Joint optimization of product tolerance design, process plan, and production plan in high-precision multi-product assembly

Daisuke Tsutsumi[*a,b], Dávid Gyulai[c], András Kovács[c], Bence Tipary[c], Yumiko Ueno[a], Youichi Nonaka[a], Kikuo Fujita[b]

[a]*Hitachi Ltd., Research & Development Group, 292 Yoshidacyo, Totsuka, Yokohama, Kanagawa, 244-0817, Japan*

[b]*Department of Mechanical Engineering, Osaka University, 2-1 Yamadaoka, Suita, Osaka 565-0871, Japan*

[c]*EPIC Centre of Excellence in Production Informatics and Control, Institute for Computer Science and Control (SZTAKI), Kende u. 13-17, H-1111 Budapest, Hungary*

Abstract

With the ever-increasing product variety faced by the manufacturing industry, investment efficiency can only be maintained by the application of multi-product assembly systems. In such systems, the product design, process planning, and production planning problems related to different products are strongly interconnected. Despite this, those interdependent decisions are typically made by different divisions of the company, by adopting a decomposed planning approach, which can easily result in excess production costs. In order to overcome this challenge, this paper proposes an integrated approach to solving the above problems, focusing on the decisions crucial for achieving the required tolerances in high-precision assembled products. The joint optimization problems related to product tolerance design and assembly resource configuration are first formulated as a mixed-integer linear program (MILP). Then, a large neighborhood search (LNS) algorithm, which combines classical mathematical programming and meta-heuristic techniques, is introduced to solve large instances of the problem. The efficiency of the method is demonstrated through an industrial case study, both in terms of computational efficiency and industrial effectiveness.

[*]Corresponding author

Email address: `daisuke.tsutsumi.jh@hitachi.com` (Daisuke Tsutsumi[*])

Keywords: Assembly, Design optimization, Tolerancing, Process planning, Production planning

1. Introduction and motivation

In response to diversifying consumer preferences, many companies from the automotive, electronics, and consumer goods industries are forced to increase product variety [1, 2, 3]. The situation is often complicated further by the changes of the conventional manufacturer-supplier relationships, e.g., in the automotive industry, where a single supplier now serves many manufacturers. Therefore, the supplier must increase its product variety, and the demand for multi-variety production grows. As a consequence, requirements of new products often cannot be satisfied by existing manufacturing and assembly lines, and therefore, investment into new equipment is inevitable. There are also attempts to lift manufacturing constraints by introducing general purpose equipment, but excessive generalization or flexibility of equipment can also lead to low production rate and low return on investments [4].

In the conventional product development process, different phases of the process focus on different issues to be resolved: first of all, product design has to meet customer specifications by selecting appropriate design alternatives. When a product design is available, process planning is responsible for realizing the design by defining the assembly resource configurations. In the operation stage, production planning assigns products to resources over time to satisfy demand in the most efficient way. An important business challenge is to maintain profits via internal efficiency by minimizing total production costs while using existing assembly resources efficiently. However, with the ever-changing product portfolio, not only the existing resources, but also investments into new production resources are part of the game.

The increase of product variety is often led by the product design department, whereas process and production planning are carried out in subsequent steps [5]. Hence, in the conventional product development process, there is no appropriate feedback mechanism, and as a result of limited consideration of production aspects in product design, it is not possible to benefit from the introduction of a common assembly system that enables multi-product assembly [6]. Consequently, individually optimized and less versatile assembly systems are abused, leading to a decline in return on assets due to excessive investment [7]. In general, the key challenge in multi-product assembly is to find the best tradeoff between product design, process and production planning aspects considering a portfolio of diverse products, changing demand volumes, alternative resources, and investment options over time.

However, there are traditional walls among the product design, process planning, and production planning domains that altogether constitute the product development process [5, 6]. Several traditional methods, for example the well-known *Design for Manufacturing and Assembly* (DFMA) approach [8] attempt to break these walls and enforce production aspects in product design. Nevertheless, their application is limited, and they often provide unsatisfactory feedback. The objective of this research is to open new avenues from production back to product design for the efficient use of existing assembly resources. It is important to highlight that the proposed method is completely based on formal mathematical models, instead of the commonly applied rule-based decisions.

