



HAL
open science

Intelligent modular design with holonic fuzzy agents

Egon Ostrosi, Alain-Jérôme Fougères, Zai-Fang Zhang, Josip Stjepandić

► **To cite this version:**

Egon Ostrosi, Alain-Jérôme Fougères, Zai-Fang Zhang, Josip Stjepandić. Intelligent modular design with holonic fuzzy agents. *Advances in Manufacturing*, 2021, 9 (1), pp.81-103. 10.1007/s40436-020-00331-0 . hal-04647256

HAL Id: hal-04647256

<https://hal.science/hal-04647256>

Submitted on 13 Jul 2024

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

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

Intelligent Modular Design with Holonic Fuzzy Agents

Egon OSTROSI^{a1}, Alain-Jérôme FOUGÈRES^b, Zaifang ZHANG^c, Josip STJEPANDIĆ^d,

^a Université de Bourgogne Franche-Comté, UTBM, Pôle Industrie 4.0, Pôle ERgonomie et Conception des Systèmes ERCOS/ELLIADD EA4661, Belfort, FRANCE

^b ECAM Rennes, Campus de Ker Lann - Bruz, Rennes, FRANCE

^c School of Mechatronic Engineering and Automation, Shanghai University, Shanghai, 200240 PR CHINA

^d PROSTEP AG, Darmstadt, GERMANY

Abstract

The modular designs in recent times use various technologies along with multiple models. The effective utilization of the 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. The design of modular intelligent products requires to consider dynamically and holistically customer requirements, product functions, solutions, process or service constraints, and their fuzziness. This paper proposes using holonic fuzzy agents to fulfil the properties of intelligent models and the functional requirements of modular design. The set of fuzzy functions agents and their corresponding fuzzy solutions agents are found from the customization of the Product Service System in the sub-network fuzzy function agent–fuzzy solution agent. Then, from the attractor agent recognition, the fuzzy function and fuzzy solution agents interact to form the holonic fuzzy module agents. Holonic structure's fundamental property is also followed by communication. The flexibility and agility of the software agent make holonic structure of intelligent modules adaptable. An application illustrates the proposed intelligent modular design.

Article Highlights

- The formalization of holonic modules and corresponding intelligent models are proposed.
- An intelligent module holon is created around structurally stable elements called attractors.
- An intelligent module holon emerges in the network of *requirement fuzzy agents–function fuzzy agents–solution fuzzy agents–constraints fuzzy agents*
- The dynamic of intelligent module holon depend on the dynamics of conflict between agent attractors.

Keywords: Modularity, Modular Design, Intelligent Design, Holonic Agents, Product Platform, Product Variety

¹ Corresponding Author, E-mail: egon.ostrosi@utbm.fr

1 Introduction

Development of a sustainable and intelligent design requires modularity to be an integral part of the design. The use of modularity in product design covers technical and business aspects from quantitative as well as qualitative angles. The human cognitive abilities are considered to be roots of modularity [1].

A product is termed as modular when its components and sub-modules can be combined seamlessly [2]. The modules are required to work independently and should not be dependent on other modules. A product consists of several components that are connected to work as a whole unit. The lack of connectivity between its components defines the modularity [3]. The similar interconnected physical components are encapsulated to form the modules. The flow of information between components of modules allows it to meet functional requirements. The modules are characterized by minimal interaction with external components and maximum interaction between components within the module. The functionality of the module is limited to predefined scenarios that control its functionality using interface and connections. The functionality of the modules can be controlled using three ways: (a) the interface between various components within the module, (b) the interconnection with nonadjacent components within the product, and (c) the behaviour of components as a bridge between other components via interfaces.

The intelligent products are aimed to provide flexibility in design to the architects and system designers. The methodology used in the design of intelligent products is important. Handling of modularity and its management require the use of intelligent multidisciplinary collaborative and distributed platforms. This challenge can be best met using multi-agent paradigm and provide innovative technologies that can handle the dynamic environment.

Holon, consisting of interrelated semi-autonomous modules, is an advantageous way of modelling modular intelligent systems. Each interrelated semi-autonomous modules in Holon has a hierarchical structure ending with elementary modules. Concerning the holon, the concept of the attractor is used to deal with the form and stability of the module [4]. An attractor is a stable element or a stable group of elements toward which a module formation tends to evolve. This paper uses product functions or physical product solutions as attractors. Thus, the functions and solutions of a product are identified. It means that functions and solutions are designed as entities capable of exhibiting intelligent behaviour. 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 subnetworks of holonic fuzzy agents.

The paper is organized as follows: in the second section, the state of art analyses and discusses the modular design in relation to different aspects. In the third section, a holonic fuzzy agent modelling is presented. In the fourth section, the proposed holonic fuzzy agent model for intelligent holonic modules formation is outlined. In the fifth section, the application of the proposed model is shown. Finally, conclusions and perspectives of this research study are presented.

2 Modular Design: the State of Art

In this section, the research related to modular design is analyzed and discussed with regard to four perspectives: (1) Technologies, (2) Product Service System, (3) Artificial Intelligence and (4) Industrial Applications.

2.1 Modular design and Technologies

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

The information is partitioned into three categories to achieve modularity [5]: *Architecture, Interfaces, and Standards*. The module's architecture provides module functions and specifications. The modules 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 a function to the modulus physical components. The module interfaces are crucial for proper utilization of modules. Modular products attributes include [2, 6]: *commonality of modules, combinability of modules, function binding, interface standardization, loose coupling of components*.

The processes mentioned below are used for modular design [6]: (1) *the identification of product architecture and reusable components from existing products*, (2) *the agglomeration and adaptation of singular building blocks into modules to derive a new design*, and (3) *assessment of product performance and cost*.

Researchers have suggested various modular design strategies such as Functional Modeling [7], Axiomatic Design [8], Design Structure Matrix [9], Modular Function Deployment [10]. Variant Mode and Effects Analysis (VMEA) [11] can be used along with the architecture development process [12]. 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 into account various issues.

The integration of holistic lifecycle characteristics of a product's architecture is a critical issue [13]. The product architecture depends on its configuration and modularity [14]. The life cycle design must take into account the product architecture. However, the theoretical foundation required for developing modular product architecture is missing. The industry has considered it as intellectual knowledge instead of the scientific-engineering issue. The modularity concept is also followed while designing a family of products and platform-based product development [15]. 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. The platform includes the basic architecture and the interfaces to the optional items to make an end product different from others. The holistic view of modular design necessitates the inclusion of backend and front-end issues including product family, product portfolio, manufacturing and production, platform-based product family design, and supply chain management [16].

The current trend is to use tools like product configurators for the development of the modular design [17]. Using interaction and mutual integrations, these tools provide optimal functionality. The product configurator, a commercial and multifunctional tool, also interfaces between delivery and sales in an enterprise environment. The configurator tool ensures compliance with the specifications defined for the model. The product logic based on rules and limitations is implemented using the configurator. The customer provides his requirement through the user interface, and then the product meeting his requirement is selected. After due diligence on validity and cost analysis of models to be used, the bill of material is drawn. The integration of product configuration is required with other IT systems such as CAX technologies, Product Lifecycle Management (PLM), and Enterprise Resource Planning (ERP). However, management and synchronization of configuration data with the above mentioned IT applications require proper attention to the deployment of integrated product configuration.

In recent Product Data Management (PDM) systems, the modular product's structure is mapped to a common structure [15, 18]. PDM system database manages optional items like other items in master records along with their attributes. The management of an article depends on the bill of quantities required. The PDM systems handle the changes in product structure using order neutral BOMs along with varying and optional positions. This methodology allows the creation of explicit BOMs required for production and benefits product development.

However, managing large variants of the products makes data management complicated and require 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, and structures, and explicit variants are managed using configuration and clone modules. The report generation module is used for generation of reports as and when required.

2.2 Modularity and Product Service System

The modularity, from a management perspective, can be taken as a strategy for efficient design of complex products, procedures, and services that bring rationalization in the system. The modularity may be taken as a product development methodology that is used from several product designs [19, 20].

