Model-based System Reconfiguration: a descriptive study of current industrial challenges

Lara Qasim, Marija Jankovic, Sorin Olaru, Jean-Luc Garnier

Model-based System Reconfiguration: a descriptive study of current industrial challenges

Lara QASIM, Marija JANKOVIC, Sorin OLARU, Jean-Luc GARNIER

Abstract System Reconfiguration is essential in management of complex systems because it allows companies better flexibility and adaptability. System evolutions have to be managed in order to ensure system effectivity and efficiency through its whole lifecycle, in particular when it comes to complex systems that have decades of development and up to hundreds of years of usage. System Reconfiguration can be considered and deployed in different lifecycle phases. Two significant phases are considered for configuration management and System Reconfiguration: design-time –allowing system performances by modifying the architecture in early stages– and run-time – allowing optimization of performances during the in-service operations. This paper gives an overview of a field research being initiated and currently undergoing to capture the strengths and the shortages in the current industrial landscape. It also discusses possible future management strategies with regard to identified issues and challenges.

Lara QASIM
Thales Technical Directorate
1 avenue Augustin Fresnel – 91767 Palaiseau Cedex
Laboratoire Génie Industriel, CentraleSupelec
3 rue Joliot Curie, 91190 Gif-sur-Yvette
Lara.Qasim@thalesgroup.com
Lara.Qasim@centralesupelec.fr

Marija JANKOVIC
Laboratoire Génie Industriel, CentraleSupelec
3 rue Joliot Curie, 91190 Gif-sur-Yvette
Marija.Jankovic@ecp.fr

Sorin OLARU
Laboratoire de Signaux et Systemes, CentraleSupelec
3 rue Joliot Curie, 91190 Gif-sur-Yvette
Sorin.Olaru@centralesupelec.fr

Jean-Luc GARNIER
Thales Technical Directorate
1 avenue Augustin Fresnel – 91767 Palaiseau Cedex
Jean-Luc.Garnier@thalesgroup.com

1 Introduction

In today's competitive market, companies are concerned with developing systems effectively while reducing cost and time overruns. The primary objective of Systems Engineering is to develop systems that are operational regarding defined contexts and environments. Systems Engineering sustains complex systems activities with the aim to satisfy internal and external stakeholders requirements [1,2]. Configuration management, technical management and life cycle model management are formalized as a set of processes during design-time and run-time for management of systems through their lifecycles [1,2,3].

"System configuration" is defined in Systems Engineering as a set of elements that compose a system in terms of hardware devices, software, interfaces, human profiles and processes [2]. System configuration is one of the important aspects addressed by the system management. The objective of system configuration management is to ensure effective management of an evolving system during its lifecycle [2]. System configuration can be characterized with regards to economic, environmental, legal, operational, behavioral, structural, and social aspects that are necessary to demonstrate a capability. As a counterpart, "System Reconfiguration" is defined in this paper by subsequent changes of the system configurations with the objective of maintaining or improving the capabilities provided by the system. System Reconfiguration is in particular necessary in two major system life-cycle phases: development (or "design-time") and in-service phase (or "run-time").

At design-time, one can identify several reasons for configuration changes. Systems may evolve to improve the performance by taking into account information coming from operational data. Changes are introduced to system configurations in order to correct errors, and mismatches during the development, testing, and deployment of the system. Stakeholders' requirements for system evolution may also drive to changes.

At run-time, the objective is to optimize system performances according to the context or the mission. Configuration optimization in terms of capabilities and available resources is needed to cope with environment or mission evolutions. In case of starvation of resources, the end-user needs must be dealt with by optimizing the remaining resources.

This paper aims at presenting current challenges that are related to system configuration based on industry research. Section 2 gives the state of the art of the domains related to System Reconfiguration. Section 3 describes the methodology used in this research. Section 4 presents the industrial challenges and the issues. Section 5 analyzes and discusses the industrial observation, and also gives an insight into future works. Finally, section 6 draws conclusions.

