An integrated framework for design, management and operation of reconfigurable assembly systems

Massimo Manzini, Johannes Unglert, Dávid Gyulai, Marcello Colledani, Juan Manuel Jauregui-Becker, László Monostori, Marcello Urgo

PII: S0305-0483(16)30965-3
DOI: 10.1016/j.omega.2017.08.008
Reference: OME 1814

To appear in: Omega

Received date: 16 December 2016
Revised date: 1 August 2017
Accepted date: 9 August 2017

Please cite this article as: Massimo Manzini, Johannes Unglert, Dávid Gyulai, Marcello Colledani, Juan Manuel Jauregui-Becker, László Monostori, Marcello Urgo, An integrated framework for design, management and operation of reconfigurable assembly systems, Omega (2017), doi: 10.1016/j.omega.2017.08.008

This is a PDF file of an unedited manuscript that has been accepted for publication. As a service to our customers we are providing this early version of the manuscript. The manuscript will undergo copyediting, typesetting, and review of the resulting proof before it is published in its final form. Please note that during the production process errors may be discovered which could affect the content, and all legal disclaimers that apply to the journal pertain.
Highlights

- The research faces the increasing of product varieties, reduction of volumes and reduction of product life cycle trends.

- The paper proposes an integrated approach for the configuration and reconfiguration of an assembly cell considering several assembly technologies.

- The approach considers configuration of cellular systems, layout optimization, production planning evaluation and robust reconfiguration.

- An automotive industrial case is provided; it grounds on car spare parts assembly process.
An integrated framework for design, management and operation of reconfigurable assembly systems

Massimo Manzini\textsuperscript{a,}\textsuperscript{*}, Johannes Unglert\textsuperscript{b}, Dávid Gyulai\textsuperscript{c,d}, Marcello Colledani\textsuperscript{a}, Juan Manuel Jauregui-Becker\textsuperscript{b}, László Monostori\textsuperscript{c,d}, Marcello Urgo\textsuperscript{a}

\textsuperscript{a}Politecnico di Milano, Department of Mechanical Engineering, Milan, Italy
\textsuperscript{b}University of Twente, Department of Design, Production and Management, Enschede, The Netherlands
\textsuperscript{c}Fraunhofer Project Center PMI, Institute for Computer Science and Control, Hungarian Academy of Sciences, Budapest, Hungary
\textsuperscript{d}Budapest University of Technology and Economics, Department of Manufacturing Science and Engineering, Budapest, Hungary

Abstract

Manufacturing has to cope with the continuously increasing variety of products, change of volumes and shortening product life cycles. These trends also affect the automotive sector: the frequent introduction of new models, materials and assembly technologies put the suppliers of make-to-order parts under pressure. In this context, the design of assembly systems and their management are of paramount importance for the companies’ competitiveness. In this paper, we propose an approach for the design and reconfiguration of modular assembly systems through the integration of different computational tools addressing the design of the system, the optimization of the layout, the planning of reconfiguration actions as well as production planning. Integrating these computational tools and iterating through the resulting workflow and feedback allow to consider the outcomes and dependencies of alternative decision sequences holistically with the objective of an effective and efficient approach to production system design and management. The viability of the approach is demonstrated through the application to an automotive case study.

Keywords: reconfiguration, manufacturing, design, layout, assembly, uncertain market

1. Introduction and motivation

Throughout the last decade, manufacturing industry has been confronted with an increasing variety of products and the consequent reduction of production volumes, together with the continuous shortening of products’ life cycle (ElMaraghy et al., 2013). In this context, the design of manufacturing systems becomes a complex task that entails manufacturing strategy decisions, has long-term impacts and involves a major commitment of financial resources (Terkaj et al., 2009b). Hence, manufacturing systems must be able to smoothly and rapidly adapt to the fast evolving market dynamics.

Different system paradigms have been proposed to efficiently and effectively adapt to the market dynamics, e.g., flexibility (Hallgren and Olhager, 2009) and reconfigurability (Koren et al., 1999; ElMaraghy, 2005). These paradigms implement specific technological features such as modularity and changeability, to enable modifications of the production systems in response to the needs of the market (Wiendahl et al., 2007). Moreover, the concept of co-evolution of products, processes and production systems has been identified as a key factor in companies to manage strategically and operationally the propagation of engineering changes, and to gain competitive advantage from the resulting market and regulatory dynamics (Tolio et al., 2010; Terkaj et al., 2009a).

In this paper, we focus the attention on tier-one automotive suppliers of car-body assemblies. Original Equipment Manufacturers (OEMs) typically rely on this class of suppliers to produce spare parts for the aftermarket. Nevertheless, as OEMs are moving towards an increasing variety of models, suppliers are also involved in the production of parts during the ramp-up of new models, as well as in complementing the OEMs production capacity for high-volume car models to help managing demand peaks. Besides the high variety of models characterizing these market segments, the automotive industry is also experiencing a continuous technological evolution. To remain competitive, the suppliers have to match this evolution in the design and man-
agement of their production systems, also providing the capability to integrate new technologies into the system. In addition, suppliers have very limited degrees of freedom, since OEMs decide about the design of the products, the associated assembly technologies and, in many cases, also the equipment that must be used. Hence, the only strategic factors the suppliers can exploit are the design and management of their production systems to quickly adapt to the OEMs requirements. Consequently, suitable design and operation policies must be applied to ensure that the high variety of products with low, medium and high demand can be constantly satisfied.

As a response to these challenges, we propose an integrated design approach for batch assembly systems, organized in a cellular layout. The variety of products, assembly technologies and processes to be considered in the design of such systems results in a broad range of potential system designs. The proposed integrated approach aims to help managing this complexity through a holistic approach to design evaluation. The approach integrates four computational tools to support (i) the definition of the system’s configuration, (ii) the selection of the cell’s detailed layout and assembly process’ options, (iii) the production planning and (iv) the reconfiguration steps that have to be taken at certain moments. The four complementary tools aim to support the design of assembly systems and their managing policies. Using them in an integrated way enables to increase the level of details incrementally, and gain additional knowledge about the system. Moreover, feedback loops are implemented between the computational tools to improve the design or manage possible infeasibility. The integration of the decision-support tools aims at providing a robust solution able to cope with the co-evolution of the system together with the products and the production technologies. In this fashion, the configuration, layout planning and reconfiguration of the system consider long-term decisions, while the planning of production is based on the short-term horizon.

The paper is organized as follows: Section 2 describes the state-of-the-art in the research areas related to the integrated approach, Section 3 addresses the overall formulation and notation, Section 4 provides the structure of the integrated design approach while Section 5 presents the computational tools and their interplay in detail. Section 6 illustrates the application to a case representative for the automotive industry, while Section 7 provides a summary of the approach and future research directions.

2. Design and management frameworks: state-of-the-art

The approach we present in this paper is based on the integration of different computational tools to support a wide range of design and planning decisions throughout the life cycle of production systems. As highlighted by Ivanov et al. (2015) and Battaia et al. (2014), integrated decision support methods can offer significant benefits over isolated ones to solve complex problems that cannot be handled by a single model. Hence, the approach allows to determine and connect solutions for the emerging sub-problems on different levels of detail to avoid sub-optimal decisions. Due to the modular implementation of the proposed approach, the state-of-the-art analysis considers literature dealing with system configuration, layout, management and integration of these aspects.

Cellular manufacturing systems as means to achieve manufacturing flexibility has been a subject for research already for a long time (Selim et al., 1998). Even though advances are documented in more recent publications such as (Papaioannou and Wilson, 2010), a number of challenges in the field are still present today. One aspect that has been identified as vital for successfully applying the cellular concept is the consideration of its dynamics in design models, as described by Goldengorin et al. (2013). The authors conclude that, since the product mix changes over time, also the cell’s layout must be adjusted periodically to obtain systems, which are robust with regard to a changing product mix, or dynamic, realizing smooth changes of the system’s structure.

Hu et al. (2011) and Koren and Shpitalni (2010) suggest to combine the layout and production planning of systems to match the system structure with the customers’ demands. Nevertheless, they argue that this topic so far received little attention by researchers. Li et al. (2011) argue that the throughput of the system is usually determined by considering the bottleneck process only, without considering the applied production sequence. Moreover, setups and changeovers seem to be rarely considered during the design phase of the line: Nazarian et al. (2010), Boysen et al. (2007) and Battini et al. (2011) expose that the link with production planning and the resulting actual batch sizes and changeovers appears to be rather loose. An integrated methodology focusing on the automotive assembly process is presented by Ceglarek et al. (2015), where the authors consider the configuration of a remote laser welding assembly line together with the production process and task sequencing. The approach focuses on the design and high level performance evaluation, without
Matta et al. (2007) describe an approach to design reconfigurable systems, estimating the system’s ramp-up performances and also considering the reconfiguration option to increase or decrease the system’s capacity. In addition, the authors generate a robust solution by applying a Markov decision process to consider multiple time periods. An approach that takes into account the design of multi-product flexible transfer lines and its reconfiguration is presented by Tolio and Urgo (2013). In particular, the configuration phase consists of assigning operations and equipment components to selected stations, while during the reconfiguration phase, the system’s equipment components are rearranged to match the changed requirements.