For products with very distinctive features, modularity is not required as creativeness is necessary for taking care of specialized haptics, styling or colours. As the number of products requiring distinctive features is small, the modular design is not cost-effective. Only a holistic controlling approach for the integration of product variants can lead to monetary benefits [21]. The holistic approach allows assessment and holistic management of modular product families using Modularity-Balanced-Score-Card (M-BSC). The modular product development also requires to take 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 customers requirement [22]. Mass customization is an amalgamation of mass production and customization. Two approaches used for mass customization are mass and craft (single-piece) production. The mass production uses standardization and scale of economies to create cost-effective products. The craft production, on the other hand, uses a high level of customization to meet the specific requirement of customers.

Creation of customized products calls for using specifically created components along with standardized and configurable modules. This requires identifying a fixed and a variable area of product structure. The variable area is then used for customization. Generally, configuration systems are used for product customization. Mass customization of product design requires the use of generic conceptual procedures [23]. These generic conceptual procedures require business processes analysis and redesign, company's product portfolio analysis and modelling, configuration software selection, software programming, and implementation as well as the further configuration system development towards Product-Service System (PSS).

2.3 Modular design and Artificial Intelligence

Artificial intelligent allows the creation of efficient computing techniques. Evolutionary algorithms are used to identify separable modules and simultaneously optimize the number of modules [24], product configuration based on modules [25], as well as consideration of the product lifecycle [26].

The use of multi-objective grouping genetic algorithm (MOGGA) can create several alternative product modularizations [27]. In this ambit, it is shown that modules could be identified using neural network algorithms [28]. Complex uncertain problems with inherently distributed global knowledge shared among several agents can be solved using the agent paradigm [29]. The flexibility of the agent-based system due to their distribution and decentralization allows them to be used in an industrial environment [30]. Multi-agent systems have been found useful in modelling and simulation of adaptive or self-adaptive systems such as emergence and self-organization [31]. Use of the reactive and autonomous agents allow

updating and use information from other agents as well as their environment. The product design configuration uses fuzziness in early-stage [32, 33] of the product lifecycle design [32, 33]. Fuzzy agents optimize and model information fuzziness model, knowledge fuzziness, and interaction fuzziness in distributed and collaborative design [34]. Fuzzy agents process fuzziness information after receiving them and interact within a multiagent system. A fuzzy agent can find out its interest with another fuzzy agent. The use of configuration grammars allows formalization of configurations structural problems [35] and implementation is done in grammar-based multi-agent platform [36]. The decision required in modular product collaborative design can also be supported by agent-based systems with a large knowledge base [37].

2.4 Modularity in Industrial applications

Dilemmas, doubts and uncertainties, are emerging during modular design. What is the right method to apply for modularization?: a) function based modularization, b) component-based modularization or c) function-component modularization. These represent one of the most common situations in which engineers experience dilemmas in product modularization [38]

An analysis of engine blocks modular design shows that the functional knowledge and its proper restructuring by the designer derive the modular design [38]. The modularization approaches for conventional products and that for 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 [39]. Ideal module size required can be determined using the following four steps: design, technical feasibility, economic viability, and tool development. The small modules size is found to facilitate manufacturing and logistics. In case of an aeroplane, the increase in the number of modules leads to a rise in fuel consumption due to a higher number of interfaces required between the modules. The predicted lifecycle costs provide a proper measure of determining ideal module size. The minimization of total life-cycle costs leads to global optimization.

The modular design uses knowledge-based technologies integrated with numerical technologies. PADM and 3D-CAD systems are used for designing and documenting the plant designs where machines with more than 10,000 parts are there [17]. The design is customized as per customer and market requirements, the technical feasibility of the product, business requirements, and manufacturing ability. The design should not reduce the offering and have unnecessary complexity. Web-based product configurator can be used for the proper product, while KBE and CAD models are used for product presentation. The KBE and CAD models are highly adaptable, and product configurator is used to provide the required product variant required by the user. The configurator communicates directly with various sections, including internal systems (ERP, CRM, PDM) without involving CAD system. This methodology enables the generation of bottom-up relationship knowledge and use of ERP object list for setting up assembly plans. CNN machine design [40] or components' design in the automobile sector [41] use product configuration with KBE. 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. ERP system provides tools for business operations, whereas the PLM system provides tools from product development.

2.5 Modularity and Further Development

The modular product design removes system-based interdependency hurdles. The modular design requires a rational and coherent design process that takes care of constraints and technical requirement at the beginning of the process. Modularity assessment includes ways to

increase the effectiveness of modularity. Judging the relationship between effectiveness and level of modularity is an open-ended issue.

The design of a flexible product that has good adaptability needs extra efforts in its design and manufacturing. The design and development of intelligent systems require modules to have adaptability and suitability to allow integration with different types of large systems. An intelligent product maximizes the design space for system designers and architects.

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 between them. The self-sustainable models and products can be designed using these approaches. Thus, the design of intelligent products necessitates the use of intelligent modules. The development of intelligent modular products requires intelligent models and intelligent tools. The modularity and variant management can be handled nicely using intelligent collaborative and distributed platforms. The holonic and multi-agent model has shown the potential to take up this challenge and provide innovative technologies that can tackle dynamic involved in modular product development. The paper looks into this issue.

3 Modular design through holonic fuzzy agent

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

3.1 Equations for Holonic modules

A product is described using its decompositions by engineers. The decompositions of a product can be defined in many ways and depend on the information that requires highlighting [42]. 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* [4]. The module's decomposition ability together with the definition of holon defines its complete property. The double-heading indicates the inclusion of holon in the holarchy, which is a typical vertical arrangement. The modules in nested hierarchical order and progressive accumulation can be represented by holarchy. Holons are considered a unit due to their being self-complete, but they can be part of a larger system.

Many applications such as holonic systems design [43], holonic manufacturing [44–46], manufacturing control and scheduling [47, 48], holonic assembly [49, 50], and product design [51] use the concept of holon.

Three cases of holonic modularization can be distinguished: a) physical solution (or component) based modularization, a) function based modularization and c) 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 $[S]$, the fuzzy relationship between the elements of $[S]$ can be characterized by different degrees of affinity. This dependency can be written as:

(3.1)

$$\left[\tilde{S}^{k=0} \right] = \left[\tilde{C} \right]^{k=0} \left[\tilde{S}^{k=0} \right]$$

where: $\left[\tilde{S}^{k=0} \right]$ is the initial vector ($k=0$) representing the fuzzy physical solutions $\tilde{S}^{k=0} = \{\tilde{s}_1^0, \tilde{s}_2^0, \dots, \tilde{s}_n^0\}$ and $\left[\tilde{C} \right]^{k=0}$ 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 $\tilde{\mathfrak{R}}_1(\tilde{s}_j^{k=0}, \tilde{s}_{j'}^{k=0})$ can be defined. The fuzzy relationship $\tilde{\mathfrak{R}}_1(\tilde{s}_j^{k=0}, \tilde{s}_{j'}^{k=0})$ represented by the matrix $\left[\tilde{C} \right]^{k=0}$, is characterized by the membership function $\mu_{\tilde{\mathfrak{R}}_1}(\tilde{s}_j^{k=0}, \tilde{s}_{j'}^{k=0})$ which takes values between 0 and 1. In practice, the affinity relationship between solutions may sometimes be 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 $\tilde{\mathfrak{R}}_1(\tilde{s}_j^{k=0}, \tilde{s}_{j'}^{k=0})$ and is characterized by the fuzziness. The holonic decomposition of the *fuzzy structural network*, corresponding to the holonic decomposition of the matrix $\left[\tilde{C} \right]^{k=0}$ into sub-matrixes, will yield the holonic decomposition of the product into *fuzzy structural module holons*:

(3.2)

$$\left[\tilde{S}_j^k \right] = \left[\tilde{C}_j^k \right] \left[\tilde{S}_j^k \right]$$

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

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

(3.3)

$$\left[\tilde{F}^{k=0} \right] = \left[\tilde{B} \right]^{k=0} \left[\tilde{F}^{k=0} \right]$$

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

The degree of interaction characterizes the product's fuzzy functions set $\left[\tilde{F}^{k=0} \right]$. These interactions can be defined by a fuzzy relationship $\tilde{\mathfrak{R}}_2(\tilde{f}_j^{k=0}, \tilde{f}_{j'}^{k=0})$ represented by the matrix $\left[\tilde{B} \right]^{k=0}$. In this, the membership function $\mu_{\tilde{\mathfrak{R}}_2}(\tilde{f}_j^{k=0}, \tilde{f}_{j'}^{k=0})$ can have a value between 0 and 1. The value 1 of membership confirms the identification of the relationship between product functions by the designer whereas value 0 negates the identification. When a functional network takes characteristics of fuzziness, it is termed as a *fuzzy functional network*. The *fuzzy functional network's* holonic decomposition is holonic decomposition of the matrix $\left[\tilde{B} \right]^{k=0}$

into sub-matrixes. This decomposition provides holonic decomposition of the product into *fuzzy functional module holons*:

$$\left[\tilde{F}_j^k \right] = \left[\tilde{B}_j^k \right] \left[\tilde{F}_j^k \right] \quad (3.4)$$