2 State of the art

To keep the system functioning correctly, companies need to manage evolutions of system configurations. System Reconfiguration has been treated in several research domains with different levels of maturity. The concept of system configuration has been initially developed in the fault detection, isolation, and reconfiguration related studies. Reconfiguration in fault tolerant control (FTC) is mainly addressing the dysfunction of interconnected dynamic systems that are more or less complex (see section 2.1 addressing both considerations). These two considerations shall be addressed in Systems Engineering and during configuration management at run-time (see section 2.2). Configuration management at design-time can be yielding from different development activities with consideration of system configuration and evolution in the case of change management and propagation (see section 2.3).

2.1 Fault detection, isolation and reconfiguration

In operations, it is important to supervise and control component or sub-system operations, for example via a feedback loop, to maintain the system's desired behavior. Once a fault in one component has been identified within a supervision activity, the controller must react by reconfiguring the system to cope with these abnormal behaviors. Literature is addressing these concerns as fault detection, isolation, and reconfiguration or FTC. The primary purpose of FTC functionalities is to overcome the malfunctions while maintaining desirable stability and performance properties [4,5].

Passive and active FTC functionalities exist, depending on their management of detected faults. Passive FTC functionalities are robust control activities that handle faults within a predefined quality of service. On the other hand, active FTC functionalities react to a detected fault and perform reconfiguration so that the stability and the performances can be maintained [6]. In active FTC functionalities, the fact that the controller is reconfigurable means that one can adaptively address non-predefined faults.

A typical active FTC functionality relies on two fundamental mechanisms: fault detection and isolation (FDI) sometimes referred to as "fault diagnosis" [7], and reconfiguration control mechanisms (RC) [4]. The reconfiguration control aims at masking the fault either by switching to a redundant system/component or by revising the controller structure. In some cases, the available resources do not allow counteracting fault effects. In such cases, the best solution is to allow system degradation when the performance is accepted to be out of the optimal area [5].

There are different techniques used in fault detection and isolation. They are classified into model-based and data-based techniques [4]. Model-based techniques use system models to estimate the system states and parameters. Data-driven techniques, on the other hand, rely on classifiers and signal processing [4]. In this paper, the interest lies in changes and deviations in the system state addressed by model-based techniques while data-driven techniques fall out of interest.

Reiter [8], in his theory of diagnosis, proposes a method that requires a model describing the system. Given observations of a system, diagnosis compares the observed system with the expected behavior (modelled system) to determine the malfunctioning components. Reiter's theory has been extended to deal with the model-based diagnosis of different kinds of systems in different domains of applications [9-10]. Identifying faults in malfunctioning systems is important but repairing these systems so that they can continue their missions is an essential problem to be addressed. Reiter's theory of model-based diagnosis has been extended to a theory of reconfiguration [11]. Much research has been conducted to use the model-based analysis concepts in the reconfiguration control design and analysis algorithms [12-15].

2.2 *Configuration Management and system adaptability*

Adaptability can be understood by the fact that systems have to face continuously evolving situations and must be able to reconfigure their structure or their behavior to maximize their ability to accomplish required functions [16].

Moreover, research efforts have been conducted to design flexible and agile systems supported by a reconfiguration agent. Boardman and Sauser [17] define agility as the ability of the system to quickly detect and destroy unintended behaviors. In Systems Engineering, flexibility means the ability of the system to respond effectively and efficiently to potential internal or external changes [18]. AlSafi [19] proposes an approach based on reconfiguration software agents that allows manufacturing systems to adapt to changes in the manufacturing requirements and the environment by generating an alternative of a new feasible configuration. An approach of designing reconfigurable systems using a multi-agent system is described in [20]. A model-based oriented approach that supports the adaptation processes based on a run-time transformation of the system architecture is proposed in [21].

2.3 Change prediction and propagation in the conceptual design and basic engineering phase

Most design activities can be considered as modifications of previous designs. New product or system development is then an incremental process involving integrating modifications (changes) to existing designs where innovation and ideas are only used in some parts of the products while other parts remain relatively unchanged [22]. Literature underlines several strategies addressing system configuration and reconfiguration at design time including system modularity, reusability (Components of the shelf - COTS), system platform design and change management.