Considering the main design and management aspects in the scope of this paper, an interesting work is presented by Hu et al. (2011), where multiple approaches to designing assembly systems are reviewed and summarized, taking into account reconfigurability, flexibility and co-evolution aspects. Even though all the cited approaches cope with the overall configuration and management problem by integrating support methods on different levels, the information transferred between the models mostly flows in one direction. Seizing the opportunity to review and improve higher-level decisions based on the more detailed solutions resulting from lower-level support approaches is not considered.

Delorme et al. (2014) state that assembly line configuration problem is a widely studied, relevant industrial topic involving a set of various optimization sub-problems. Guschinskaya et al. (2008) consider the configuration problem for assembly systems without buffers by using a heuristic multi-step approach. They face the problem of grouping operations in several stations and minimizing the total equipment cost. They propose partitioning the layout design process into several steps and introducing technology constraints and precedence relations sequentially. All manufacturing operations are considered with a fixed assignment of the machine tools to spindles and without taking into account alternative machining processes. The extension of this work is presented by Maksoud et al. (2014), where also reconfiguration actions are considered. In both cases, the volumes to be produced and also the processing times are treated as deterministic parameters. A similar configuration problem is faced in (Guschinskaya et al., 2011) by using a three-step genetic approach. Other approaches focus on the robust design (Papakostas et al., 2014), balancing (Chica et al., 2016) and scheduling (Koca et al., 2015) of assembly lines, considering the flexibility of the resources, and uncertainty of the key parameters. Another stream of research is represented by Heilala et al. (2006) where an approach combining design and simulation for modular assembly systems is presented, it is focused on minimizing the total cost of ownership. This approach tries to cope with the overall management problem by considering a specific class of assembly systems, but it does not offer the opportunity to choose between alternative technology solutions in terms of type of equipment and task sequence. An approach that considers different equipment alternatives is presented by Michalos et al. (2015). The design approach consists of two phases: first, the equipment components are selected for the system and, subsequently, arranged to form the assembly line configuration using an optimization algorithm. Alternative criteria for the optimization are taken into account considering the fixed demand on medium-long time period of years.

The above described literature shows a representative part of the vast number of contributions to support assembly system configuration, reconfiguration and management striving for well-performing solutions suitable for the respective problems. In the last few years, however, new reconfigurable technologies and equipment became available to help manufacturers to face the issues described in the introduction. These new system paradigms require novel approaches to support design, operation and management tasks. Moreover, these approaches address configuration and planning in a separate way, resulting in a sub-optimal overall set of decisions. Hence, it appears beneficial to provide integrated models that reflect the relevant decision aspects and obtain consistent solutions for the sub-problems. Some of the presented approaches suggest to combine complementary models by establishing links among them. Yet, this linkage can result in unidirectional workflows that make it difficult to refine earlier design decisions based on the knowledge derived from the later stages in the execution of the models. As a result, feedback procedures that allow to iterate between the distinct models appear a rarely considered opportunity for improving the final solutions.

Addressing these gaps was the motivation for developing the approach proposed in the following, integrating four different approaches to support the initial system configuration, layout optimization, reconfiguration planning, production planning and simulation.
3. Reconfigurable assembly line design problem formulation

The overall configuration problem is partitioned into three sub-problems. First, a suitable system configuration must be identified. This sub-problem involves determining the number of cells and the assignment of the products and production technologies, thus also defining the routing of products among the cells. Secondly, the pieces of equipment must be arranged into the assembly cell architecture to define a specific layout and, consequently, the task sequencing is performed. In this phase, also the possibility of using alternative equipment for a given operation is taken into consideration. Then, the planning of production is considered to check whether the requirements of the OEM in terms of delivery dates, can be guaranteed. Eventually, the reconfiguration problem is considered on a long time horizon, determining the most suitable actions for system reconfiguration.

Specifically, we consider the production problem in terms of a set of products $P$ to be assembled, using a set of modular equipment groups called Functional Assembly Groups (FAGs). A FAG consists of one or more pieces of equipment required for specific production processes (e.g. the power source of a welding machine) together with the machine tools and fixtures needed to accomplish a class of assembly operations, such as resistance spot welding, gluing or hemming. The processing information for the product $p \in P$ is provided in terms of technological parameters (e.g. number of weld joints, hemming or gluing length) and the associated unitary processing times. Additional non-operational data related to the FAGs (e.g. the floor space requirements, the investment costs and depreciation period) is taken into account. FAGs are modular devices, easily organizable into the general layout of the reconfigurable cells (Figure 1). Hence, they act as enabler for the changeability of the production system. The FAGs can be managed in two different ways when considered in relation to two different time horizons: in the short term, the available machine tools can be changed to cope with different parts to be assembled, which is referred to as changeover. On a longer time horizon, however, there is an opportunity to modify the set of available FAGs, i.e., acquiring new ones or dismissing available ones; we refer to this as reconfiguration as it entails the modification of the available equipment. Given the equipment associated to an FAG, the set of assembling operations can be executed in different alternative ways. These alternative options —called execution modes—are provided in the set $K$ in terms of the associated task sequencing and processing times. Further details on execution modes and the associated technological characteristics are provided in Section 5.2.

The FAGs are connected by 7-axis robots moving along a central rail to transport and handle the parts within the assembly cell. Additional stations allow the workers to load and unload the assembly cell (I/O stations). The automated processes are managed by a control unit and the cell is fenced.

In formal terms, the layout configuration is represented by a variable $z = (E, F, V)$ where $E$ defines the

<table>
<thead>
<tr>
<th>Table 1: Notation for data and variables.</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p$</td>
</tr>
<tr>
<td>$h_p$</td>
</tr>
<tr>
<td>$l_p$</td>
</tr>
<tr>
<td>$d_{p}$</td>
</tr>
<tr>
<td>$a_{p_c}$</td>
</tr>
<tr>
<td>$n_{j}$</td>
</tr>
<tr>
<td>$f_{j}$</td>
</tr>
<tr>
<td>$e_j$</td>
</tr>
<tr>
<td>$c_{e}$</td>
</tr>
<tr>
<td>$q$</td>
</tr>
<tr>
<td>$c_{h_{op}}$</td>
</tr>
<tr>
<td>$c_{i}^{\text{mp}(f,v)}$</td>
</tr>
<tr>
<td>$c_{i}^{\text{cc}(f,v)}$</td>
</tr>
<tr>
<td>$c_{i}^{\text{cc}(z_{ct})}$</td>
</tr>
<tr>
<td>$c_{i}^{\text{cc}(z_{ct})}$</td>
</tr>
<tr>
<td>$B_{p}^{\text{p}}$</td>
</tr>
<tr>
<td>$d_{p}^{\text{p}}$</td>
</tr>
<tr>
<td>$l_{j}^{\text{p}}$</td>
</tr>
</tbody>
</table>
Figure 1: Schematic 2D-layout of the reconfigurable cell architecture.

position of each FAG j on the layout, while F defines the set of machine tools that are required for the FAGs to process the different parts, and V defines a proper execution mode for each assembly activity chosen from the set K.

As described in Section 1, the proposed approach also aims at providing a reconfiguration strategy to cope with the evolution of the production requirements. To formalize this evolution, a probabilistic scenario model is used, to structure the uncertainty associated to future events. A stochastic scenario tree is defined, consisting of a set of nodes Ω over a set of time periods T. Each node ω ∈ Ω in the tree is associated to a set of production requirements, i.e., the products P(ω) to be produced and the associated volumes d_p(ω), the average lot sizes I_p(ω) and the assembly processes B_p(ω). An occurrence probability π(ω) is assigned to each node. A path starting from the root of the tree and ending in a leaf represents a specific evolution scenario with its occurrence probability.

Considering the overall time horizon of the scenario tree, the multi-cell system configuration is selected from a set of alternative configurations S. Hence, grounding on this candidate architecture, for each cell c ∈ C and node ω ∈ Ω in the scenario tree, a specific cell configuration z_c is provided (i is the time period corresponding to ω), able to cope with the requirements in node P(ω). Within the time period associated to this node, the machine tools available are changed-over every time a different part must be assembled according to the production management strategy. The operation paradigm is the multi-product assembly cell, where the production of batches of parts needs to match the customer demand and the associated delivery dates. When moving from one node to another, the cell configuration z can change, thus requiring a reconfiguration action. The overall design and reconfiguration approach of a cell aims at defining the best for each cell over the whole scenario tree considering the associated cost (investment, reconfiguration, operational, etc.).