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

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

$$\left[\tilde{F}^{k=0} \right] = \left[\tilde{A} \right]^{k=0} \left[\tilde{S}^{k=0} \right] \quad (3.5)$$

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

The set of functions of each product can be taken care of using various physical solutions sets. Alternatively, the set of functions of each product can be partially taken care of using physical solutions 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 $\left[\tilde{A} \right]^{k=0}$ represents a fuzzy relationship $\tilde{\mathfrak{R}}_3(\tilde{f}_i^{k=0}, \tilde{s}_j^{k=0})$. The fuzzy relationship provides relations between the fuzzy set of functions $\tilde{F}^{k=0} = \{ \tilde{f}_1^0, \tilde{f}_2^0, \dots, \tilde{f}_m^0 \}$ and the fuzzy set of physical solutions $\tilde{S}^{k=0} = \{ \tilde{s}_1^0, \tilde{s}_2^0, \dots, \tilde{s}_n^0 \}$. The corresponding membership function $\mu_{\tilde{\mathfrak{R}}_3}(\tilde{f}_i^{k=0}, \tilde{s}_j^{k=0})$ defined in $[0, 1]$ provides the extent to which a product's function can be met using alternative physical solutions set. The membership function $\mu_{\tilde{\mathfrak{R}}_3}(\tilde{f}_i^{k=0}, \tilde{s}_j^{k=0})$ provides designer's view of product functions available using a set of alternative physical solutions. In this manner, the fuzzy relationship between product functions and related physical solutions provides 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*:

$$\left[\tilde{F}_j^k \right] = \left[\tilde{A}_j^k \right] \left[\tilde{S}_j^k \right] \quad (3.6)$$

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

3.2 Holonic fuzzy agent modelling

The adaptive systems modelling and simulation takes care of many features including: several heterogeneous interacting products, unpredictable emergence, complicated transition laws, multiple scales, ill-defined boundaries, combinatorial explosion, adaptivity to changing environments, sensitive dependence on initial conditions, co-evolving subsystems, multilevel or non-equilibrium dynamics, and interaction and self-organization of autonomous agents [52]. For our work, we have focused on the following three properties out of many complex adaptive systems characteristics outlined above:

- *Self-organization property.* An organization is formed by distributing a set of elements in a hierarchical level. A self-organizing system dynamically adjusts this structure as per the requirement, and this property is base for module formation.
- *Nonlinearity property.* A complex system's response and behaviour are not fixed and are governed by the feedback loops and nonlinear relationships present. The response of the system to these linear or non-linear behaviours determines the nonlinear property of the module. This nonlinear property is necessary for the modules and is handled by the reactivity of the module's holonic fuzzy agents.
- *Emerging behaviour property.* From nonlinear interactions in complex systems and their self-organized or chaotic structures, behaviours emerge; this property will be fundamental for holonic fuzzy agents' generation and finally module formation.

Holonic fuzzy agents can also be modelled to integrate the behaviour of both fuzzy agents and holonic agents. Let's start by defining a model of fuzzy agent adapted from [29]: the definition of a fuzzy agent-based system \tilde{M}_α is given in (Table 1, Eq. 3.7); the definition of a fuzzy agent $\tilde{\alpha}_i$, whose behaviour is of *feedback loop* type $\langle \text{perceive, decide, act} \rangle$ in a fuzzy context, is given in (Table 1, Eq. 3.8); and the definition of a fuzzy interaction $\tilde{t}_i \in \tilde{I}$ between two fuzzy agents is given in (Table 1, Eq. 3.12).

Table 1. The fuzzy agent model

Elements of model	Meaning
$\tilde{M}_\alpha = \langle \tilde{A}, \tilde{I}, \tilde{P}, \tilde{O}, \Phi_{\tilde{A}} \rangle$ <p>(3.7)</p>	Fuzzy agent-based system where $\tilde{A}, \tilde{I}, \tilde{P}, \tilde{O}$ and $\Phi_{\tilde{A}}$ are respectively: 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.
$\tilde{\alpha}_i = \langle \Phi_{\Pi(\tilde{\alpha}_i)}, \Phi_{\Delta(\tilde{\alpha}_i)}, \Phi_{\Gamma(\tilde{\alpha}_i)}, K_{\tilde{\alpha}_i} \rangle$ <p>(3.8)</p>	Fuzzy agent where $\Phi_{\Pi(\tilde{\alpha}_i)}, \Phi_{\Delta(\tilde{\alpha}_i)}$ and $\Phi_{\Gamma(\tilde{\alpha}_i)}$ are respectively: function of observation Erreur ! Source du renvoi introuvable. , function of decision (5.5), function of action (5.5), and knowledge of a fuzzy agent $\tilde{\alpha}_i$. 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 it below), and dynamic knowledge (observed events, internal states).
$\Phi_{\Pi(\tilde{\alpha}_i)} : (E_{\tilde{\alpha}_i} \cup I_{\tilde{\alpha}_i}) \times \Sigma_{\tilde{\alpha}_i} \rightarrow \Pi_{\tilde{\alpha}_i}$ <p>(3.9)</p>	Function of observation where $E_{\tilde{\alpha}_i}, I_{\tilde{\alpha}_i}, \Sigma_{\tilde{\alpha}_i}, \Pi_{\tilde{\alpha}_i}$ are respectively: the finite sets of observed events, interactions, states, and perceptions of a fuzzy agent $\tilde{\alpha}_i$.
$\Phi_{\Delta(\tilde{\alpha}_i)} : \Pi_{\tilde{\alpha}_i} \times \Sigma_{\tilde{\alpha}_i} \rightarrow \Delta_{\tilde{\alpha}_i}$ <p>(3.10)</p>	Function of decision where $\Pi_{\tilde{\alpha}_i}, \Sigma_{\tilde{\alpha}_i}, \Delta_{\tilde{\alpha}_i}$ are respectively the finite sets of perceptions, states, and decisions of a fuzzy agent $\tilde{\alpha}_i$.

$\Phi_{\Gamma(\tilde{\alpha}_i)} : \Delta_{\tilde{\alpha}_i} \times \Sigma \rightarrow \Gamma_{\tilde{\alpha}_i}$ Function of action where $\Delta_{\tilde{\alpha}_i}, \Sigma_{\tilde{\alpha}_i}, \Gamma_{\tilde{\alpha}_i}$ are respectively the finite sets of decisions, states, and actions of a fuzzy agent $\tilde{\alpha}_i$.

(3.11)

$\tilde{t}_i = \langle \tilde{\alpha}_s, \tilde{\alpha}_r, \tilde{\gamma}_c \rangle$ Fuzzy interaction where $\tilde{\alpha}_s$ is the fuzzy agent source of the fuzzy interaction, $\tilde{\alpha}_r$ is the fuzzy agent destination, and $\tilde{\gamma}_c$ is a fuzzy act of communication ($\tilde{\gamma}_c \in \tilde{\Gamma}$ and $\tilde{\Gamma} = \{ \text{inform, diffuse, ask, reply, confirm} \}$).

(3.12)

The model of the fuzzy agent being proposed, a model of holonic fuzzy agent can be defined. This new model, integrating agent, fuzzy and holonic concepts, respects the following holonic properties:

- self-organization, self-adaptation, transcendence, self-efficiency, dynamics, and emergence, are intrinsic properties of the multi-agent paradigm;
- self-similarity, self-preservation, self-expansion, and robustness, are intrinsic properties of a holonic agent paradigm;
- utility, self-well-being, and flexibility, are intrinsic properties of a fuzzy agent paradigm.