The product development process targeted by this research has many sub-processes with complex interdependencies. Among these sub-problems, focus is given to product tolerance design and resource configuration for assembly processes related to achieving the specified tolerances. As product quality is also affected by tolerance schemes [9], tolerance design is also one of the most important steps in product design development. Figure 1 shows the location of tolerance design and assembly resource configuration within the product development process.

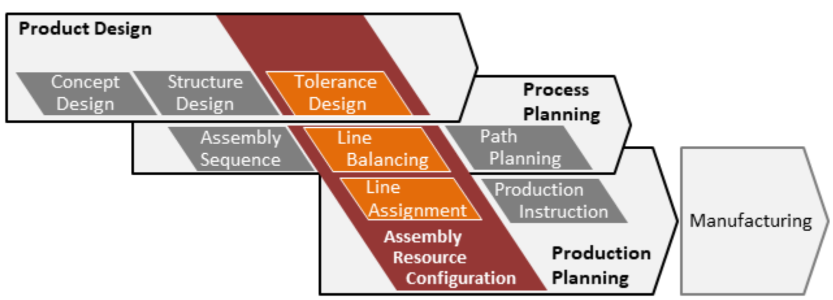

Figure 1: Location of tolerance design and assembly resource configuration within the product development process. Figure adapted from [10].

The structure of the paper is as follows. In Section 2, a literature review is provided, summarizing conventional approaches in each area of the product development process. In Section 3, the problem in scope is formally defined.

Section 4 introduces the proposed solution approach in detail. Then, a case study is provided in Section 5 to evaluate the efficiency of the proposed methodology in terms of computational efficiency and industrial effectiveness. An outlook on practical applications is given in Section 6. Finally, conclusions are drawn and directions for future research are pointed out.

2. Literature review

The product development process targeted in this research can be divided into product design, which determines the functionality, structure, and geometry of the product; process planning, which defines the manufacturing and assembly technologies together with the required resources to produce the product according to its specifications; and production planning that matches the manufacturing and assembly process to resources over time to satisfy demand. This section summarizes the state-of-the-art in each of the above fields, with special attention to earlier attempts to integrate them.

2.1. Product design stage

While various systematic methodologies have been defined in the literature to support product design [11, 12], a major step was taken towards the consideration of subsequent stages of the product development process with the introduction of various *Design for eXcellence* (DFX) approaches. Notably, *Design for Manufacturing* (DFM) focuses on the ease of manufacturing the individual parts; *Design for Assembly* (DFA) addresses the efficiency of assembling the parts; whereas *Design for Manufacturing and Assembly* (DFMA) seeks to combine the benefits of both DFM and DFA. These methods all seek to reduce overhead, material and labor costs, as well as product development time by using standards and defining appropriate rules. At the same time, they focus on different stages of the production process and accordingly, apply different strategies. The most widespread DFMA approaches are Boothroyd and Dewhurst's method [8], the general production checklists by Huang [13], the Hitachi Assembly Evaluation Method (AEM) [14], the Lucas method [15] and assembly-oriented design by Redford and Chal [16]. Design frameworks and automatic tools are proposed to exploit concurrent design possibilities, considering product life-cycle features already in the early conceptual design phase. Molcho et al. [17] take on bridging the gap between the designers, process planners and manufacturers by establishing a knowledge and rule base.

Nevertheless, the crucial role of generic guidelines and rules of thumb is ubiquitous in the above approaches, except for some specific applications, such as the design of battery systems for electric vehicles in [18]. A major problem with conventional DFMA in general multi-product assembly is that too strict guidelines and the difficulty of updating the guidelines make it impossible to avoid product designs that violate the guidelines. As a result, investment into additional manufacturing and assembly equipment is inevitable, which means that the efficiency of conventional DFMA decreases in multi-product assembly.