4. Assembly system design and management framework

In response to the findings of Section 2, the decision-makers should be able to consider the dependencies and mutual effects of the various configurations and planning decisions, using the respective results as feedback to refine previous decisions. The objective is to use the feedback to guide the approach to a feasible solution considering the different sub-problems under study. Hence, the proposed integrated approach is organized according to the workflow in Figure 2 and connects four computational tools that make use of a common data structure and act in an integrated way: the Assembly System Configuration tool, the Assembly Cell Configuration tool, the Production Planning and Simulation tool and the Reconfiguration Planning tool. Starting from the set of production requirements, the Assembly System Configuration tool allows to explore the design space, compare different system configurations and to identify the most promising ones. To this end, the tool synthesizes production system configurations that are based on the various opportunities for assigning the available equipment and products to one or multiple cells, resulting in configurations that allow for a high-level performance analysis. The aim of the tool is to allow identifying advantageous system configurations matching various, user-specified requirements. These high-level designs are then handled by the Assembly Cell Configuration tool. This computational tool —grounding on the system-wide configuration— goes into the layout configuration process by arranging the equipment into a cell layout, selecting the proper task sequencing and evaluating the dynamic performances of the proposed solution. The latter is further evaluated from the management point of view by the Production Planning and Simulation tool. Taking into consideration detailed orders and delivery dates, this computational tool sequences the production batches in the assembly line over a short-term planning horizon, also taking into consideration the availability of raw materials and personnel. The performance indicators of the system are evaluated through a DES model, analyzing the system in greater detail. The three tools are designed
to work in an integrated way, i.e., feedback opportunities between the design steps are implemented, with the aim of improving the configuration design and managing possible infeasibilities. Eventually, the Reconfiguration Planning tool addresses the evolution of the assembly cell according to the evolution of the production requests modeled through scenario tree, like the bottom right of Figure 2. The aim is to provide a robust design for the assembly line, consisting of an initial configuration and a set of reconfiguration steps to match the uncertain market evolution described by the paths along the scenario tree. The whole approach grounds on a common description of the production problem to be addressed, whose notation is reported in Table 1.

5. Description of the computational tools

Each computational tool contributes with specific capabilities to the objectives of the support framework. The peculiarities of the approaches and their interplay are described in the following sections.

5.1. The Assembly System Configuration tool

A few knowledge-based approaches for automated design of production systems have been reported in literature. While the support tools described by Mellrichamp et al. (1990); Borenstein (1998); Lee et al. (2006) and Khan et al. (2011) generate single system design solutions for their respective design problems, only two approaches have been found by the authors that generate and present multiple system configurations for a given production problem, and allow the users to compare the alternatives. In these approaches, however, the comparison focuses on the performance properties of the systems (Michalos et al., 2012), or aims at configuring an individual dedicated transfer line Delorme et al. (2012). Many opportunities exist for allocating the production resources J and products P to multiple production cells and make possible a vast number of production system configurations with differing performance profiles. The resulting freedom of design will remain undiscovered by these approaches as they focus either on performance, attributing less importance to the design implications of the solutions, or on a single line, ignoring the opportunity to consider multiple lines simultaneously. Hence, the Assembly System Configuration tool aims to explore a large variety of solutions, which can consist of one or more cells and fulfill design

Figure 2: Design and management workflow for reconfigurable assembly cells, highlighting the flow of the main input and output data.

<table>
<thead>
<tr>
<th>Input from database</th>
<th>product information (P, T, Ω) resource information (J)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input before runtime</td>
<td>scaling factors (SF, SF_op) requirements for design and performances (R)</td>
</tr>
<tr>
<td>Solution approach</td>
<td>(I) user enters scaling factors SF, SF_op and R</td>
</tr>
<tr>
<td></td>
<td>(II) algorithm creates and analyzes configurations</td>
</tr>
<tr>
<td></td>
<td>(III) tool visualizes preliminary configurations</td>
</tr>
<tr>
<td></td>
<td>(IV) user assesses configurations</td>
</tr>
<tr>
<td></td>
<td>(V) user selects preferred configurations</td>
</tr>
<tr>
<td>Main output</td>
<td>feasible, preferred system configurations S</td>
</tr>
</tbody>
</table>

Table 2: Summarized information flow in the Assembly System Configuration tool.
Therefore, the Assembly System Configuration tool automatically generates many configurations of the assembly system and analyzes their key performance indicators (KPIs) to enable the exploration of the design space. This is supported by visualizing the generated system configurations and their performances in versatile interfaces, such as the one in Figure 3, where each configuration is represented by one dot in the X-Y plot. To achieve this objective, multiple steps of design synthesis and analysis are executed in an automated way for each candidate solution. At the beginning, the tool determines the number of assembly cells in the system, afterwards the production equipment is selected and assigned to the cells; as the last design step, the products are assigned to cells. Once the design is completely specified, the KPIs of the synthesized solution can be determined. All candidate configurations are visualized so that the decision-makers can explore the design properties and performances of the solutions. Based on these characteristics, the users can specify feasible regions of the design space by imposing constraints and generating new solutions, or selecting the most suitable system configurations. By iteratively specifying design or performance constraints, generating matching solutions and assessing the results, the knowledge-based tool enables to concurrently assess various options to configure the production system, and facilitates developing feasible assembly system configurations.

As basis for the generation of system designs, the system configuration algorithm (see Table 3) analyses the expected product demand and technical information of each product in $P$. Therefore, the input data contains the technical description of the production processes of the various products as well as the demand cases ($DC$) that represent the uncertain evolution of the market in terms of production volumes in the form of lowest, medium and highest expected product demand across all scenario nodes $\Omega$ in the entire time horizon of consideration $T$. Furthermore, the data provide information about the production equipment $I$ (e.g. $FAG$s, 7-axis robots) in terms of investment cost, processing speed and shop floor requirement.

Firstly, the maximum number of cells is calculated by multiplying the specified minimum number of cells by the user-controlled cell scaling factor ($SF_c$; see steps 1-2, Table 3). Afterwards, the algorithm sets the number of cells by instantiating a random value between the minimum and maximum ($LB_c$ and $UB_c$).

Secondly, the input information is used to determine the minimum capacity requirement for each process type $j$ (e.g. for the $FAG$ for mechanical joining, see step 4). Dividing these values by the available capacity for each equipment type and rounding up the result allows to obtain the discrete minimum number of required $FAG$ instances. After that, the operation scaling factor ($SF_{op}$) is applied to yield the respective upper bound for the possible number of equipment instances. This scaling factor can be controlled by the decision-maker to specify the allowed sizes of the systems and maximum capacity reserve. Again, a random value ($nFAG_j$) between the minimum and maximum numbers of instances ($LB_j$ and $UB_j$) is instantiated. By distributing the quantity of production resources for each process type randomly to the cells (step 10), the system hardware design is completed. After the system design is specified, the heuristic-based algorithm allocates the processes required to manufacture the product portfolio to the specific cells and resources (step 11-15). First the products are sorted by total demand volume; secondly, the products are sequentially allocated by finding the cells that have the required $FAG$s for producing the product. If a production step can be performed in multiple cells, the cell is chosen that yields the least increase of capacity utilization. If insufficient capacity is left on the cell, allocation is attempted on other cells. In case no options exist for allocation, the system configuration is discarded.

After allocating the products, the KPIs of the total system and its different sub-systems are calculated for each demand case $DC$, such as the total investment, required floor space, expected utilization of productions cells, logistics cost, storage cost, estimated product-related cost and lead-time. In case the parameter values are outside of the range specified in the user requirements $R$, for instance if the allocation requires more capacity than available or the system covers more area than specified by the user, the procedure checks if the maximum number of attempts has been reached. If this is the case it exits, otherwise the solution is discarded and synthesis of a new one attempted instead. In case the requirements are fulfilled, the solution is added to the preliminary set $S_{prel}$.

As output of the tool, the design and performance properties of the various generated system configurations $S_{prel}$ are visualized in interfaces that allow decision-makers to interactively assess the solutions. Due to the many opportunities for configuring the system with regard to production resources and product allocation, an enormous number of system designs can be generated by the tool. To generate preferably relevant solutions, various design strategies are implemented to result in three different system types: (i) system configurations in which user-defined product families are
**System Config-algorithm** \((P, T, \Omega, J)\)

**Input:** \(SF_c, SF_{op}, R\)
1. \(LB_c = 1\)
2. \(UB_c = SF_c\)
3. Set random value between \(LB_c\) and \(UB_c\)
4. **ForEach** operation type \(j\) in \(J\)
   5. calculate \(Cap_{\min}\) of operation type \(j\)
   6. \(LB_j = Cap_{\min}/Cap_j\)
   7. \(UB_j = LB_j \cdot SF_{op}\)
   8. \(nFAG_j = \) random value between \(LB_j\) and \(UB_j\)
   9. **ForEach** instance \(nFAG_p\) in \(nFAG_j\)
      10. assign \(nFAG_p\) to a random cell
11. **Sort** products by total demand volume
12. **ForEach** product \(p \in P\)
   13. find cells that can produce \(p\)
   14. allocate \(p\) to cell with lowest utilization increase
   15. analyze system configuration
   16. **ForEach** DC in \(T, \Omega\)
      17. analyze system performance in DC
      18. compare analysis results to \(R\)
      19. **If** \(R\) violated
         20. **If** number of attempts reached
         21. **Then** exit procedure
         22. **Else** add system configuration to set \(S_{prel}\)
23. **Output:** \(S_{prel}\)