Thus, a set \tilde{H} of holonic fuzzy agents, forming a holonic fuzzy agent-based system respecting equation **Erreur ! Source du renvoi introuvable.**, is defined recursively as follows:

- 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 $\tilde{H}_i^0 = (\{\tilde{\alpha}_i\}, \{\tilde{\alpha}_i\}, \emptyset) \in \tilde{H}$;
- $\tilde{H}_i^j = (\tilde{H}_0, \tilde{H}', \tilde{R}) \in \tilde{H}$, where $\tilde{H}' \subseteq \tilde{H}$ is the set of holonic fuzzy agents that participate in \tilde{H}_i^j , $\tilde{H}_0 \subseteq \tilde{H}'$ is the non-empty set of holonic fuzzy agents that represent \tilde{H}_i^j to the environment and is responsible for coordinating the actions inside \tilde{H}_i^j . \tilde{R} defines a fuzzy relationship inside \tilde{H}_i^j (i.e., \tilde{R} specifies the holonic fuzzy agent organization). Each \tilde{H}_i^j respects equation (3.8), where $K_{\tilde{h}_i} = K_{\tilde{\alpha}_i} \cup (\tilde{H}_0, \tilde{H}', \tilde{R})$.

The model of holonic fuzzy agent being defined, we can apply it to the specific problem of intelligent *holonic modular* formation. In this case, fuzzy functions and solutions are fuzzy agentified (3.13). Each fuzzy function of the set \tilde{F} is transformed into a fuzzy agent $\tilde{f}_{[1..m]} \in \tilde{A}_F$, each fuzzy solution of the set \tilde{S} is transformed into a fuzzy agent $\tilde{s}_{[1..n]} \in \tilde{A}_S$, and each generated module of the set \tilde{M} is defined as a fuzzy agent $\tilde{m}_{[1..q]} \in \tilde{A}_M$:

$$\text{Agentification: } \tilde{F}, \tilde{S}, \tilde{M} \rightarrow \tilde{A}_F, \tilde{A}_S, \tilde{A}_M \quad (3.13)$$

Then, from these three sets of fuzzy agents, holonic fuzzy agents are generated (3.14):

$$\text{Holonisation: } \tilde{A}_F, \tilde{A}_S, \tilde{A}_M \rightarrow \tilde{H}_F, \tilde{H}_S, \tilde{H}_M \quad (3.14)$$

where $\tilde{F}_j^k \in \tilde{H}_F$, with $\tilde{F}_j^0 \in \tilde{A}_P$ and $\tilde{F}_j^{k>1} = \{ \tilde{F}_1^{k-1}, \dots, \tilde{F}_p^{k-1} \}$; $\tilde{F}_j^k \in \tilde{H}_F$ with $\tilde{S}_j^0 \in \tilde{A}_S$ and $\tilde{S}_j^{k>1} = \{ \tilde{S}_1^{k-1}, \dots, \tilde{S}_p^{k-1} \}$; $\tilde{M}_j^0 \in \tilde{A}_M$, $\tilde{M}_j^i = \langle \tilde{F}_j^k, \tilde{S}_j^k, \tilde{R}_j^k \rangle$ 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 performs in three phases:

Phase 1: Modelling of fuzzy agents. In this phase, all the communities of fuzzy agents necessary for the holonic modularity are built.

Phase 2: Customization of the Product Service System. In this phase, the fuzzy network between fuzzy functions agent, between solution agent and between fuzzy function agents and solutions agents are customized to respond to different services in the product life cycle.

Phase 3: Formation of holonic modules. In this phase, from fuzzy consensual solution agents and their affinities, the optimal configuration emerges.

4.1 Modelling of fuzzy agents

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

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

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

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

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

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

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

The building of fuzzy multi-agent system takes into consideration: (1) communities building using fuzzy agent, (2) defining interactions between fuzzy function agents and the fuzzy requirement agents, (3) defining interaction among fuzzy function agents, (4) defining interactions between fuzzy solution agents and fuzzy function agents and (5) defining interactions between fuzzy solution agents and fuzzy constraint agents. The expert's knowledge in four considered domains: (a) customer requirements, (b) functions, (c) solution and (d) process or service constraints, allows specifying of interactions between fuzzy agents.

4.2 Customization of the Product Service System

This phase leads to the emergence of a complete customized network of fuzzy agents. Fuzzy requirement agents use their model to work out customer's requirement and after that these are modelled into fuzzy values.

Once the fuzzy set of requirements are available along with 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. 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 get activated.

The fuzzy service constraint agents use their model to get the specifications of the services. The service preferences are then transformed into fuzzy values. When the fuzzy agent solutions

interact with the fuzzy constraint agents, new fuzzy set of active solution agents will emerge that will satisfy considered services constraints.

Finally, the fuzzy set of consensual solution agents come out as a fuzzy set of active solution agents meeting constraints of the services and customer requirements.

4.3 Formation of Holonic modules

In holon architecture, self-similarity can be observed in “horizontal” and “vertical” space. 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 an “*intelligent module within a module.*” Secondly, our proposal postulates that an intelligent module holon is created around defined physical elements such as function or solution called *attractors*.

The *attractors* are structurally stable elements in the complete customized network of requirement fuzzy agents –function fuzzy agents –solution fuzzy agents –constraints fuzzy agents. Properties relating to holonic fuzzy module formation are shown in Table 2.

Table 2. Properties of fuzzy module formation

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

The solution or function can be weighted w_i as per the designer’s preference. As such engineers have knowledge of the acceptable value of variables from various sources and their experience. The acceptable intervals are those where the designer has a positive preference. While specifying the designer’s preference for the fuzzy set elements, the α -levels (acceptable levels) which are equal to or more than preferred threshold value by the designers can be used.

The distributed algorithm of holonic module formation executed by each holonic fuzzy function agent \tilde{F}_i^x is presented in Table 3. The main steps of this algorithm are identified by indices S_i :

- (S0) *initial conditions;*
- (S1) *coefficient of attraction force;*
- (S2) *vertical “self-similarity”;*
- (S3) *fuzzy attractor identification;*
- (S4) *attraction of fuzzy solution holons;*

- (S5) attraction of fuzzy function holons;
- (S6) equivalent fuzzy module holons and transfer;
- (S7) new fuzzy module holons;
- (S8) fuzzy modules within a fuzzy module; and
- (S9) evaluation of end of fuzzy module holons formation.