As products or systems are often based on past designs rather than designing from scratch, change management and change propagation are very important to manage the development of complex systems. Giffin [23] defines change propagation as the process by which a change to one part or element of an existing system configuration or design results in one or more additional changes to the system when those changes would not have otherwise been required.

To manage the risk of changes early in the design process, there is a need to assess which systems are most likely to be affected by a change, and what the impact of such a change would be. Clarkson [24] proposes a change prediction method to calculate the risk of direct and indirect change propagation. In [25], an efficient engineering change management (ECM) is presented as a key enabler for the agile product development of physical products.

After analysis of the academic state of the art, no research has currently been found on integrated approach aiming at describing reconfiguration process and actions for both design-time and run-time; even if such a king of approach should provide a global strategy to enhance flexibility, efficiency, reusability, modularity, and configurability.

3 Methodology

The research presented in this paper is action-based research [26]. This means that at least one of the authors is also an engineer in industry. This paper is addressing the first stages of this research. The research methodology (Fig. 1) is based on the exploration of current literature through the examination of papers supported by data collection. According to Blessing and Chakrabarti [27], observation and data gathering are essential to analyze and understand the industrial context and to propose a descriptive study that covers both empirical studies and their analysis to form new hypotheses.

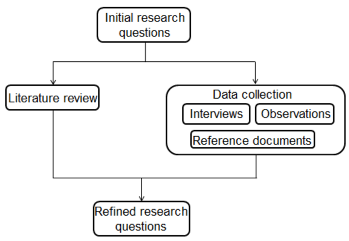

Fig. 1 Research methodology

In the first phase of this action-based research, the aim is identifying the current challenges in management of system configurations and overall System Reconfiguration process with regard to existing literature. This research is based on triangulation of the data: interviews, direct observations and company reference documents. To understand where challenges, limitations, and opportunities lie, interviewing can be used to support engineering design research [28]. To ensure objectivity, the interview has been designed according to a structured list of questions. Objective of the interviews is to find out the definitions related to system management (including system configuration and System Reconfiguration), in terms of artefacts and process that governs system or product life cycle activities, like Systems Engineering, manufacturing and in-service operations. Moreover, questions about the different methods and tools used in the configuration management and System Reconfiguration processes have been included in our survey.

For the interviews, 17 Thales experts have been identified with different levels of involvement in system management. Since Thales deals with different types of systems in various operational contexts, the identified persons are classified into two categories: 1) people working in transversal activities and 2) subject matter experts. Currently, interviewed persons belong to the first category. The results of these interviews are presented in section 4.

4 Analysis of current industrial challenges

The objective of this research is to propose integrated model-based support for instantiation of system configurations and System Reconfiguration addressing both design-time and run-time. In this context, a field study is conducted to identify existing data, process, issues and challenges in order to better understand the type of needed support.

With a systemic approach, System Reconfiguration is considered as a mean to improve the performance and the quality of service to be provided by the system. An essential benefit of System Reconfiguration is to reduce the redundancy that comes along with different problems like increased cost, space limitations, weight, and high energy consumption. To allow system management, including instantiation of system configurations and reconfiguration, interviewed experts have been underlying the importance of concurrent element management, i.e., resources, functions, (Fig. 2) in different lifecycle phases of safety and security-critical systems. Reconfiguration allows improving systems in terms of performance and effectiveness to ensure an increased availability and a continuity of service.

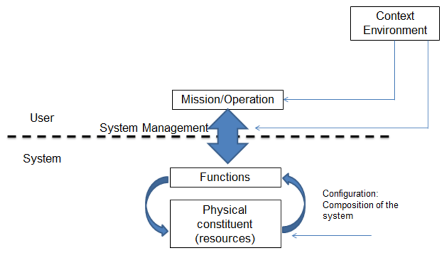

Fig. 2: Optimization of the concurrent elements (resources, functions)

Experts interviewed during this research have emphasized the need to support systems evolution during each lifecycle phase. According to them, “Reconfiguration is an everyday question.” For instance, during dismantling one should think of reconfiguration because dismantling a service or a product might have an impact on the overall services provided by the system. However, as for the perimeter considered in the enterprise, 2 phases of systems lifecycle seem to be critical: design and operations (Fig. 3).