Table 3: System configuration algorithm

Produced in separate cells; (ii) system configurations in which products are allocated to the cells without requiring transfers between cells; (iii) system configurations in which all products can be transferred between the production cells and follow distinct routes. Considering the design procedure, the difference between these architectures is the allocation requirement specified by the users. According to their preferences, each cell of the configuration needs to have sufficient processing capacity for (i) an entire product family; (ii) all production steps of the individual products; (iii) or complementary subsets of the products’ production steps. The visualization of the various design and performance properties of the system configurations in the GUI can be adjusted to suit the users’ preferences and encourages to compare the configurations, for instance by selecting the most relevant performance parameters to contrast the solutions on the Pareto-fronts of these parameters. Eventually, requirements can be formulated for all performance and design properties of solutions. By iteratively adding

<table>
<thead>
<tr>
<th>Inputs from previous tool</th>
<th>system configurations (S)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Additional inputs</strong></td>
<td>ex modes (K) demand (d_p(\omega), \forall p, \omega)</td>
</tr>
<tr>
<td></td>
<td>lot size (l_p(\omega), \forall p, \omega)</td>
</tr>
<tr>
<td></td>
<td>assembly processes (b_{ij}(\omega), \forall p, \omega)</td>
</tr>
<tr>
<td><strong>Objective and decision</strong></td>
<td>layout arrangement (Z = 1) performance evaluation (T_{eval}(Z) = \Theta())</td>
</tr>
<tr>
<td><strong>Main output</strong></td>
<td>operative time (T_{eval}(Z), \forall p, \omega)</td>
</tr>
</tbody>
</table>

Table 4: Summarized information flow in the Assembly Cell Configuration tool.

more requirements and using the algorithms to synthesize new, matching solutions, the users can narrow down the number of solutions according to their constraints and proceed with the set of the most suitable solutions \(S\).

In this way the automated design procedure and set-based presentation of the solutions aims at creating an awareness of alternative concepts for organizing the production system. Simultaneously, it stands for a novel approach to reduce the time needed for conceptualizing and analyzing a large number of alternative system configurations (Unglert et al. (2016)).

5.2. The Assembly Cell Configuration tool

After identifying a promising system configuration with the Assembly System Configuration tool, the Assembly Cell Configuration tool aims to increase the level of detail of the design in terms of the physical layout, the task sequencing and a dynamic performance evaluation using a fast analytical method, also considering failure and repair probabilities of each FAG. The Assembly Cell Configuration tool uses as inputs (i) the products and FAGs assigned to the cell from the Assembly System Configuration tool, (ii) the detailed forms of realizing the functionality of an FAG described in the execution mode set \(K\) and (iii) the production requirements coming from the scenario tree. With this information, the approach generates a set of detailed layout configurations and their task sequences considering alternative execution modes. The output of the tool is the estimated total production time for each layout in all scenario nodes (Table 4).

In particular, an execution mode, first presented by Angius et al. (2016), is a possible technically feasible arrangement of the equipment and the associated sequence of tasks to execute a set of operations on each product. Together with the definition of the layout, also the execution mode for each FAG is defined. The alternative execution modes (illustrated in Table 5) are:

1. Part worked inside the FAG. In this case a FAG has its own working cube. The part is transported
into the working cube from the input station (or from the previous machine) by a 7-axis robot with a proper handling tool. Once the part is inside the FAG, it is processed.

2. **Part blocked in the fixture while FAG works on it.** In this case, the FAG takes advantage of an external fixture to work on a part. The 7-axis robot is used to move the part to and from the fixture.

3. **Part blocked in the fixture while the 7-axis robot works on it.** In this case, the 7-axis robot operates the process on the part while it is blocked in the fixture. The 7-axis has to load a specific machine tool of the FAG (e.g., glue gun for an adhesive joining) in order to execute the operations.

4. **Part handled by the 7-axis robot while the FAG works on it.** In this case, the 7-axis robot handles the part while the equipment in the FAG execute the process.

Hence, an execution mode has to be chosen for each FAG, affecting the layout, the sequence of operations and, consequently, also the performance of the process.

In order to calculate the performance of an assembly cell configuration, we define the state of the system through a vector \( s = [c_1, c_2, \ldots, c_N, r] \) where:

- \( c_i \), \( 1 \leq i \leq N \), describes the state of the \( i \)th FAG in the cell, assuming the values: Operative (O), Starved (S) and Blocked (B).
- \( r \) describes the state of the 7-axis robot assuming

### Table 5: Execution modes description

<table>
<thead>
<tr>
<th>Execution Mode</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>R1 loads the part from the turn table (1) R1 moves to the FAG R1 releases the part in the fixture (2) R2 joins the sub-assemblies R1 loads the part R1 moves to the turn table (1) R1 releases the part in the mould</td>
</tr>
<tr>
<td>2</td>
<td>R1 loads the part from the turn table (1) R1 moves to the FAG R1 releases the part in the fixture (2) R2 joins the sub-assemblies R1 loads the part R1 moves to the turn table (1) R1 releases the part in the mould</td>
</tr>
<tr>
<td>3</td>
<td>R1 loads sub-assemblies on the fixture (1) R1 moves to the machine tool rack (2) R1 loads the needed machine tool R1 moves to the fixture (1) R1 joins the parts R1 moves to the machine tool rack (2) R1 releases the tool and loads the clamp</td>
</tr>
<tr>
<td>4</td>
<td>R1 loads the part from fixture (1) R1 moves to the FAG R2 joins the sub-assemblies while R1 holds it R1 moves to the fixture (1) R1 releases the part in fixture (1)</td>
</tr>
</tbody>
</table>
values in \([0, 1, \ldots, N, N+1]\) where: 0 is the idle state and the values \(i \in [1, \ldots, N+1]\), are operative states in which the 7-axis robot is processing a part that will be transported to the FAG \(i\). \(N+1\) is a dummy module representing the completion of the process.

In order to describe the system dynamics, the following considerations apply:

1. parts are moved only by the 7-axis robot;
2. components are always available;
3. no additional storage is possible in the cell, hence, they are blocked after the end of service until the 7-axis takes the processed part;
4. the 7-axis robot moves the parts only if the destination FAG or fixture is empty;
5. the 7-axis robot is always able to unload a part that has been processed inside the last system FAG.
6. the load and unload times are negligible;
7. all the machines of the system work asynchronously in relation to the others.

The dynamics of the assembly cell are described by logical expressions, modeling the sequence of events. We describe an event in terms of its pre-conditions and post-conditions: i) \(<\text{pre-conditions}>\) is the logical expression describing under which conditions an event can occur; ii) \(<\text{post-conditions}>\) describes how the state of the system will change if such event occurs. We denote an event with \(<\text{pre-conditions}>\rightarrow<\text{post-conditions}>\). For brevity, each logical expression will indicate only those variables that are directly involved in the event. Given a generic state \(s\), the events describing the system dynamics are the following:

\[
c_1 = O \land c_{i+1} = S \land r = 0
\]

\[
\rightarrow c_i = S \land r = i + 1, 2 \leq i \leq N
\]

\[
c_1 = O \land (c_{i+1} \neq S \lor r \neq 0)
\]

\[
\rightarrow c_i = B, 2 \leq i \leq N
\]

\[
c_1 = O \land c_2 = S \land r = 0 \rightarrow c_1 = O \land r = 2
\]

\[
c_1 = O \land (c_2 \neq S \lor r \neq 0) \rightarrow c_1 = B
\]

\[
r = k \land k > 1 \land \exists i > 1 : (c_i = B \land c_{i+1} = S)
\]

\[
\rightarrow c_k = O \land c_1 = S \land r = i
\]

\[
r = k \land k > 1 \land c_1 = B \land c_2 = S
\]

\[
\rightarrow c_k = O \land c_1 = O \land r = 2
\]

\[
r = k \land k > 1 \land \exists i : (c_i = B \land c_{i+1} = S)
\]

\[
\rightarrow c_k = O \land r = 0
\]

Events (1) and (2) model the end of a service at the \(i\)th FAG. In particular, (1) corresponds to the case in which the 7-axis robot is idle and can remove the part from FAG \(i\) immediately, while in (2) the 7-axis robot is already processing a part. In the first case \(c_i\) will be blocked whereas in the second it will be starred. Events (3) and (4) refer to the first FAG returning operative after working due to the continuous availability of raw parts. Events (5) and (6) describe the case in which the 7-axis finishes to move a part to a FAG \(k\) and takes another part from another FAG \(i\). The difference between these two events is that FAG \(i\) always returns operative when the part is removed whereas every other working cubes get starved because it must wait for a new part. Event (7) describes the 7-axis robot becoming idle when the cell does not contain any other part ready to be transported. As further constraint, we assume that the 7-axis serves the FAG closest to the end of the line amongst those that are blocked. We assume that every change of state occurs according to Markovian distributions and the underlying stochastic process is a Continuous Time Markov chain (CTMC). Therefore, in the most simple scenario, all the service times are exponential distributed.