While most contributions on product design and process planning deal with ideal, nominal products, real manufactured and assembled products never match the nominal design precisely. On the contrary, the allocated tolerances are decisive on the applicable manufacturing and assembly processes, and consequently, on production costs as well. For this reason, this paper focuses primarily on the tolerance design sub-problem of product design.

In reality, product geometry and dimensions deviate from the nominal because of variations during both the manufacturing and the assembly processes. *Dimensional tolerances* have been for long the primary means for expressing the allowable deviations of parts and products, and geometrical tolerances have been formally defined and standardized only recently by the introduction of *Geometrical Dimensioning and Tolerancing* (GD&T) [19] for a richer characterization of the allowed deviation. For modeling *cascading tolerances* in assemblies, vector-chain approaches are the most widespread both in academics and industrial practice [20]. This approach, as well as all other mainstream models assume that relevant quality features of the final product are described by so-called *Functional Key Characteristics* (FKCs) which are influenced by different factors. The dimension chains related to different FKCs are often interrelated. Despite this, until the 1990s, all major works on tolerance optimization assumed independent dimension chains. The first contribution in tolerance design that can handle interrelated dimension chains is considered to be [21]. A method for evaluating multiple FKCs simultaneously in the assembly of compliant parts, such as sheet metal, using a combination of *Finite Element Analysis* (FEA), tolerance analysis, and Monte Carlo simulation is proposed in [22].

Product tolerance design, when product structure is perfectly defined, reduces to the problem of *tolerance allocation*, i.e., assigning tolerances to given individual dimensions. Various computational approaches have been applied to solving this problem, including genetic algorithms [21, 23], ant colony optimization [24], particle swarm optimization [20], numerical methods [25, 26] for optimizing a well-defined objective, as well as ontologies and rules to determine tolerances by exploiting technological knowledge (but without explicitly considering any optimization criterion) [27]. Most of these works address the best distribution of dimensional tolerances for optimizing some interpretation of the total production cost, while only a few contributions are available that account for geometrical tolerances as well.

2.2. Process and production planning stage

With a focus on assembled products, *assembly planning* (AP) creates a detailed assembly plan to craft a complete product from individual parts, considering aspects like product and part geometries, available resources (machines, tools, fixtures, feeders, etc.) as well as technological constraints [28]. When AP is solved by some automated techniques, following the manufacturing nomenclature, it is also called *computer-aided (assembly) process planning* (CAPP). Solving AP/CAPP requires making diverse types of decisions, and accordingly, it is usually solved by some decomposition approach. A typical decomposition scheme subdivides AP into the following three sub-problems [28]: (1) *Assembly Sequence Planning* (ASP), in which a sequence of (expectantly technologically and geometrically feasible) assembly operations is computed; (2) *Assembly Line Balancing* (ALB), in which the assembly operations are assigned to assembly stations in such a way that station workloads are balanced; and (3) *Assembly Path Planning* (APP), which computes collision-free paths for joining different parts or sub-assemblies in individual assembly operations.

Various data models have been proposed to enable the automatic generation of process plans. Such models capture information on the target product, the applied equipment, as well as the manufacturing and assembly process. Specifically, data models have been proposed for describing the product structure and its features [29], product structure extended with tolerances and quality [30], workers' abilities and ergonomics [31], fixtures and grasping [32]. In addition, models dedicated to specific fields have been proposed, such as the final assembly of automotive vehicles [33] and aircrafts [34]. Finally, there are ambitious initiatives, such as the ontology model by NIST [35], aiming at the generalization of the data models to assembled products, but these have not been put to practical use so far. However, process planning methods are still specialized for a given product family or an assembly technology, and lack a feedback mechanism for product design, in particular for multi-product assembly.

2.3. Computational methodology

The combined product tolerance design, process planning, and production planning problem addressed in this paper is a complex combinatorial optimization problem. For production planning models similar in their structure, mathematical programming, and especially *mixed-integer linear programming* (MILP) approaches have been predominant and have proven efficient [36]. Nevertheless, when solving complex and large instances of the problem, mathematical programming approaches might be insufficient on their own, and the application of meta-heuristics may become the most effective approach.