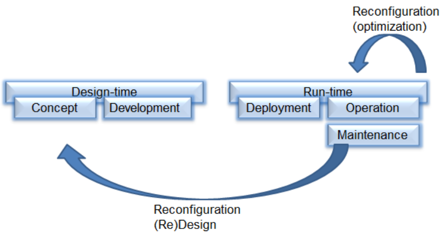

Fig. 3: Reconfiguration in different lifecycle stages

At design-time, a system has to be studied in terms of functions permitting to accomplish specific operational capabilities. Reconfiguration at design-time means optimizing the implemented resources to achieve capabilities demanded by the customer. At deployment, before starting the mission, instantiation of a configuration aims at ensuring that the total resources provide the functions needed to accomplish the mission.

At run-time, the functions and resources essential for the mission are monitored as the reconfiguration process relies mainly on the awareness of the system state concerning the health of available resources with regards to the functional modes to be guaranteed. When a function becomes unavailable because of a faulty resource, reconfiguration aims to pick up the capability from other available resources (P1 to P2 in the example of Fig. 4). If the function is no more accessible, then reconfiguration aims at steering the system into a degraded mode (P3 to P4 in the example of Fig. 4).

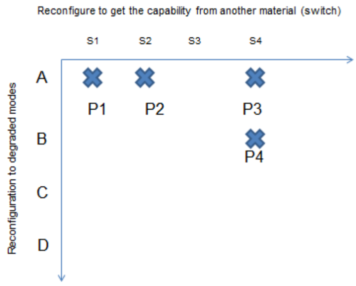

Fig. 4: Reconfiguration during operation to optimize the resources

Configuration management and System Reconfiguration are needed at both design-time and run-time. These processes rely on different **data** and **models** to be collected, reused, generated and managed during the whole relevant system or product life cycle, according to formal and unformal **contracts** committed with stakeholders. Some challenges pertaining to configuration management and reconfiguration have been identified: challenges related to data, modeling issues, contracting and certification, system and context taxonomy. The following sub-sections discuss these issues.

4.1 Data related challenges

The instantiation of configurations and reconfiguration processes rely fundamentally on data. Data necessary for instantiation of configurations and reconfiguration need to be collected and verified from day zero up to the retirement. Indeed, these data can originate from different phases of system or product lifecycle and can have different structure, support, and management systems. This sub-section discusses the issues related to data.

a) Data availability and accessibility

Data availability and accessibility can be a real issue at the technical or operational levels. For instance, in some applications, data collection cannot be possible due to harsh working environments; for example, fuel rod temperature measurement within a nuclear reactor in operation. In other cases, measured data cannot be transmitted directly; therefore communication technologies are needed to give access to these data. However, when communicating data, one should take into account all the measures to secure these data. When dealing with operational data, privacy, integrity and confidentiality become a real issue. Consequently, secured data processing for strategic and tactical applications (e.g. military systems) may become an essential requirement.

b) Data shared across stakeholders

Different stakeholders are involved in complex systems, such as: system designers, developers, customers and end-users. In addition to that, a specific team can be responsible for the system at each of the system lifecycle stages. Therefore, it could be difficult to collect data shared between the different stakeholders because of complex organizational interfaces and intellectual property.

c) Data storage

The quantity of data needed for instantiation of configurations and reconfiguration can be considerably depending on data saving strategy. In that case, data storage for instantiation of configurations and reconfiguration is also being considered as a critical issue. To avoid storing data continuously, front-end pre-processing can be implemented as a way to lessen data storage. However, pre-processing relies on detection of thresholds and hence the problem of threshold definition can arise.