Table 3 Holonic module formation based on holonic function attractors

Algorithm	Meaning
(S0) Given a set of n fuzzy function holons $\{\tilde{F}_1^0, \dots, \tilde{F}_n^0\} \in \tilde{H}_F$; Given a set of m fuzzy solution holons $\{\tilde{S}_1^0, \dots, \tilde{S}_m^0\} \in \tilde{H}_S$; Given a set of p fuzzy module holons $\{\tilde{M}_1^0, \dots, \tilde{M}_p^0\} \in \tilde{H}_M$;	// initial conditions // initial conditions // initial conditions
EACH function holon agent \tilde{F}_i^x DO {	
(S1) $CAF \leftarrow CAF_{encours}$ $nbX_i^x \leftarrow 0$ $_h \leftarrow false$	// current CAF = CAF _{encours} , for instance, CAF=1.0 // number of solution holons that \tilde{F}_i^x satisfies = 0 // \tilde{F}_i^x is not included in a function holon of higher level
(S2) WHILE ($_h == false$) { FOR j in $1..Card(S^x)$	// while F_i^x is not included in a holon of higher level // loop to know how much solution holons satisfy \tilde{F}_i^x
$nbX_i^x \leftarrow nbX_i^x + Exchange_1(\tilde{F}_i^x, \tilde{S}_j^x)$	// Exchange to determinate it
(S3) IF ($\max(Card(\tilde{X}_i^x), nbX_i^x)$) THEN	// \tilde{F}_i^x is satisfied by the greatest number of solution holons
$A_i \leftarrow \tilde{F}_i^x$	// \tilde{F}_i^x is an attractor of function holons
(S4) generate(\tilde{F}_i^{x+1}) generate(\tilde{S}_i^{x+1})	// generation of a new function holon $\tilde{F}_i^{x+1} \leftarrow \tilde{F}_i^x$ // generation of a new solution holon $\tilde{S}_i^{x+1} \leftarrow \tilde{S}_i^x$
(S5) calculate($CAF_i, \tilde{S}_i^x, \tilde{S}_i^{x+1}$) FOR k in $1..Card(\tilde{S}_i^x)$	// calculation of $CAF_i \leftarrow Card(\tilde{S}_i^{x+1})/Card(\tilde{S}_i^x)$ // loop to get the list of function holons satisfied by the
$F_m \leftarrow Exchange_2(\tilde{F}_i^x, \tilde{S}_k^x)$	// solution holons that satisfy \tilde{F}_i^x
(S6) FOR l in $1..Card(F_m)$ $CAF_i.[\tilde{F}_i^x] \leftarrow Exchange_3(\tilde{F}_i^x, \tilde{F}_l^x, \tilde{S}_i^{x+1})$	// loop for each function holon satisfied by solutions of \tilde{S}_i^x // exchange to get the CAF of \tilde{F}_i^x
IF ($CAF_i.[\tilde{F}_i^x] \geq CAF_{encours}$) THEN	// \tilde{F}_i^x is an attractor for the function holon \tilde{F}_i^x
$\tilde{F}_i^{x+1} \leftarrow \tilde{F}_i^{x+1} \cup \tilde{F}_l^x$ inform($\tilde{F}_i^x, \tilde{F}_l^x, \tilde{F}_i^{x+1}$)	// \tilde{F}_l^x is included in the new function holon \tilde{F}_i^{x+1} // \tilde{F}_i^x is informed of inclusion in the function holon \tilde{F}_i^{x+1}
ELSE inform($\tilde{F}_i^x, \tilde{F}_l^x, !\tilde{F}_i^{x+1}$)	// \tilde{F}_l^x is informed of not inclusion in the holon \tilde{F}_i^{x+1}
(S7) generate($\tilde{M}_i^{x+1}, A_i, \tilde{F}_i^{x+1}, \tilde{S}_i^{x+1}$)	// generation of the new module holon \tilde{M}_i^{x+1}
ELSE	// \tilde{F}_i^x is not satisfied by the greatest number of solutions
wait(inform($\tilde{F}_j^x, \tilde{F}_i^x, a$))	// \tilde{F}_i^x is waiting for information on possible attraction
IF ($a == \tilde{F}_i^{x+1}$) THEN	// \tilde{F}_i^x is attracted by the function holon \tilde{F}_j^x
$_h \leftarrow true$	// \tilde{F}_i^x is included in a function holon of higher level
(S6') $A_i \leftarrow \tilde{F}_j^x$	// attractor of \tilde{F}_i^x is the function holon \tilde{F}_j^x
ELSE $_h \leftarrow false$	// \tilde{F}_i^x is never included in a holon of higher level
(S8) lower(CAF)	// lower the current CAF (CAF = CAF - ϵ , $\epsilon = [0, CAF]$)
(S9) } END WHILE	
} END DO	

5 Application

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

$$C_O = \eta_{IO} \frac{1}{k_{IO}} C_I \quad (5.1)$$

where :

$$- k_{IO} = \frac{\omega_I}{\omega_O} \quad (5.2)$$

- η_{IO} is the mechanical efficiency of the gearbox.

Most gearboxes are reversible. In this case, the input function can be assigned to shaft O, and the output is assigned to shaft I. The mechanism is then 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. Dividing the input power P_I the gearbox leads to the satisfaction of this requirement. It can be written as:

$$P_O = \eta_1 \frac{P_I}{n} + \dots + \eta_i \frac{P_I}{n} + \dots + \eta_n \frac{P_I}{n} \quad (5.3)$$

- P_I : input power of the gearbox
- P_O : output power of the gearbox
- η_i : the mechanical efficiency of each division of gearbox

$$P_O = \eta_{IO} P_I \quad (5.4)$$

with :

$$\eta_{IO} = \frac{1}{n} (\eta_1 + \dots + \eta_i + \dots + \eta_n) \quad (5.5)$$

The functional architecture of a parallel gearbox, with the input of power I (Input) and the output of power O (Output), is represented in Figure 1. It shows a parallel gearbox responding to the functional architecture for $n = 3$. The input is represented by shaft 1, and the output is represented by shaft 18. The gears $S_3, S_4, S_7, S_8, S_9, S_{12}, S_{13}$ and S_{16} are used to transmit the power from the input of shaft 1 to the output shaft 2. The gears are assembled on shafts S_1, S_5, S_{10}, S_{14} and S_{18} . The pivot connections S_2, S_6, S_{11}, S_{15} and S_{17} between shafts and frame S_{19} guide the rotation of the shafts.

5.1 Formation of holonic modules

The set of fuzzy functions agents and their corresponding fuzzy solutions agents are found from the Customization of the Product Service Systems in the sub-network fuzzy function agent– fuzzy solution agent in the platform *FAPIC* [53]. Figure 2 shows the fuzzy relationship between fuzzy function and fuzzy solution agents.

Then, from the attractor agent recognition, the fuzzy function and fuzzy solution agents interact to form the holonic fuzzy module agents (Figure 3). The different steps of holonic fuzzy module agents are shown in Table 4.

Three levels of holonic fuzzy module agents emerged during the formation of fuzzy holonic modules. For instance, in the first level, five *complete* holonic fuzzy module agents

$\tilde{M}_1^1, \tilde{M}_2^1, \tilde{M}_3^1, \tilde{M}_4^1, \tilde{M}_5^1$ are recognized. The first fuzzy holonic module agent \tilde{M}_1^1 is composed by holonic function agent \tilde{F}_1^1 and the fuzzy holonic agent solution \tilde{S}_1^1 , related by the fuzzy sub-relationship \tilde{R}_1^1 . The emerged fuzzy function holon agents are shown in Figure 6. Similarly, the emerged fuzzy solution holon agents are shown in Figure 5.

Table 4 Different steps of *module holon agent* formation

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

The holarchy of fuzzy holonic module network with in-between interaction and final self-organization of fuzzy function holon agents, fuzzy solution holon agents, and fuzzy module holon agents, are shown in Figure 6. In these figures, the coloured 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 intelligent module holon depend on the dynamics of conflict between agent attractors (see Appendix).

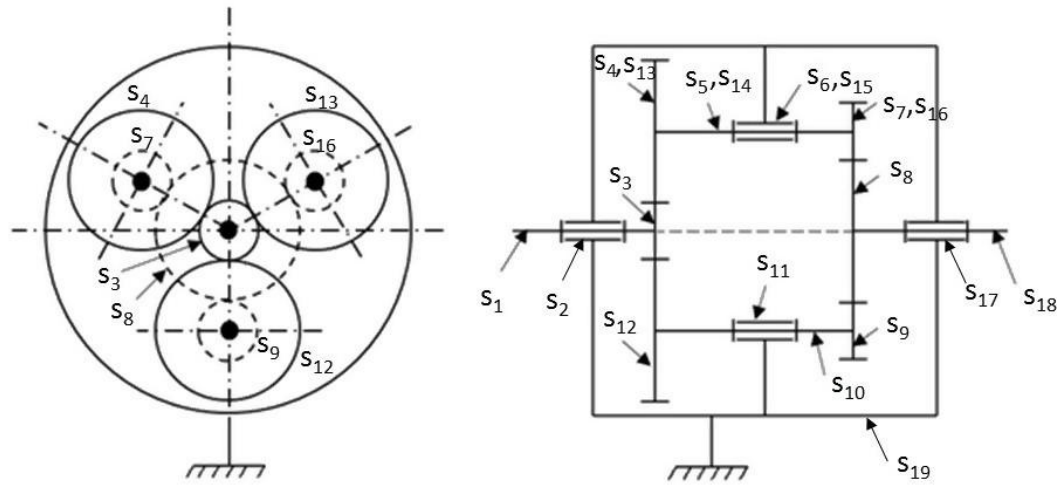
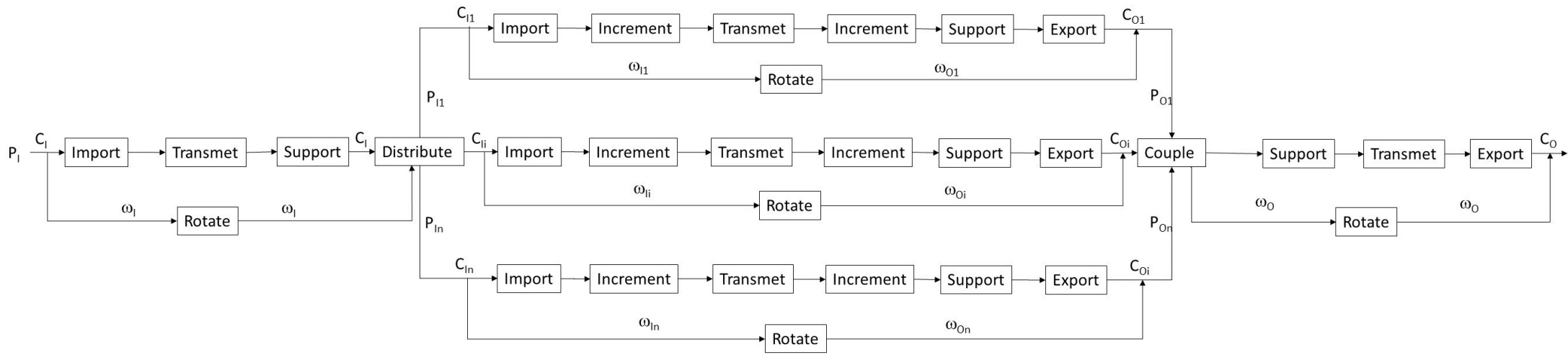