Unfortunately, the exponential distribution is frequently a bad candidate for representing actual distributions of real systems. To manage this problem, we consider phase-type distributions (PH), to model the processing times. The use of PH distribution in manufacturing system has been introduced by Altiok (1997) and then studied by Neuts et al. (2000) and Colledani and Tolio (2011). A random variable \(T\) is PH distributed if its cumulative distribution function corresponds to the time till absorption of a CTMC given a pre-fixed initial distribution. The more detailed structure of PH distributions allows the fitting of general distributions by matching their higher moments (Horvath and Telek, 2007). Due to the introduction of PH distributions the structure of the underlying CTMC gets slightly more complicated because every state of the system is expanded in order to consider also the detailed information about the aging.
of the distributions. This lead to an infinitesimal generator, denoted by $Q$, that is composed of blocks, referred as $Q_{\omega,r}$, that describe the motion of the process between two states of the system, that can be used to calculate the time to absorption of the described Continuous Time Markov Chain and, hence, the lot completion time.

The novelty of the Assembly Cell Configuration tool is the model of the reconfigurable cell architecture, grounding on the modular FAGs as well as the consideration of alternative execution modes that affect in particular:

- the selection of pieces of equipment to be arranged in the cell, faced by layout generation function $Z = \Gamma(B_{\omega}(\omega), K, J)$;
- the assembly process task sequencing impacting the performance evaluation $T_{\text{total}}(z_{ct}) = \Theta(z_{ct}, d_{\omega}(\omega), l_{\omega}(\omega), B_{\omega}(\omega), V)$.

5.3. The Production Planning and Simulation tool

<table>
<thead>
<tr>
<th>Inputs from previous tool</th>
<th>technological parameters $(r_{pu}^{p}, t_{pu}^{p})$ resource information $(a_{pu}, c_{pu})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Additional inputs</td>
<td>contractual delivery volumes $d_{pu}$ resource pool information $n_{pu}$</td>
</tr>
<tr>
<td>Objective and decision</td>
<td>operation costs minimization</td>
</tr>
<tr>
<td>Main output</td>
<td>production lot-sizes $x_{pu}$ operation-related costs</td>
</tr>
</tbody>
</table>

Table 6: Summarized information flow in the Production Planning and Simulation tool.

In general, production planning is responsible for matching supply with demand, by balancing the internal capacities with the order stream by transforming the customer needs into production orders (Meyr et al., 2015; Karimi et al., 2003). Medium-term planning and scheduling problems—considering shared and reconfigurable resources—are solved by Li et al. (2014); Safaei and Tavakkoli-Moghaddam (2009) and by Chen et al. (2014), however, these solutions cannot be applied in the proposed workflow, due to the different system architecture (constraints) and/or objectives. The execution of plans directly incurs operative costs that need to be respected when seeking for the cost-optimal reconfiguration strategy. Therefore, the production planning tool of the workflow aims at predicting the future-expected operation related costs, characterizing a given cell configuration besides the forecast order stream. The proposed production planning method is able to handle the reconfigurable cells by applying constraints on the usage of exchangeable FAGs, preventing to hurt capacity limitations and thus result in feasible plans. Besides the planning, the second major part of the Production Planning and Simulation tool is a novel discrete event simulation (DES) model, implemented to execute the calculated plans by adding realistic random events (e.g. machine breakdowns) and representing the possible random-nature of the production parameters, as they might have additional impact on the operational level performace and cost factors (Ponsignon and Mönch, 2014). The novelty of the DES is rooted in the model structure and building procedure: as the cells have fixed parts and also some changeable equipment, the models reflect the real physical architecture and operation of the cells by having static model elements, as well as dynamically-created blocks. Besides the hardware elements of the cell not supposed to undergo any modification, the static part of the model includes statistics related objects, as well as a central model controller. The latter is responsible for managing the setup events and controlling the capacities (human and machine), by communicating with the modular, dynamically created cell blocks via status and trigger signals (Gyulai et al., 2016).
calculating the operational costs in question. The decision variables determine the production lots $x_{puc}$, specifying
the volume of product $p$ assembled in cell $c$ in period $u$ over a discrete time horizon $U$. Assembled products can be either delivered to the customer ($s_{pu}$) or kept in the inventory ($i_{pu}$), however, the latter is associated with inventory costs. Besides the assignment of production lots and machine capacities, an important decision is to
determine the headcount of operators $h_{cu}$ working at cell $c$ in period $u$. The production planning problem is for-
mulated by a mixed integer linear programming model
by (8)-(21).
\[
\text{minimize } \sum_{p \in P} \sum_{u \in U} \left( c_b^y b_{pu} + c^y y_{pu} \right) + \sum_{c \in C} \sum_{u \in U} c^h h_{cu} \tag{8}
\]
subject to
\[
\sum_{c \in C} \sum_{p \in P} r_{jp} y_{puc} \leq n_j \quad \forall u, j \tag{9}
\]
\[
\sum_{p \in P} \left( d^x_{pu} x_{puc} + t^y_{pu} y_{puc} \right) \leq t^h h_{cu} \quad \forall c, u \tag{10}
\]
\[
\sum_{p \in P} \left( d^m_{pu} x_{puc} + t^w_{pu} w_{puc} \right) \leq t^p \quad \forall c, u \tag{11}
\]
\[
x_{puc} \geq d_{pu} \quad \forall p, u \tag{12}
\]
\[
\sum_{c \in C} y_{puc} \leq 1 \quad \forall u, c \tag{13}
\]
\[
x_{puc} = \Lambda y_{puc} \quad \forall p, u, c \quad \Lambda > \max_{p \in P} \left( \max_{c \in C} h_{cu} \right) \tag{14}
\]
\[
x_{puc} \geq y_{puc} \quad \forall p, u, c \tag{15}
\]
\[
y_{puc} \leq d_{pu} \quad \forall p, u, c \tag{16}
\]
\[
w_{puc} \leq y_{puc} \quad \forall p, u, c \tag{17}
\]
\[
y_{puc} = y_{puc} - y_{puc-1} \quad \forall p, u, c \tag{18}
\]
\[
w_{puc} + \sum_{q \in Q} \left( y_{pu} - r_{qu} \right) \leq 1 - y_{p,u-1,c} \quad \forall p, u, c \tag{19}
\]
\[
l_{pu} - b_{pu} = l_{pu-1} - d_{p,u-1} - s_{pu} + \sum_{c \in C} x_{puc} \quad \forall p, u \tag{20}
\]
\[
0 \leq l_{pu}, s_{pu}, l_{pu}, b_{pu} \in Z^+ \tag{21}
\]

The objective of production planning is to minimize the
operative costs, consisting of the sum of backlog, in-
ventory holding and operator costs (8). The constraints express
the FAG requirements (9), human (10) and ma-
chine (11) capacity requirements. The contractual vol-
umes need to be delivered on time (12), capacity and in-
ventory shortage occur backlogs. Constraints (13)-(16)
express the products produced, while (17)-(19) specify
the setup requirements when the production of product
$p$ is started in cell $c$. The balance equation (20) is
responsible for linking the subsequent time periods with
each other through the delivery, inventory and produc-
tion volumes. The integrity conditions are defined by
(21).

Similarly to the other tools of the workflow, the main
contribution of the Production Planning and Simulation
tool is the capability of coping with the peculiar modular-
and reconfigurable cell architecture described. The
system-specific constraints of the mathematical model
control the resource consumption by combining the use
of fixed $(C)$ and exchangeable $(J)$ resources in the pro-
duction plan. Moreover, the applied DES model also
applies a novel model building procedure and simula-
tion approach to represent the system operation in a re-
alistic way, with the static-built central model controller
and the dynamically created technological blocks.

5.4. The Reconfiguration Planning tool

<table>
<thead>
<tr>
<th>Inputs from previous tool</th>
<th>set of cell configurations and their layout $Z$</th>
<th>total production time $T_{pro}(z_{ct})$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Additional inputs</td>
<td>$\text{FAG }$ purchasing cost $c^{\text{FAG}}(j,v)$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>machine tool purchasing cost $c^{\text{M}}(f,v)$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>hourly operative cost $c^{h}$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>demand $d_f(j,\omega), v, p, q$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>lot size $l_{p}(\omega), v, p, w$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>occurrence probability $\pi(\omega), v_0$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>relations between $v_0 \in \Omega$</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Main output</th>
<th>minimum expected cost</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>minimum-cost configuration evolution $z_{ct}, \forall t$</td>
</tr>
</tbody>
</table>

Table 7: Summarized information flow in the Reconfiguration Planning tool.