d) Uncertainty and data verification

Data necessary in the instantiation of configurations and reconfiguration process can have different sources. System technical data collected from sensors installed either independently from the system or in an embedded manner. External observation of the system; from operators or maintenance teams, is also considered a source of data. Indeed, the uncertainty about the collected data is variable. Data verification is necessary to address this uncertainty. However, data verification is not evident because it is linked at least to the level of knowledge and completeness of the data.

e) Data combination

Instantiation of configurations and reconfiguration rely on a priori and a posteriori data. These data need to be combined to allow instantiation of configurations and reconfiguration at both design-time and run-time. These data can be of different nature and are collected in one phase and later used in other phase of the life cycle. For example, during design-time, data from past systems can be used to modify the system design. At run-time, data from maintenance or the current operational situation can lead to system reorganization. Data combination is not trivial and requires in-depth data analysis to consider the degrees of uncertainty.

4.2 Modeling issues

Different types of models can be used for the instantiation of configurations and reconfiguration. For example, depending on the system and its context, models can be continuous or discrete. Systems are built of constituent sub-systems leading to nested modeling. In instantiation of configurations and reconfiguration, different levels of systems are involved: the system of interest (the system fulfilling the operational mission); but also enabling systems for development, manufacturing, maintenance, health monitoring, supervision and control. In order to achieve instantiation of configurations and reconfiguration, multi-level modeling is needed to combine and conjointly manage these levels. In addition to that, data to be modeled can be of different nature: internal (technical), external (environmental and operational). There is a difficulty in modeling these data. The modeling and analysis techniques need to be adapted to address different types of data.

4.3 Contracting and certification

In the industry, developers and solution providers are usually concerned with contracting and certification. Contracts include information about usage profiles, configuration alternatives, operational contexts, quality of service, reliability, availability, safety, security, etc. The contracted configurations are tested, validated and certified in advance. However, during operations, when considering dynamic operation, the new alternatives are instantiated during operations; therefore they are not initially approved. The efforts needed to cope with this situation are not negligible because of lack of metrics required for the certification process. This activity may last for a long time leading to penalties due to a schedule overrun. Consequently, challenges related to certifying, assessing and selecting the newly emerging configurations have to be taken into account.

4.4 System and context taxonomy

The Thales Group develops a very large set of system and product types. Most of the time, these systems and products do not stand alone and are integrated into different operational contexts (e.g., platform or infrastructure) and larger systems (or systems of systems). Consequently, the concerns related to these systems are extremely different. For example, reconfiguring a closed system, assembly or a platform; is needed to propose new configurations while integrating new technologies. When considering distributed System Reconfiguration, there is a need to address the problem of connectivity between the system elements. Moreover, in a system of systems (SoS), constituent systems are mostly independent and this can lead to emerging effects. In addition to that, the interfaces between these constituent systems of a SoS may evolve. Consequently, the SoS reconfiguration must be considered as an agile capability. The methods and mechanisms for reconfiguration might be slightly different according to the system types. Indeed, this variation leads to complexity when trying to build the support framework progressively with the aim of overall generalization across systems and industry. A holistic method for reconfiguration is an essential challenge because it must be as abstract as possible to adapt to system and context taxonomies.

5 Discussion and future work

This research seeks to build on previous work and aims at proposing an integrated approach for model-based instantiation of system configuration and reconfiguration addressing both design-time and run-time. In particular, the approach aims at using models for representing system configuration and health monitoring system to harness complexity. The system management framework will be studied in order to propose a configuration manager (Fig. 5) including an engine and a knowledge base. This knowledge base will contain models representing the system configurations and reconfiguration rules. The configuration engine will be in charge of applying the relevant configuration.

As foreseen, instantiation of configurations and reconfiguration problems are at the intersection of different domains. Indeed, the knowledge base needs to include models related to technical, contractual and operational domains. When events occur, the engine part of the configuration manager generates reconfiguration actions. This process is possible by referring to policies included in the knowledge base as models.