Figure 1: Functional Architecture and conceptual solution (n=3)

Export Co	\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_0	\tilde{f}_{31}	0	0	0	0	0	0	0,8	0	0	0	0	0	0	0	0	0	0,9	0,9	0,8

Figure 2 Fuzzy Relationship between fuzzy function agents and fuzzy solution agents

Level 3			\tilde{S}_1^3																			
Level 2		\tilde{S}_1^2															\tilde{S}_2^2					
Level 1	\tilde{S}_1^1					\tilde{S}_2^1				\tilde{S}_3^1				\tilde{S}_4^1		\tilde{S}_5^1						
	\tilde{s}_4	\tilde{s}_5	\tilde{s}_6	\tilde{s}_7	\tilde{s}_{19}	\tilde{s}_9	\tilde{s}_{10}	\tilde{s}_{11}	\tilde{s}_{12}	\tilde{s}_{13}	\tilde{s}_{14}	\tilde{s}_{15}	\tilde{s}_{16}	\tilde{s}_1	\tilde{s}_2	\tilde{s}_3	\tilde{s}_8	\tilde{s}_{17}	\tilde{s}_{18}			
\tilde{H}_1^3	\tilde{H}_1^1	\tilde{f}_6	0,9	0,8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
		\tilde{f}_7	0,9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		\tilde{f}_8	0,9	0,9	0	0,8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		\tilde{f}_9	0	0	0	0,9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		\tilde{f}_{10}	0,8	0,8	0,8	0,8	0,8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		\tilde{f}_{11}	0	0,8	0	0,9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		\tilde{f}_{12}	0,8	0,9	0,9	0,8	0,8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	
		\tilde{H}_2^1	\tilde{f}_{13}	0	0	0	0	0	0,8	0	0,9	0	0	0	0	0	0	0	0	0	0	
			\tilde{f}_{14}	0	0	0	0	0	0	0	0,9	0	0	0	0	0	0	0	0	0	0	
			\tilde{f}_{15}	0	0	0	0	0	0,8	0,9	0	0,9	0	0	0	0	0	0	0	0	0	
			\tilde{f}_{16}	0	0	0	0	0	0,9	0	0	0	0	0	0	0	0	0	0	0	0	
			\tilde{f}_{17}	0	0	0	0	0,8	0,8	0,8	0,8	0	0	0	0	0	0	0	0	0	0	
	\tilde{f}_{18}		0	0	0	0	0	0,9	0,8	0	0	0	0	0	0	0	0	0	0	0		
	\tilde{f}_{19}		0	0	0	0	0,8	0,8	0,9	0,8	0	0	0	0	0	0	0	0	0	0		
	\tilde{H}_3^1	\tilde{f}_{20}	0	0	0	0	0	0	0	0	0,9	0,8	0	0	0	0	0	0	0	0		
		\tilde{f}_{21}	0	0	0	0	0	0	0	0	0,9	0	0	0	0	0	0	0	0	0		
		\tilde{f}_{22}	0	0	0	0	0	0	0	0	0,9	0,9	0	0,8	0	0	0	0	0	0		
		\tilde{f}_{23}	0	0	0	0	0	0	0	0	0	0,8	0,8	0,9	0	0	0	0	0	0		
		\tilde{f}_{24}	0	0	0	0	0,8	0	0	0	0	0,8	0	0	0,8	0	0	0	0	0		
		\tilde{f}_{25}	0	0	0	0	0	0	0	0	0	0,8	0	0,9	0	0	0	0	0	0		
		\tilde{f}_{26}	0	0	0	0	0,8	0	0	0	0	0,8	0,9	0,9	0,8	0	0	0	0	0		
		\tilde{H}_4^1	\tilde{f}_1	0	0	0	0	0	0	0	0	0	0	0	0	0,9	0	0	0	0	0	
	\tilde{f}_2		0	0	0	0	0	0	0	0	0	0	0	0	0,9	0	0	0	0	0		
	\tilde{f}_3		0	0	0	0	0,8	0	0	0	0	0	0	0	0,9	0,8	0	0	0	0		
	\tilde{f}_4		0	0	0	0	0,8	0	0	0	0	0	0	0	0,9	0,9	0,8	0	0	0		
	\tilde{H}_6^1	\tilde{f}_5	0,9	0	0	0	0	0	0	0,9	0,9	0	0	0	0,4	0	0,9	0	0	0		
	\tilde{H}_2^2	\tilde{H}_5^1	\tilde{f}_{28}	0	0	0	0	0,8	0	0	0	0	0	0	0	0	0	0	0	0,8	0,9	
			\tilde{f}_{29}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0,9	
			\tilde{f}_{30}	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0,9
			\tilde{f}_{31}	0	0	0	0	0,8	0	0	0	0	0	0	0	0	0	0	0	0,8	0,9	0,9
	\tilde{H}_3^2	\tilde{H}_7^1	\tilde{f}_{27}	0	0	0	0,9	0	0,9	0	0	0	0	0,9	0	0	0	0,9	0	0,4		