The Reconfiguration Planning tool addresses the layout
arrangement process of a single cell in a multi-
period time horizon. As described in Section 3, we as-
sume that the market evolution can be modeled through a
scenario tree in which each node represents a set of
requirements related to product mix and volumes. To
match the evolution of the requirements over time, an
assembly cell needs to be suitably reconfigured. Re-
configuration refers to the change, insert or move one
or more pieces of equipment in the assembly cell. In the
Reconfiguration Planning tool, all possible evolu-
tions of the market requirements (i.e., a specific path
from the root of the scenario tree to a leaf) are taken
into account in the formulation of an optimization pro-
blem, to find the best configuration and reconfiguration
plan for all the different paths. The aim is achieving ro-
busness over the whole scenario tree, e.g., acquiring re-
sources and equipment in advance (proactive approach)
or waiting for the occurrence of a specific event to proceed with a proper reconfiguration (reactive approach). The reconfiguration approach takes advantage of the set of alternative system configurations $Z$ and the estimated total production time $T_{\text{total}}(s_{\omega})$ coming from the previous step. In addition, it considers market evolution and its features, as summarized in Table 7. The optimization problem is:

$$
\begin{align*}
\min & \quad c^{\text{inv}}(z_0) + c^{\text{op}}(z_0) \\
+ & \sum_{\omega \in \Omega} \pi(\omega) \left( E[c^{\text{inv}}(z_{\omega t}) | z_0] + E[c^{\text{op}}(z_{\omega t}) | z_0] \right) \\
\text{subject to} & \quad c^{\text{inv}}(z_{\omega t}) = \sum_{j \in J} c^{\text{arch}}(j, v) + \sum_{f \in F} c^{\text{tool}}(f, v) \\
& \quad c^{\text{op}}(z_{\omega t}) = T_{\text{total}}(z_{\omega t}) \cdot c^{\text{hour}} \\
& \quad T_{\text{total}}(z_{\omega t}) = \Theta(z_{\omega t}, d_p(\omega), l_p(\omega), B_p(\omega), V) \\
& \quad Z = \Gamma(B_p(\omega), K, J) \\
& \quad \omega \in \Omega, \quad z_{\omega t} \in Z, \quad j \in J, \quad f \in F, \quad v \in V, \quad v \in K, \quad \omega \in \Omega
\end{align*}
$$

The depicted configuration strategy aims at minimizing the objective function (22), representing the expected value of the overall cost over all the scenarios. In particular, $c^{\text{inv}}$ and $c^{\text{op}}$ are investment and operational cost respectively, calculated by using a given layout $z_{\omega t}$ in scenario node $\omega$ considering $\Omega$, as the set of scenario nodes under study. Equation (22) is made up of two parts, the first one takes into account the initial configuration decision $z_0$ while the second one considers future decisions $z_{\omega t}$, computing the expected values of expected costs in time period $t_0$ and based on the initial configuration decision $z_0$. A discount rate $q$ is applied, using $t_0$ as the time stage of the considered scenario node.

In both cases, investment costs are calculated according to (23) taking into account the set of $FAGs$ $j \in J$ to be acquired in the layout and the machine tools $f \in F$ acquired. In particular, the investment costs consider the decision about execution mode in the set $V = 1, \ldots, V \in K$. Equation (24) describes the operational cost as the product of the total time $T_{\text{total}}(z_{\omega t})$ and the hourly operational cost $c^{\text{hour}}$. The total time is the time to produce all the parts required in a given scenario $\omega$. It is estimated through the performance evaluator described in Section 5.2 and formalized in (25); it considers the completion time of the quantity to be produced $d_p(\omega)$, the batch size $l_p(\omega)$, the assembly process $B_p(\omega)$ and the execution mode selected for each $FAG j$. Only feasible layout configurations are considered for the optimization. Equation (26) defines how the set of configurations $Z$ is generated, considering the set of assembly processes $B_p(\omega)$ and the feasible, alternative execution modes $K$ to be implemented in the given set of $FAG J$. The model (22)-(26) is solved by considering each evolution scenario in isolation defined as a path from the root of the scenario tree ending in a leaf. Each solution is then used as input in (22) together with the occurrence probability associated to the evolution scenario addressed to calculate the expected value over all the scenarios.

The novelty of the Reconfiguration Planning tool is that it allows to generate layout configurations, evaluate their performances and thus develop plans for configuring and reconfiguring the assembly cell through several periods. In particular, the proposed approach takes advantage of the capability of the new assembly cell paradigm to support both changeover and reconfiguration actions. The result is a two-level reconfiguration tool, able to minimize the expected overall design and management cost of an assembly cell over the considered scenarios.

### 5.5. Interoperability and integration of the platform

To achieve an integration of the computational tools, all of them operate on the same database, enabling their sequential use. The central database ensures the interoperability of the tools by means of the Core Manufacturing Simulation Data (CMSD) standard model (Lee et al., 2011). Moreover, workflow-specific interfaces make possible the transfer of data between the tools. The tools are typically executed in the sequence presented in the order of the workflow in Figure 2. Nevertheless, the coupling between the tools aims to exploit the information feedback between them. The intention behind is that in case a solution turns out to be infeasible at a certain stage, the root solution can be refined by the tool working upstream in the workflow. In the proposed methodology, two main feedback loops are defined to exchange information among the tools.

After identifying a favorable system configuration from the alternatives in the set $S$ generated by the Assembly System Configuration tool, an individual cell of the chosen configuration is considered in detail using the Assembly Cell Configuration tool. In this step, it is important to evaluate whether the selected $FAGs$ can be arranged into a layout that is still compliant with the assumptions used in the Assembly System Configuration tool.
tool with regard to the cycle times and the available capacity. In case the production rate does not reach the target value, the bottleneck operations and the corresponding hardware are identified. Based on this information another solution from the set $S$ is used as input for the Assembly System Configuration tool; or the input data for the Assembly System Configuration tool is redefined to synthesize an updated set $S$ of system configurations (Figure 2). Otherwise, in case the layout can realize the target values, it can be considered valid and accepted.

The second main feedback loop is implemented to backlink the results of the Production Planning and Simulation tool to the Assembly Cell Configuration tool. In this case, the information added on the lower level primarily refers to the batch sizes $l_p(o)$, coming from the production planning tool. When calculating the layout configuration and the corresponding process sequence the batch sizes are assumed fixed. The planning tool, however, can determine variable batch sizes in order to match the requirements of the customers, leading to different average batch sizes. In this case, the evaluation of the performance is performed again considering the new average batch sizes.

Hence, the operational costs calculated by the Production Planning and Simulation tool can be used as new input for the Reconfiguration Planning tool to refine the configuration, considering the detailed logistics constraints, which can be decisive for the selection of the best configuration and reconfiguration actions.

6. Industrial application case

6.1. Presentation

In the following sections, the proposed approach is applied to the production of spare parts for the automotive sector. As described in Section 1, the scope of the tools is to support application environments characterized by a high-mix low-volume demand. To reflect this context, we consider the production of $|P|=4$ different products over a time horizon of 9 months divided in $|T|=3$ subperiods, each lasting three months. Switching from a time period to the following, the production volumes of the various products can change. The inherent uncertainty of the future evolution of demand is modeled through the scenario tree and the demand cases described in Table 11 (Appendix). Each product taken into account has its own assembly process which is described through the sequence of assembly operations, the specific equipment type and machine tools used, as well as the duration of the operations (included in Table 11, Appendix). The joining technologies taken into account are typical for the assembly of car bodies, such as different types of mechanical joining (nut pressing, riveting), resistance and adhesive joining. As described in Section 3, each of these technologies is embodied by fixed equipment (e.g. 7-axis robots) and exchangeable FAG devices and their respective machine tools. For this reason, the process requirements described in Table 11 (Appendix) include both the tool type represented with $T\#$, and the processing time in seconds.

For each FAG and the other production equipment components the specific 2D dimensions are known, as well as the technological indication regarding failure and repair rate. This data is used during system configuration, analysis and layout generation. Moreover, also financial information such as the purchasing price for specific equipment (from the minimum one of 10,000 € of the modular device, to the maximum one of 120,000 € of the control unit), operative hourly cost (50 €/h) or the time and cost for reconfiguration (2 working weeks, 20,000 € respectively) is taken into account, all presented in Table 10 (Appendix).

Based on this problem description, the integrated approach was applied. In the following Sections, the application of each tool is described and the results for a single scenario are presented.

6.2. Assembly System Configuration tool application

First, the Assembly System Configuration tool uses available information about expected situations of future product demand (cf. highest, medium and lowest demand case in Table 11, Appendix) to generate design candidates that can be used to face these scenarios. To this aim, instances of the equipment components from the database (cf. Table 10, Appendix) are clustered into multiple assembly cells and products are allocated afterwards. Due to the random factors that vary the system design parameters, various production systems are generated for the presented problem. These configurations are analyzed with regard to their performance.

Afterwards, the Assembly System Configuration tool visualizes the various generated designs and related performance parameters to the decision-makers: Using the adjustable GUI of the tool (cf. Figure 3) for exploring and comparing solutions, they can exclude solutions by specifying design and performance constraints for the design candidates to eventually obtain various, feasible solutions, such as the ones shown in Table 8. The resulting system configurations embody different production strategies, which are described in column Config ID, Table 8.

Decision makers can assess the configurations with regard to their varying cost profiles that are caused
by differing resource clustering and product allocation. The distinct performance of the presented configurations stems from the allocation of bottlenecks and amount of required changeovers in the system, which both affect direct production cost. Also the higher number of transports and higher stock levels in the multicell solutions have influence on the logistics and storage cost. Additionally to these operational KPIs, decision makers can take into account required investment and space, as well as the degree of system utilization (cf. Table 8). In the presented case, the single cell configuration may be the most suitable solution, if decision-makers seek for a solution that strikes a balance of investment and total cost.