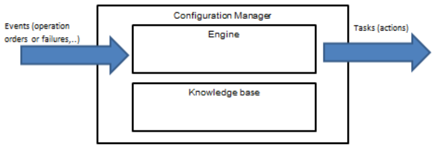

Fig. 5: Configuration manager implementation

Future work should concentrate on identifying models and data to be included in the knowledge base. Models and data can be clustered in two main categories: 1) models and data allowing reconfiguration (giving the system configurations with states and modes), 2) models and data for performance of reconfiguration (including rules and conditions).

This research seeks to address action models representing the mechanisms of reconfiguration in both phases: design and operation. Methods from complexity science sound promising to address reconfiguration in the design phase. Change prediction method can be used to predict how changes propagate in the system linkage model (Design structure matrix DSM) while addressing the risks related to propagation to allow evaluation of possible alternatives and redesign. The Model Predictive Control (MPC) can be used to address the on-line (run-time) optimization of systems. This method relies on enriching values for process input (feedback from operation), and the control action is determined as a result of the online optimization problem. Future work should focus on demonstrating the utility and underlying limitations of such methods. With regard to the challenges identified in section 4, the application of the proposed approach within industry requires careful consideration.

A better understanding of the reconfiguration process and the characteristics of related data and mechanisms are necessary to improve the proposed approach.

In next steps of the research, interviews will be completed with case studies in order to capture the specifics related to system and context taxonomy. Attention is required to identify the mechanisms of reconfiguration for both design-time and run-time. This allows addressing life cycle processes in an overall manner with the goal of generalization across systems and Industry.

6 Conclusion

System configuration and reconfiguration are essential as they support system evolutions to ensure affectivity and efficiency of systems through their life cycles. This paper identifies current industrial challenges related to system configuration and reconfiguration at design-time (development) and run-time (in-service phase). This paper discusses identified challenges –with regard to the literature review and interviews of experts– and gives initial proposals to address the reconfiguration problem. Some difficulties such as modeling issues and system taxonomy have been identified. An initial reflection on configuration manager is presented. Future work addressing change prediction (CPM) and model predictive control (MPC) methods have been discussed allowing for definition of further actions.

References

1. ISO/IEC/IEEE/15288: Systems and software engineering- System life cycle processes. (2015).
2. INCOSE(2015). Systems engineering handbook: A guide for system life cycle processes and activities. (4th ed.) D. D. Walden, G. J. Roedler, K. Forsberg, R. D. Hamelin, & T. M. Shortell (Eds.). San Diego, CA: International Council on Systems Engineering. Published by John Wiley & Sons, Inc.
3. NASA, NASA Systems Engineering Handbook, vol. 6105, no. June. (2007).
4. Y. Zhang and J. Jiang, “Bibliographical review on reconfigurable fault-tolerant control systems,” Annu. Rev. Control, vol. 32, no. 2, pp. 229–252, (2008).
5. F. Stoican and S. Olaru, Set-Theoretic Fault Detection and Control Design for Multisensor Systems. (2013).
6. J. Eterno, J. Weiss, D. Looze, and A. Willsky, “Design issues for fault tolerant-restructurable aircraft control,” in 1985 24th IEEE Conference on Decision and Control, (1985), pp. 900–905.
7. R. Isermann, “Supervision, fault-detection and fault-diagnosis methods - An introduction,” in Control Engineering Practice, (1997), vol. 5, no. 5, pp. 639–652.
8. R. Reiter, “A theory of diagnosis from first principles,” Artif. Intell., vol. 32, no. 1, pp. 57–95, (1987).
9. F. Kuntz, S. Gaudan, C. Sannino, É. Laurent, A. Griffault, and G. Point, “Model-based diagnosis for avionics systems using minimal cuts.,” in Martin Sachenbacher and Oskar Dressler and Michael Hofbaur. DX 2011, Oct 2011, Murnau, Germany, (2011), pp. 138–145.
10. H. T. Ng, “Model-Based, Multiple Fault Diagnosis of Time-Varying, Continuous Physical Devices,” in Proceedings 6th Conference on A. I. Applications, (1990), pp. 9–15.
11. J. Crow and J. Rushby, “Model-Based Reconfiguration: Toward an Integration with Diagnosis,” in In Proceedings of AAAI-91, (1991), pp. 836–841.
12. G. Provan and Y.-L. Chen, “Model-based diagnosis and control reconfiguration for discrete event systems: an integrated approach,” in Proceedings of the 38th IEEE Conference on Decision and Control, (1999), vol. 2, pp. 1762–1768.
13. K. J. Russell and R. P. Broadwater, “Model-based automated reconfiguration for fault isolation and restoration,” in 2012 IEEE PES Innovative Smart Grid Technologies (ISGT), (2012), pp. 1–4.
14. Y. Cui, J. Shi, and Z. Wang, “Backward Reconfiguration Management for Modular Avionic Reconfigurable Systems,” IEEE Syst. J., vol. 12, no. 1, pp. 137–148, (Mar. 2018).
15. S. Shan and Z. Hou, “Neural Network NARMAX Model Based Unmanned Aircraft Control Surface Reconfiguration,” in 2016 9th International Symposium on Computational Intelligence and Design (ISCID), (2016), no. 2, pp. 154–157.
16. M. Ludwig and N. Farcet, “Evaluating Enterprise Architectures through Executable Models,” in In: proceedings of the 15th international command and control research and technology symposium, (2010).
17. J. Boardman and B. Sauser, “System of Systems - the meaning of of,” in 2006 IEEE/SMC International Conference on System of Systems Engineering, (2006), no. April, pp. 118–123.
18. R. Nilchiani and D. E. Hastings, “Measuring the value of flexibility in space