Figure 3 Fuzzy module holon agents

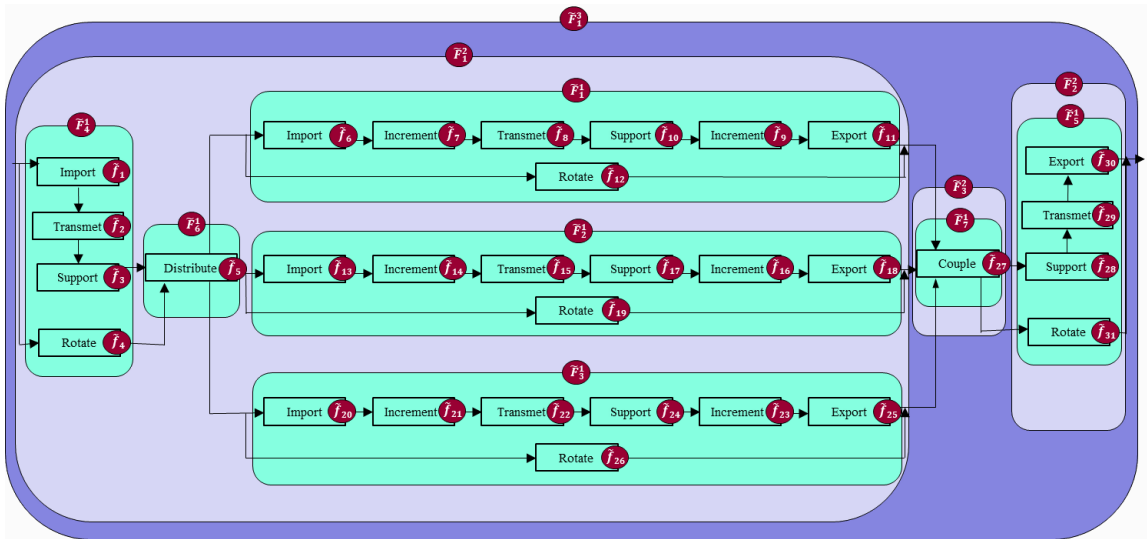


Figure 4 Fuzzy function holon agents

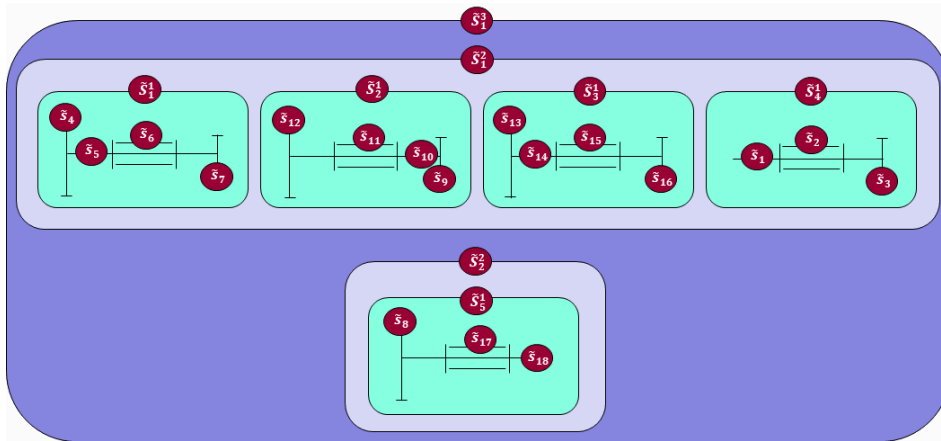


Figure 5 Fuzzy solution holon agents

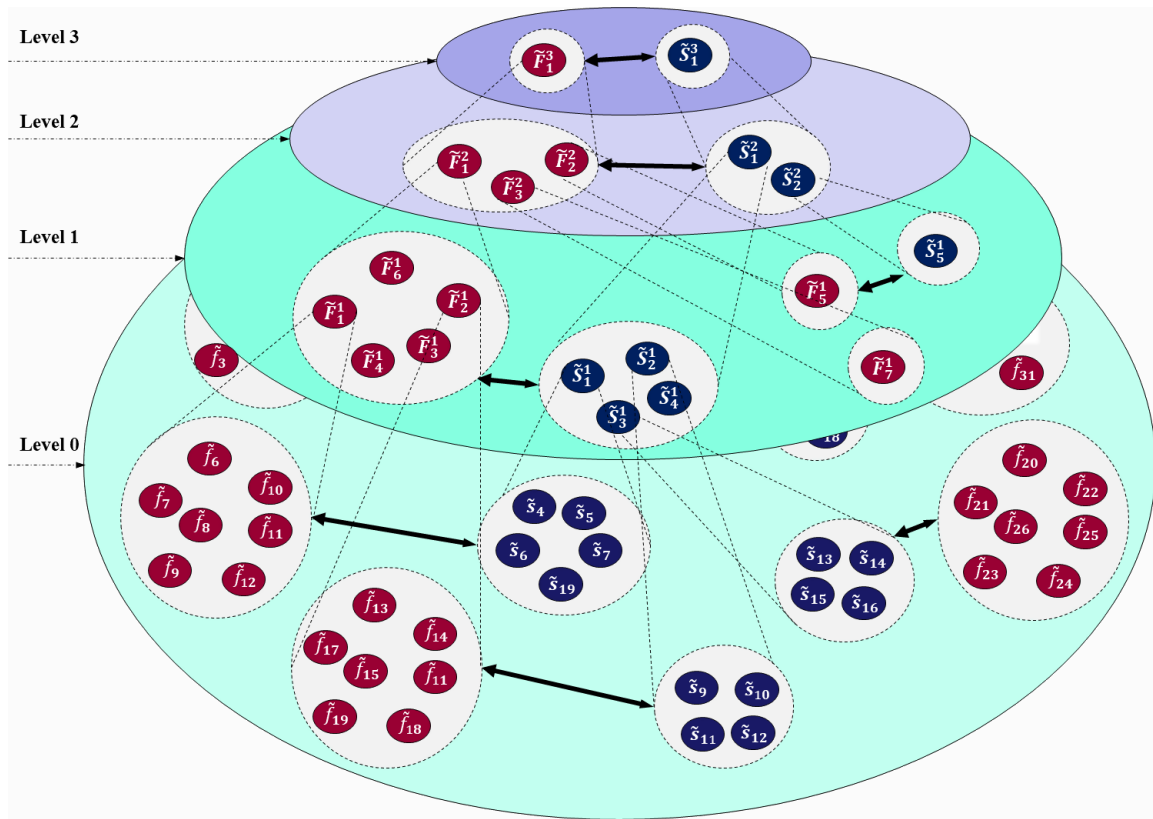


Figure 6: Hierarchy of fuzzy module holon agents

6 Discussion

The proposed model for intelligent modular design uses concepts such as holon, attractor, and uncertainty. The formation of multi-scale modules is facilitated by holon and the attractor. This multi-scale formation is adaptable, unlike traditional module formation.

Formation of the communication network by the holonic fuzzy agents removes the distinction between continuous-discontinuous formations of traditional modules. Holonic structure's fundamental property is also followed by communication. This leads to the formation of *holarchy of modules*, a nested hierarchical order. Holarchy is a hierarchy of fuzzy module holons.

The module formation problem is modernized with the use of holarchic model. The embedding of fuzzy module holon in whole fuzzy module holons get influenced by the whole and also influences it. The fuzzy module holon though complete in itself is part of whole fuzzy module holons as well as made of many decomposed fuzzy module holons. 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. 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 the knowledge of product design and knowledge of PSS. However, the formation of the module is not a straight forward problem and embattled with unpredictable changes and sudden transformations. The dynamic of intelligent module holon depend on the dynamics of conflict between agent attractors.

7 Conclusion

The modular design technologies current trend is to use a combination of technologies such as 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 systems development, therefore, demands the use of concurrent and holistic engineering approaches. Using these approaches, the design of intelligent self-sustainable models and products is feasible.

Using the concepts of intelligent holons and integrating the fuzziness in the modelling of holonic fuzzy agents, this study is focused on the modular design. It is suggested that modules, as well as their models, should be designed holistically, as adaptive objects.

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, dynamicity and agility, to the holonic structure, and thus, its adaptability.

Further developments include the study of the dynamic behaviour of intelligent modules in digitalized design network via the analysis of the creation or destruction of attractors identified and defined in holonic fuzzy module agents.

8 Acknowledgements

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

9 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. Issa H, Ostrosi E, Lenczner M, Habib R (2017) Fuzzy holons for intelligent multi-scale design in cloud-based design for configurations. *J Intell Manuf* 28:1219–1247
5. Ong SK, Xu QL, Nee AYC (2008) *Design Reuse in Product Development Modeling, Analysis and Optimization*, World Scientific Publishing. Singapore
6. Fogliatto FS, Silveira GJ da (eds) (2011) *Mass customization: engineering and managing global operations*. Springer, London
7. Stone RB, Wood KL (2000) Development of a functional basis for design. *J Mech Des* 122:359–370
8. Suh NP (2001) *Axiomatic Design: Advances and Applications*, 1 edition. Oxford University Press, New York
9. 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

10. Lange MW, Imsdahl A (2014) Modular Function Deployment: Using Module Drivers to Impart Strategies to a Product Architecture. In: Simpson TW, Jiao J (Roger), Siddique Z, Hölttä-Otto K (eds) *Adv. Prod. Fam. Prod. Platf. Des. Methods Appl.* Springer New York, New York, NY, pp 91–118
11. Fischer JO (2008) *Kostenbewusstes Konstruieren: Praxisbewährte Methoden und Informationssysteme für den Konstruktionsprozess.* Springer-Verlag
12. Schuh G, Arnoscht J, Aleksic S (2012) Systematische Gestaltung von Kommunalitäten in Produkten und Prozessen. *ZWF Z Für Wirtsch Fabr* 107:322–326
13. Ostrosi E, Stjepandić J, Fukuda S, Kurth M (2014) Modularity: New Trends for Product Platform Strategy Support in Concurrent Engineering. *Adv Transdiscipl Eng* 414–423
14. Zhuo L, Yoke San W, Kim Seng L (2008) Integrated approach to modularize the conceptual product family architecture. *Int J Adv Manuf Technol* 36:83–96
15. 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
16. 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
17. Stjepandić J, Ostrosi E, Fougères A-J, Kurth M (2015) Modularity and Supporting Tools and Methods. In: Stjepandić J, Wognum N, J.C. Verhagen W (eds) *Concurr. Eng. 21st Century Found. Dev. Chall.* Springer International Publishing, Cham, pp 389–420
18. Bruun HPL, Mortensen NH, Harlou U, Wörösch M, Proschowsky M (2015) PLM system support for modular product development. *Comput Ind* 67:97–111
19. Sun J, Chai N, Pi G, Zhang Z, Fan B (2017) Modularization of Product Service System Based on Functional Requirement. *Procedia CIRP* 64:301–305
20. Tinnilä M, Lipponen M, Rajahonka M, Bask A (2010) The concept of modularity: diffusion from manufacturing to service production. *J Manuf Technol Manag* 21:355–375
21. Jung M (2005) *Controlling modularer Produktfamilien in der Automobilindustrie.* Deutscher Universitätsverlag, Wiesbaden
22. Piller FT, Tseng MM (2010) *Handbook of Research in Mass Customization and Personalization.* World Scientific
23. Custódio DT, Roehe Vaccaro GL, Nunes FL, Vidor G, Chiwiacowsky LD (2018) Variant product configuration of industrial air handling units in a MTO environment. *Int J Adv Manuf Technol* 95:1025–1037
24. Kreng VB, Lee T-P (2004) Modular product design with grouping genetic algorithm—a case study. *Comput Ind Eng* 46:443–460
25. 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