6.3. Assembly Cell Configuration tool application

The Assembly Cell Configuration tool considers the product allocation and the pieces of equipment identified in the previous step, and arranges them in alternative layouts. The different execution modes described in Section 5.2 are considered in order define the task sequencing and set up the performance evaluation model.

To illustrate the results of the tool, we consider two different layouts (A and B) generated in relation to two different execution modes for all the assembly operations, namely number 1 and 4. Both layouts pursue the Single cell architecture coming from the Assembly System Configuration tool where all the products are processed in the same cell. According to the scenario tree, we consider the first time period only, in scenario \( \omega_0 \), and test the two layouts producing part types 1 and 3. The two layouts are represented in Figure 5, Appendix.

The two considered execution modes imply a different task sequence for the assembling operations and, consequently, their performances are different. Table 12 reports the throughput of the solutions, highlighting that layout A performs significantly better compared to layout B. The reason is that the for the latter, the 7-axis robot is used to move parts but also to hold them while the FAGs work. Hence, the 7-axis is loaded more, compared to layout A, causing the assembly process to take more time. In addition to the performance evaluation, also the different costs associated to the two solutions must be taken into consideration, as reported in Table 12 (Appendix), to support the selection of the best alternative.

6.4. Production Planning and Simulation tool application

Applying the Production Planning and Simulation tool, one can analyze the future-expected operative costs and production batch sizes, based on the contractual delivery volumes known already in the early design stage. Relying on the defined application case, the inputs of the tool are the system configurations in the subsequent time periods, as well as the delivery volumes agreed with the customers. The main purpose of the tool is to refine the estimation on the batch sizes: in case of the previous tool of the workflow, average batch sizes and frequency of the deliveries are considered, while in this case they are calculated by matching the order stream with the detailed system structure. Executing these plans in the DES model of the system allows to determine more accurate operative costs compared to the previous tool, as additional information can be utilized, such as inventory, personnel and backlog costs. The refined operative costs are meaningful feedback information that can be applied by the Reconfiguration Planning tool to select the cost-optimal reconfiguration strategy. Besides, the batch sizes can be utilized by the Assembly Cell Configuration tool to evaluate and/or refine the cell configuration.

<table>
<thead>
<tr>
<th>Config ID</th>
<th>System configuration</th>
<th>Area occupied ( [\text{m}^2] )</th>
<th>Initial investment ( [\text{\texteuro}] )</th>
<th>Direct prod. cost ( [\text{\texteuro}] )</th>
<th>Logistics and storage cost ( [\text{\texteuro}] )</th>
<th>Total cost ( [\text{\texteuro}] )</th>
<th>Non-utilized capacity value ( [\text{\texteuro}] )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dedicated cells</td>
<td>P[1]→ C1: [Op1, Op2, Op3, Op4];</td>
<td>142</td>
<td>1,650,000</td>
<td>8,758</td>
<td>13,135</td>
<td>21,893</td>
<td>403,599</td>
</tr>
<tr>
<td>Split processes 1</td>
<td>P[1-4]→ C1: [Op1, Op2];</td>
<td>67</td>
<td>749,000</td>
<td>9,856</td>
<td>19,535</td>
<td>29,391</td>
<td>177,394</td>
</tr>
<tr>
<td>Split processes 2</td>
<td>P[1,3,4]→ C1: [Op1];</td>
<td>70</td>
<td>764,000</td>
<td>13,777</td>
<td>24,870</td>
<td>38,647</td>
<td>177,223</td>
</tr>
<tr>
<td>Split processes 3</td>
<td>P[1,3,4]→ C1: [Op1];</td>
<td>104</td>
<td>1,247,000</td>
<td>11,265</td>
<td>33,470</td>
<td>44,735</td>
<td>300,485</td>
</tr>
<tr>
<td></td>
<td>P[1-2]→ C3: [Op3];</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>P[1-2]→ C4: [Op4];</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 8: The system configurations in set S generated by the Assembly System Configuration tool. "P", "C" and "Op" denote products, cells and operations, respectively. Performances presented for the case of lowest demand.
In the experiments, four different scenarios are analyzed by the planning and DES models. In the first scenario (contractual), the contractual delivery volumes and frequency were applied (represented by variables \(d_{pu}\)), evaluating the solutions calculated by the Assembly Cell Configuration tool considering the ideal order stream. In the other three scenarios (Sc 1-3), delivery frequencies are increased by splitting the total volumes in smaller parts. In these scenarios, the total volumes are the same, while the delivery frequency is increased by 10–20–30% subsequently. This results in smaller production batch sizes, more changeovers and thus higher operative costs, which often occur in the real life. All the experiment results are reported in Table 13, Appendix.

The results show that even in the contractual case, the operative costs are higher than the ones considered by the previous tools. This refined information can be applied by the Reconfiguration Planning tool, if one assumes that contractual volumes will not change in the future. A more conservative solution is to assume the operational costs resulting in one of the scenarios Sc 1-3, that led to smaller batch sizes and higher costs.

6.5. Reconfiguration Planning tool application

The Reconfiguration Planning tool exploits the results of the previous tools to select a robust robust solution with minimum cost, i.e., a configuration able to face the considered market evolutions together with a proper reconfiguration plan. The solution selected for the case study is the one in Figure 4 for the three time periods. All the equipment needed for the whole time horizon are installed in the initial configuration. In order to cope with the production of different part types, a setup is needed to switch tools and molds. The adopted solution provides fast and swift setup capability, whose impact on the production performance is low. We estimate only 30 minutes to switch from the production of one part type to another. The solution provided adopts execution mode number 1 among the alternative ones available, as described in Section 5.2.

Table 9 reports the overall costs for the scenario \(\omega_0 \rightarrow \omega_1B \rightarrow \omega_2C\), associated with the solution proposed by the robust approach. This one is compared with two alternative solutions, i.e., the single path optimum approach and the initial node optimum approach. The first one takes into account a single scenario \(\omega_0 \rightarrow \omega_1B \rightarrow \omega_2C\) and looks for the best configuration in each time period; the second one considers the best solution for the first time period only. The comparison considers investment, reconfiguration and operational costs, showing that the robust solution is the minimal-discounted-cost configuration, able to avoid the need of reconfiguration actions (as needed in the single path optimum approach). The robust approach suggests the installation of pieces of equipment in advance respecting the actual need; pursuing this strategy incurs no reconfiguration costs. These costs could have relevant impact on the discounted total cost, as for the single path approach reported in Table 9 in which two reconfiguration actions are considered.

The robust approach solution also gives up pursuing the local optimality in the second time period (compared with the second strategy solution) addressing global optimality in all time periods. It becomes clear that the myopic approach, which pursues the initial optimum configuration, provides a solution that is infeasible in the last time-period.

<table>
<thead>
<tr>
<th>Cost Type</th>
<th>Initial node optimum</th>
<th>Single path optimum</th>
<th>Robust</th>
</tr>
</thead>
<tbody>
<tr>
<td>Investment</td>
<td>9,955</td>
<td>10,380</td>
<td>13,399</td>
</tr>
<tr>
<td>Operational</td>
<td>-</td>
<td>20,000</td>
<td>20,000</td>
</tr>
<tr>
<td>Reconfiguration</td>
<td>-</td>
<td>-</td>
<td>20,000</td>
</tr>
<tr>
<td>Total (discounted)</td>
<td>547,640</td>
<td>575,640</td>
<td>604,668</td>
</tr>
</tbody>
</table>

Table 9: Comparison in terms of costs (all expressed in €) between the solution obtained with the presented robust approach and two alternative approaches, the single path optimum and the initial node optimum.

In order to assess the robustness of the approach, we...
tested the selected solution in the most extreme scenarios in the tree, those with the minimum demand stream \((\omega_0 \rightarrow \omega_1 \rightarrow \omega_2)\) and the maximum one \((\omega_0 \rightarrow \omega_1 \rightarrow \omega_F)\). Since the behavior of the robust solution in the first time period of both scenarios \((\omega_0)\) has been already addressed (Table 9), we focus the analysis on the other scenario nodes, considering the production volumes to be addressed and an available time of 480 h. To exploit the results of the performance evaluation, the time needed to satisfy the requirements is calculated. Specifically, 64 h are needed in scenario node \(\omega_1\), 209 h in scenario node \(\omega_2\), 427 h in scenario node \(\omega_3\) and 411 h in scenario node \(\omega_4\). Hence, the robust solution is capable to satisfy the market requests in all the scenarios, including the extreme ones with the lowest occurrence probability that were initially used as input for the Assembly System Configuration tool.

6.6. Evaluation of solutions applying feedback loops

In the proposed approach, we considered two feedback loops. The first one returns the configuration obtained by the Assembly Cell Configuration tool to the Assembly System Configuration tool. This feedback is relevant in case it is not possible to define a feasible configuration of the cell starting from the system configuration provided. Grounding on this feedback, the Assembly System Configuration tool should provide a new system configuration. In the presented example this is not the case and, hence, an iteration is not needed.