systems: A six-element framework," Syst. Eng., vol. 10, no. 1, pp. 26–44, (2007).

19. Y. Alsafi and V. Vyatkin, "Ontology-based reconfiguration agent for intelligent mechatronic systems in flexible manufacturing," Robot. Comput. Integr. Manuf., vol. 26, no. 4, pp. 381–391, (2010).
20. D. Regulin, D. Schutz, T. Aicher, and B. Vogel-Heuser, "Model based design of knowledge bases in multi agent systems for enabling automatic reconfiguration capabilities of material flow modules," in IEEE International Conference on Automation Science and Engineering, (2016), vol. 2016–Novem, pp. 133–140.
21. I. B. Rodriguez, K. Drira, C. Chassot, and M. Jmaiel, "A model-based multi-level architectural reconfiguration applied to adaptability management in context-aware cooperative communication support systems," in 2009 Joint Working IEEE/IFIP Conference on Software Architecture and European Conference on Software Architecture, WICSA/ECSA 2009, (2009), pp. 353–356.
22. K. Otto and K. L. Wood, "Product design: techniques in reverse engineering and new product development," no. September 2014, (2001).
23. M. Giffin, O. L. de Weck, G. Bounova, R. Keller, C. Eckert, and P. J. Clarkson, "Change Propagation Analysis in Complex Technical Systems," ASME J Mech Des., vol. 131, no. August, pp. 1–14, (2009).
24. P. J. Clarkson, C. Simons, and C. Eckert, "Predicting Change Propagation in Complex Design," J. Mech. Des., vol. 126, no. 5, p. 788, (Jun. 2004).
25. G. Schuh, M. Riesener, and S. Breunig, "Design for Changeability: Incorporating Change Propagation Analysis in Modular Product Platform Design," in Procedia CIRP, (2017), vol. 61, pp. 63–68.
26. S. Ottosson, "Participation action research-," Technovation, vol. 23, no. 2, pp. 87–94, (2003).
27. L. T. M. Blessing and A. Chakrabarti, DRM, a Design Research Methodology, vol. 1. (2009).
28. J. D. Summers and C. M. Eckert, "Design Research Methods: Interviewing. Portland, Oregan, USA: Workshop in ASME conference. 2013.," (2013).