A research direction of increasing importance in operations research is combining the strengths of mathematical programming and meta-heuristic approaches in so-called *matheuristics* [37]. The *large neighborhood search* (LNS) algorithm [38, 39] was motivated precisely by the need for combining exact solution methods with local search in applications where exact solution approaches (e.g., branch-and-bound for a MILP or a constraint program) outperform pure meta-heuristic approaches, but still, they do not scale up to realistic problem sizes. LNS consists in constructing first an initial solution using some heuristic, and then, iteratively looking for improvements in some neighborhood of the current solution. However, the efficient exact solution approach (MILP in our case) makes it possible to search a very large, potentially exponential size neighborhood in each iterative step. LNS has been successfully applied to various fields of combinatorial optimization, including scheduling [40] and vehicle routing problems [41].

2.4. Positioning of the paper

A simplified version of the current problem was investigated by the authors in the recent paper [10]. A decomposition approach was introduced that separated the solution of the tolerance allocation and the assembly resource configuration sub-problems. Case studies based on industrial data confirmed that the approach can effectively reduce production costs and improve investment efficiency.

The present paper addresses the generalization of the previous contribution in industrially relevant directions, including a generic tolerance model with interrelated dimension chains, multiple target assembly processes, as well as differentiating human and automated processes for achieving the required precision by adjustments. The extension of the model also required the development of novel, efficient solution approaches instead of the decomposition scheme described in [10].

3. Problem statement

In the paper, a complex optimization problem is investigated with the aim of reducing the overall production-related costs through the proper combination of product design, process planning and production planning decisions in high-precision multi-product assembly. In order to provide a comprehensible definition, the presentation of the overall problem is separated into four sub-sections as follows:

1. the *tolerance design* sub-problem, which involves the selection of the appropriate structural design alternatives and the assignment of tolerance values to individual dimensions to meet the tolerance requirements on the assembled products;

2. the *assembly resource configuration* sub-problem, which aims to match forecast demand to assembly resources, considering existing and potential future resource capabilities and capacities, as well as the process requirements according to the above defined product design;

3. the definition of the *production costs* and the *depreciation model* to characterize the quality of the solutions; and

4. a recapitulation of the *assumptions* made.

3.1. Tolerance design sub-problem

The tolerance design sub-problem is responsible for selecting the appropriate structural design alternative for each product from a list of alternatives given in the input, and for defining the tolerance values on the individual dimensions in such a way that the tolerance requirements on the assembled product are satisfied, and the total production costs are minimized. While the satisfaction of tolerance requirements can be verified solely on the solution of the tolerance design sub-problem, production costs also depend on the solution of the assembly resource configuration sub-problem.

The formal definition of the tolerance design sub-problem is the following. There is a set of products P to be produced in a common multi-product

Table 1: Notation (tolerance design).

Indices, sets	
δ	Dimension (index)
Δ	Dimension chain (index)
d	Design alternative (index)
$M(d)$	Set of manufactured dimensions of alternative d
p	Product (index)
P	Set of products
$D^+_{d,k}$	Set of dimension chains in alternative d with adjustment by process k
D^-_d	Set of dimension chains in alternative d without adjustment
Input parameters	
φ_Δ	Design specification for dimension chain Δ [$\pm$ mm]
ϱ_Δ	Adjustment range of dimension chain Δ [$\pm$ mm]
Decision variables	
y_d	Variable indicating that design alternative d is selected for production
τ_δ	Tolerance on dimension δ [$\pm$ mm]
$\overline{r}_{d,k}$	Adjustment precision on process k required for assembling alternative d [$\pm$ mm]

assembly system, containing both *existing* and *new* products. The design of the existing products is fixed. In contrast, multiple candidate structural design alternatives, provided as input by a designer, are available for the new products, whose production begins during the planning horizon. For each new product, a single design alternative must be selected for production, and the design cannot be altered later.