The second feedback uses the lot sizes calculated by the Production Planning and Simulation tool to refine the operational costs estimation of the Assembly Cell Configuration tool as described in Section 5.3. As reported in Table 13, the need to consider all the details and constraints at the planning level could entail feasible lot sizes that are different from the ones used in the Reconfiguration Planning tool (up Table 11, Appendix). Hence, the updated batch sizes can be fed back to the Assembly Cell Configuration tool so that the performance evaluation can be carried out again. Moreover, smaller lot sizes also affect operational cost and performance requiring the Reconfiguration Planning tool to look for a new optimal solution. Using the ideal batch sizes yields the same optimal configuration evolution (Figure 4), yet with different operational costs, namely 10,932 €, 10,482 € and 13,665 € for the respective periods. Grounding on these updated costs, we obtain a new overall discounted cost of 606,026 €, in contrast to the previous figure of 604,686 € (Table 9). Considering the contractual delivery dates, the new operational costs, coming directly from the Production Planning and Simulation tool, would be 13,714 €, 15,456 € and 17,779 € respectively (see Table 9 in the Appendix). The refined total discounted cost of 617,050 € would still represent the optimal value. Notice that, the impact of the updated batch sizes accounted to more than 10,000 €, thus providing a more accurate estimation of the cost for the considered solution.

6.7. Computational performance of the applied models

While the overall framework was presented as an integrated approach, each tool has been developed and implemented separately and then linked according to the described architecture.

The Assembly System Configuration tool with its graphical user interface is a prototype software environment for design space exploration. The design synthesis algorithm was implemented in C#. By using the heuristic approach for configuration synthesis, sets of solutions are generated in reasonable computation time, which is typically less than 10 seconds per set containing 50 solutions for testing problems up to \(|P|= 20, |J|= 7\). Yet, it should be noted that the time needed depends on the constraints applied by the users in combination with the occurrence probability of the configurations in the total solution space. Hence, considering a large solution space with narrow performance requirements may result in higher solution times until 50 valid configurations for the set could be found. The tool was validated by considering a real system configuration problem, verifying that the solutions that were manually designed by the production engineers could be found among the ones that were automatically generated by the tool.

The Assembly Cell Configuration tool was implemented in C++ and Java. The complexity of the layout generation and evaluation problem is affected by the number of FAGs \(|J|\) and products \(|P|\) considered for each scenario node \(\omega\). Indeed, the number of candidate layouts generated increases with the number of FAGs to be arranged within the cell; each of these candidate has to be evaluated by considering the assembly process of each product. For the use-case presented, more than 1000 layout solutions were generated, requiring a total solution time of 174 minutes on a 64 GB memory and 2.6 GHz CPU machine. With the introduction of an additional product, the computational time increases accordingly to the number of additional feasible layouts.

The production planning and DES models were implemented in FICO Xpress and Siemens Tecnomatix Plant Simulation, respectively. The planning model addresses the whole system, potentially including multiple cells (including the one under evaluation) sharing a common pool of hardware modules.
The complexity of the problem is characterized by the average values of $|P|= 20$, $|C|= 5$, $|J|= 7$, and the contractual delivery frequency of the products was $u = [4, 12]$ on a $|T|= 60$ length horizon, covering a single time bucket $t \in T$ of the reconfiguration planning model. This resulted in the average running time of 44 seconds with Xpress’s default MIP solver, until an optimality gap of at most 5% was achieved.

The Reconfiguration Planning tool was implemented in C++ and its calculation complexity is primary affected by the number of time buckets $|T|$ considered. The example considered in the previous section could be solved in less than 10 minutes when executed on a similar system like the Assembly Cell Configuration tool; with the introduction of an additional time bucket, the computational time reaches about 20 hours.

The verification process of the mathematical models consisted of two steps. First, the KPI values provided by the different models were compared, whether they differ significantly. If not, the second verification step was taken by applying the DES model, evaluating whether the generated design and technological processes meet the generic requirements, regarding the constraints applied in the models. In order to accept the results provided by the DES, the simulation model was validated by analyzing existing cells, proving that the provided KPI values match the actual ones. The Reconfiguration Planning tool considers all possible configuration evolutions and takes the minimum cost one, the verification is reached as a direct consequence.

7. Conclusions and outlook

This paper proposed an integrated approach to support the design and management of reconfigurable assembly systems, which is designed around four decision-support tools that are connected through a workflow and ground on the same data structure. The approach enables decision-makers to derive use from different decision-support techniques for multiple decision stages by enabling design space exploration, layout optimization, production planning and simulation, as well as developing optimized reconfiguration strategies.

Although integrated approaches focusing on design and management of assembly systems already exist, the reconfigurable cell architecture addressed asked for special and tailored models. These models exploit the FAG and execution mode concepts, providing solutions that leverage the capabilities of the special production system.

An additional novelty of the approach are the iterative loops with feedbacks, integrating the individual tools in the design workflow to support the allocation of the products to the cells, the detailed system design over multiple periods as well as the integration of proper management policies. Combining these tools also aims to speed up design and planning decisions, which eventually should improve planning efficiency.

A case study on an industrial example is presented to demonstrate the practical applicability and potential benefits of the approach. It is our ambition to motivate other researchers to develop similarly modular support frameworks that consider all models in the context of the preceding and consecutive steps in the design and operation processes.

Future work will focus on further harmonizing the capabilities of the individual tools to obtain a workflow providing a balanced level of detail. In this context, the implications of reconfiguring multiple cells will be integrated into the Assembly System Configuration tool. Furthermore, consideration of multiple cells will be integrated into the Assembly Cell Configuration tool, and stochastic optimization approaches—regarding the production planning problem—will be applied in the Production Planning and Simulation tool to cope with the possible uncertainty of the parameters. Regarding the Reconfiguration Planning tool, further development will address the possibility of considering risk-based objective function, to address robustness in a more effective way.

8. Acknowledgments

This research has been partially supported by the EU FP7 Project No: NMP 2013-609087, RobustPlaNet - Shock-robust Design of Plants and their Supply Chain Networks.

9. Bibliography

References


Chica, M., Bautista, J., Córdón, O., Dámas, S., 2016. A multiobjective model and evolutionary algorithms for robust time and space assembly line balancing under uncertain demand. Omega 58, 55–65.


Ponsignon, T., Mönch, L., 2014. Simulation-based performance assessment of master planning approaches in semiconductor manu-
facturing. Omega 46, 21–35.
(a) Solution that considers all the assembly operations implemented with execution mode number 1.

(b) Solution that considers all the assembly operations implemented with execution mode number 4.

Figure 5: Two possible solutions for the same problem: candidate layouts are able to satisfy market requests by considering alternative execution modes.

<table>
<thead>
<tr>
<th>Period</th>
<th>KPI</th>
<th>Ideal</th>
<th>Contractual</th>
<th>Sc 1</th>
<th>Sc 2</th>
<th>Sc 3</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>10,864</td>
<td>13,314</td>
<td>14,090</td>
<td>16,028</td>
<td>17,184</td>
</tr>
<tr>
<td>t₀</td>
<td>Batch P1</td>
<td>40</td>
<td>124</td>
<td>42</td>
<td>42</td>
<td>33</td>
</tr>
<tr>
<td></td>
<td>Batch P2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>Batch P3</td>
<td>50</td>
<td>50</td>
<td>40</td>
<td>30</td>
<td>30</td>
</tr>
<tr>
<td></td>
<td>Batch P4</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>Operational cost</td>
<td>11,478</td>
<td>15,456</td>
<td>16,627</td>
<td>18,663</td>
<td>20,677</td>
</tr>
<tr>
<td>t₁</td>
<td>Batch P1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>Batch P2</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td></td>
<td>Batch P3</td>
<td>30</td>
<td>53</td>
<td>53</td>
<td>40</td>
<td>40</td>
</tr>
<tr>
<td></td>
<td>Batch P4</td>
<td>35</td>
<td>42</td>
<td>33</td>
<td>33</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td>Operational cost</td>
<td>14,637</td>
<td>17,779</td>
<td>19,406</td>
<td>22,452</td>
<td>21,772</td>
</tr>
<tr>
<td>t₂</td>
<td>Batch P1</td>
<td>35</td>
<td>127</td>
<td>124</td>
<td>124</td>
<td>124</td>
</tr>
<tr>
<td></td>
<td>Batch P2</td>
<td>40</td>
<td>47</td>
<td>40</td>
<td>33</td>
<td>27</td>
</tr>
<tr>
<td></td>
<td>Batch P3</td>
<td>35</td>
<td>50</td>
<td>50</td>
<td>33</td>
<td>33</td>
</tr>
<tr>
<td></td>
<td>Batch P4</td>
<td>35</td>
<td>42</td>
<td>33</td>
<td>33</td>
<td>33</td>
</tr>
</tbody>
</table>

Table 13: Feedback on the resulted operation costs and batch sizes, provided by the Production Planning and Simulation tool. The Ideal includes the costs and batch sizes considered by the previous tools, whereas Contractual refines these costs. Scenarios Sc 1-3 assume that contractual delivery volume might change in the future resulting in more frequent deliveries.