26. Yu S, Yang Q, Tao J, Tian X, Yin F (2011) Product modular design incorporating life cycle issues - Group Genetic Algorithm (GGA) based method. *J Clean Prod* 19:1016–1032
27. 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
28. Pandremenos J, Chryssolouris G (2011) A neural network approach for the development of modular product architectures. *Int J Comput Integr Manuf* 24:879–887
29. Fougères A-J, Ostrosi E (2013) Fuzzy agent-based approach for consensual design synthesis in product configuration. *Integr Comput-Aided Eng* 20:259–274
30. Albrecht SV, Stone P (2018) Autonomous agents modelling other agents: A comprehensive survey and open problems. *Artif Intell* 258:66–95
31. Valckenaers P, Van Brussel H, Hadeli, Bochmann O, Saint Germain B, Zamfirescu C (2003) On the design of emergent systems: an investigation of integration and interoperability issues. *Eng Appl Artif Intell* 16:377–393
32. Deciu ER, Ostrosi E, Ferney M, Gheorghe M (2005) Configurable product design using multiple fuzzy models. *J Eng Des* 16:209–233
33. He B, Pan Q, Deng Z (2018) Product carbon footprint for product life cycle under uncertainty. *J Clean Prod* 187:459–472
34. Ostrosi E, Fougères A-J (2011) Optimization of product configuration assisted by fuzzy agents. *Int J Interact Des Manuf IJIDeM* 5:29–44
35. Ostrosi E, Haxhij L, Ferney M (2008) Configuration Grammars: Powerful Tools for Product Modelling in CAD Systems. In: Curran R, Chou S-Y, Trappey A (eds) *Collab. Prod. Serv. Life Cycle Manag. Sustain. World*. Springer London, pp 469–481
36. Ostrosi E, Fougères A-J, Ferney M, Klein D (2012) A fuzzy configuration multi-agent approach for product family modelling in conceptual design. *J Intell Manuf* 23:2565–2586
37. Huang C-C (2004) A Multi-agent Approach to Collaborative Design of Modular Products. *Concurr Eng* 12:39–47
38. Issa H, Ostrosi E, Lenczner M, Habib R (2013) Influence of Functional Knowledge Structuring for Modular Design. *Adv Mater Res* 651:595–600
39. Overmeyer L, Bentlage A (2014) Small-Scaled Modular Design for Aircraft Wings. In: Denkena B (ed) *New Prod. Technol. Aerosp. Ind.* Springer International Publishing, pp 55–62
40. Zhang Z, Chai N, Ostrosi E, Shang Y (2019) Extraction of association rules in the schematic design of product service system based on Pareto-MODGDFA. *Comput Ind Eng* 129:392–403

41. 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
42. Krstic D (2004) Computing with analyzed shapes. In: *Des. Comput. Cogn.* Springer, pp 397–416
43. Valckenaers P, Van Brussel H (2005) Fundamental insights into holonic systems design. In: *Int. Conf. Ind. Appl. Holonic Multi-Agent Syst.* Springer, pp 11–22
44. Valckenaers P, Van Brussel H, Wyns J, Bongaerts L, Peeters P (1998) Designing holonic manufacturing systems. *Robot Comput-Integr Manuf* 14:455–464
45. Van Brussel H, Bongaerts L, Wyns J, Valckenaers P, Van Ginderachter T (1999) A conceptual framework for holonic manufacturing: Identification of manufacturing holons. *J Manuf Syst* 18:35
46. Van Brussel H, Wyns J, Valckenaers P, Bongaerts L, Peeters P (1998) Reference architecture for holonic manufacturing systems: PROSA. *Comput Ind* 37:255–274
47. Leitão P, Restivo F (2008) A holonic approach to dynamic manufacturing scheduling. *Robot Comput-Integr Manuf* 24:625–634
48. Leitão P, Restivo F (2006) ADACOR: A holonic architecture for agile and adaptive manufacturing control. *Comput Ind* 57:121–130
49. Arai T, Aiyama Y, Sugi M, Ota J (2001) Holonic assembly system with Plug and Produce. *Comput Ind* 46:289–299
50. Hsieh F-S (2008) Robustness analysis of holonic assembly/disassembly processes with Petri nets. *Automatica* 44:2538–2548
51. Hashemian M (2005) Design for adaptability. PhD Thesis, University of Saskatchewan, Saskatoon, Saskatchewan. Canada.
52. San Miguel M, Johnson JH, Kertesz J, Kaski K, Díaz-Guilera A, MacKay RS, Loreto V, Érdi P, Helbing D (2012) Challenges in complex systems science. *Eur Phys J Spec Top* 214:245–271
53. Ostrosi E, Fougères A-J, Ferney M (2012) Fuzzy agents for product configuration in collaborative and distributed design process. *Appl Soft Comput* 12:2091–2105

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  0.00  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  0.80  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  0.90  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  0.80  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  0.80  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  0.00  0.00  0.00  0.00  0.00  0.00  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  0.90  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  0.00  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
0.80  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  0.80  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  0.90  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  0.00
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  0.00  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  0.00  0.00  0.00  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  0.90  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  0.00  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  0.00  0.80
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 = {f12, {f6, f7, f8, f9, f10, f11, f12}}
F1.2 = {f19, {f13, f14, f15, f16, f17, f18, f19}}
F1.3 = {f26, {f20, f21, f22, f23, f24, f25, f26}}
F1.4 = {f4, {f1, f2, f3, f4}}
F1.5 = {f31, {f28, f29, f30, f31}}
F1.6 = {f5, {f5}}

```

```

F1.7 = {f27, {f27}}

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

#Fuzzy Module Holon / Strong Attractor
(M1.1 / f12)
(M1.2 / f19)
(M1.3 / f26)
(M1.4 / f4)
(M1.5 / f31)

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

#Fuzzy Solution - Fuzzy Function Matrix Holon
16.00  1.60  1.60  1.60  1.60  0.90  0.90
0.00   14.40  0.00  0.00  0.00  0.90  0.90
0.00   0.00  14.40  0.00  0.00  0.90  0.90
0.00   0.00  0.00  6.10  0.00  1.30  0.00
0.00   0.00  0.00  0.00  6.10  0.00  1.30

#Fuzzy Function Holons
F2.1 = {F1.2, {F1.1, F1.2}}
F2.2 = {F1.3, {F1.3}}
F2.3 = {F1.4, {F1.4}}
F2.4 = {F1.5, {F1.5}}
F2.5 = {F1.6, {F1.6}}
F2.6 = {F1.7, {F1.7}}

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

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

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

#Fuzzy Solution - Fuzzy Function Matrix Holon
32.00  1.60  1.60  1.60  1.80  1.80
0.00   14.40  0.00  0.00  0.90  0.90
0.00   0.00  6.10  0.00  1.30  0.00
0.00   0.00  0.00  6.10  0.00  1.30

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

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

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

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

```

```
#####  
#Fuzzy Solution - Fuzzy Function Matrix Holon  
22.70  33.60  7.70  
1.30   6.10   0.00  
  
#Fuzzy Function Holons  
F4.1 = {F3.2, {F3.1, F3.2, F3.3}}  
  
#Fuzzy Solution Holons  
S4.1 = {S3.1, S3.2}  
  
#Fuzzy Module Holon / Strong Attractor  
(M4.1 / F3.2)  
  
#####  
# 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}  
  
#Fuzzy Module Holon / Strong Attractor  
(M5.1 / F4.1)
```