Each given structural design alternative specifies the product structure in terms of nominal geometries of the parts and their relations, which defines the dimension chains Δ of each alternative d. Different dimension chains can share common dimensions (i.e., $\Delta_1 \cup \Delta_2 \neq \emptyset$), which can be both adjusted and non-adjusted dimensions. Design requirements are given in terms of tolerance specifications φ_Δ on each dimension chain Δ.

At the same time, tolerance values τ_δ on individual dimensions δ are not part of the input (this is why the design alternatives in the input are called only *structural* alternatives); instead, they must be calculated in the tolerance design sub-problem in such a way that the requirement specifications are satisfied for the selected design alternative d for each new product p. Without loss of generality, this paper assumes symmetric tolerances, where the upper (+) and lower (-) values are equal, e.g. $\tau_\delta = \pm 0.01$ mm.

For design alternatives with adjustment, it is assumed that there is at most one adjusted dimension in each dimension chain. Moreover, adjustment takes place after the assembly of all parts related to the involved dimensions, and therefore, adjustment can compensate the deviation of all dimensions in the chain (note that this assumption can be lifted with a minor generalization of the model presented here). Consequently, tolerance specifications can be met in two different ways:

- For dimension chains without adjustment, i.e., with fully defined connections between the parts, the tolerance values on the individual parts' dimensions need to be specified so as to guarantee that the stacked tolerance values satisfy the design specifications:

$$\sum_{\delta \in \Delta} \tau_\delta \leq \varphi_\Delta$$

- For dimension chains with adjustment option, the stacked tolerance value can be greater than the design specification. However, this is compensated by adjustment within a predefined range, ϱ_Δ, which decreases the stacked tolerance. Nonetheless, the precision of the adjustment itself, r_Δ, must be taken into account:

$$\sum_{\delta \in \Delta} \tau_\delta \; - \; \varrho_\Delta + r_\Delta \leq \varphi_\Delta$$

  Moreover, the adjustment precision r_Δ itself must satisfy the design specification:

$$r_\Delta \leq \varphi_\Delta$$

Finally, each adjusted dimension is unambiguously assigned to a certain assembly process k that performs the adjustment. Accordingly, the tolerance design sub-problem also involves the specification of the required adjustment precision value $\overline{r}_{d,k}$ for process k. The notation applied for the tolerance design sub-problem is summarized in Table 1.

3.2. Assembly resource configuration sub-problem

The assembly resource configuration sub-problem is responsible for matching the demand for the products to existing, new, or upgraded assembly resources. This involves the following types of decisions:

- Deciding on the potential construction of new assembly lines.

- Automating selected processes on assembly lines.
- Upgrading selected automated processes to improve their adjustment precision (while the precision of human processes is assumed to be fixed).
- Assigning products (with their corresponding design alternatives) to assembly lines in such a way that all capacity and capability requirements are satisfied.

Formally, each product $p \in P$ in each period $t \in T$ must be assigned to some assembly lines. Products can be assigned to at most Π assembly lines at a time (noting that $\Pi = 1$ in most of the use cases investigated), which can be either existing lines or newly built lines. On the other hand, an arbitrary number of products can share the same assembly line.

Modifying the product-line assignment over the horizon is allowed, however, this comes with a changeover cost of c^X and a changeover time of a^X on the newly assigned line. Hence, the capacity constraint on assembly line l requires that the total assembly time of the products, $g_{p(d)t}\ a_d \xi_{dlt}$ (where $g_{p(d)t}$ is the forecasted demand, a_d is the per unit assembly time, and decision variable ξ_{dlt} denotes the fraction of the demand assigned to the given line), plus the potential changeover times, $a^X u_{dlt}$ (where auxiliary variable u_{dlt} indicates if there is a changeover to design alternative d on the line), cannot exceed the fixed capacity q_l of the lines:

$$\sum_d \left(g_{p(d)t}\ a_d \xi_{dlt} + a^X u_{dlt}\right) \leq q_l$$

Assembly lines consist of multiple stations that execute different assembly processes, among which focus is given to high-precision adjustment processes necessary for setting the adjusted dimensions of the selected design alternatives. Each process k can be performed by a human operator, or alternatively, it can be automated for a given automation cost. Nevertheless, once a process is automated, it cannot be downgraded to a human process later. A combination of human and automated processes is also allowed on the same assembly line.

