factory. He is the author of various scientific articles and publications in the field of virtual product creation and a board member of the International Society of Transdisciplinary Engineering (ISTE).