Each process is further characterized by its achievable adjustment precision b_{ltk}. The adjustment precision of human processes is a fixed value of b^H. On the contrary, the initial adjustment precision b_{l0k} of an automated process k, which may be insufficient to assemble the design alternatives with

Table 2: Notation (assembly resource configuration sub-problem).

Indices, sets	
t	Time period (index)
l	Assembly line (index)
r	Adjustment precision cost function breakpoint index
L	Set of assembly lines
T	Set of time periods
L^{new}	Set of potential new lines
Input parameters	
g_{pt}	Order amount for product p in period t [pcs.]
a_d	Processing time of design alternative d on assembly lines [s/pcs.]
a^X	Changeover time on assembly lines [s/pcs.]
q_l	Nominal capacity of (existing or potential new) line l [s]
b_{l0k}	Initial adjustment precision of process k of line l [$\pm$ mm]
$\underline{b}$	Possible best adjustment precision of assembly lines [$\pm$ mm]
b^H	Adjustment precision ability of the human operators [$\pm$ mm]
Π	Max. number of parallel lines for processing the same product [pcs.]
T^D	Useful life of assembly lines in the depreciation model [time periods]
C^{P0}_{lk}	Adjustment precision cost of line l in its initial state [$]
Decision variables	
x_{dlt}	Variable indicating that design alternative d is assigned to line l in period t
ξ_{dlt}	Fraction of the demand for design alternative d assigned to line l in period t
u_{dlt}	Variable indicating that design alternative d is reassigned to line l in period t
z_{lt}	Variable indicating that new line l is installed in period t
b_{ltk}	Adjustment precision of process k of line l in period t [$\pm$ mm]
v_{ltk}	Variable indicating that process k of line l is automated in period t

adjustment, can be upgraded to b_{ltk} with $b_{l0k} \geq b_{ltk} \geq \underline{b}$ by the enhancement of the automated equipment. It should be noted that $b^H < b_{ltk}$ is also allowed, which implies that automation with a substandard equipment may deteriorate the precision of the assembly process.

Then, a selected design alternative can be assigned to a line l if the adjustment precision of the line is at least as good as the precision required by that design alternative for every process k, i.e., $\overline{r}_{d,k} \geq b_{ltk} \quad \forall k, t$. The notation applied for the assembly resource configuration sub-problem is summarized in Table 2.

3.3. Production costs and depreciation model

The objective is minimizing the *total production costs*, which comprises costs related to parts manufacturing, assembly, and investments. *Manufacturing costs* are composed of the fixed, per unit base manufacturing cost c^{T0}_d

of the selected design alternative d and a tolerance cost, calculated as the sum of the costs of manufacturing the individual dimensions with the specified tolerances. Hence, the per unit manufacturing cost C_p^M of product p, with selected design alternative d and manufactured dimensions $M(d)$, can be calculated as:

$$C_p^M = c_d^{T0} + \sum_{\delta \in M(d)} C_\delta^T$$

The tolerance costs of the individual dimensions are approximated by convex piecewise linear functions for each dimension δ, specified with the breakpoints of the functions. The x coordinates $C_r^{T[x]}$ of the function are the tolerance values τ_δ, while the y coordinates $C_r^{T[y]}$ provide the costs of manufacturing dimension δ to a given tolerance τ_δ. Accordingly, the tolerance cost $C_\delta^T(\tau_\delta)$ can be calculated using the following formula:

$$C_\delta^T(\tau_\delta) = \max_{r \geq 2} \left(C_{r-1}^{T[y]} \frac{C_r^{T[x]} - \tau_\delta}{C_r^{T[x]} - C_{r-1}^{T[x]}} + C_r^{T[y]} \frac{\tau_\delta - C_{r-1}^{T[x]}}{C_r^{T[x]} - C_{r-1}^{T[x]}} \right)$$

Assembly costs are composed of the operation costs c_l^0 of the assembly line per units produced. In addition to that, a labor cost of c^H per unit is charged for the manual processes on the lines. Finally, each changeover on the lines is penalized with a changeover cost of c^X.

Further costs are related to *investments* into new or upgraded assembly equipment. New lines can be built for a base investment cost of c^L, which includes the installation of a manual assembly line. Processes on the existing or newly built lines can be automated for a given automation cost of c_k^A for each process k. Further, the precision of the automated processes can be upgraded, which is captured by a convex piecewise linear function, again given its breakpoints. Similarly to the tolerance cost function, values $C_{rk}^{P[x]}$ on the x axis provide the precision, while values $C_{rk}^{P[y]}$ on the y axis define the corresponding costs for process k. The investment costs related to a certain precision upgrade of process k can be calculated as the difference of equipment values realized in two subsequent periods, i.e., $C_{ltk}^P(b_{ltk}) - C_{ltk}^P(b_{l0k})$.

All investment costs—including the installation of new lines, upgrading the adjustment precision or the level of automation—are calculated by using a *linear depreciation model* with a useful life of T^D. The notation for cost components is summarized in Table 3.

Finally, the *total production cost* is calculated as the sum of the parts manufacturing cost, the assembly line operation cost, the assembly labor cost, the changeover cost, as well as the investment costs related to new line installation, upgrades in adjustment precision, and in the level of automation. When solving the problem, the solution that minimizes this cost is sought.

Minimizing the above complex cost function captures the problem of finding the best tradeoff between different approaches to reaching the desired product qualities. Strict precision requirements can be satisfied by manufacturing precision parts (which leads to high manufacturing costs) or by incorporating an appropriate adjustment mechanism in the product design (which comes with lower manufacturing but higher assembly costs). Likewise, the selection of human and automated assembly resources that can serve the precision requirements is a challenging problem. Moreover, the synergies between different products sharing the same assembly equipment must be exploited. Finally, it is emphasized that all cost components are expressed in monetary terms, and therefore can be summarized to constitute a single objective function, and hence, there is no need for considering complex multi-criteria optimization. Nevertheless, it must be ensured that the time horizon is long enough and demand forecasts are sufficiently reliable to capture a realistic demand volume for all products.

3.4. Assumptions

This section recapitulates the assumptions made in the above model, both during tolerance design and assembly resource configuration:

- The model focuses on the assembly of precision products, where the costs related to achieving the desired tolerances are crucial both in parts manufacturing and in assembly.
- Design requirements are expressed in terms of dimensional tolerance specifications on each dimension chain.
- There is at most one adjusted dimension in each dimension chain.
- Adjustment happens after the assembly of all related parts, and hence, it compensates the deviation of all dimensions in the chain (though, this assumption can be lifted with a minor extension of the model).
- In parts manufacturing, tolerance costs are captured by convex, piecewise linear functions assigned to individual dimensions, see, e.g., [42].

Table 3: Notation (costs).

Cost parameters	
c_d^{T0}	Base manufacturing cost of design alternative d [$]
c_l^0	Cost of operating line l for a unit time [$/s]
c^X	Cost of a changeover on assembly lines unit time [$/pcs.]
c^L	Cost of installing a new line [$/line]
c_k^A	Fix cost of automating process k [$/process]
c^H	Unit cost of human labor [$/min.]
$\left(C_r^{T[x]}, C_r^{T[y]}\right)$	Breakpoint $r \in R$ of the tolerance cost function [($\pm$ mm,$)]
$\left(C_{rk}^{P[x]}, C_{rk}^{P[y]}\right)$	Breakpoint $r \in R$ of the adjustment precision cost function of process k [($\pm$ mm,$)]
Cost function components	
C_{ltk}^P	Precision capability value of process k of line l in period t [$]
C_δ^T	Cost of manufacturing dimension δ to the selected tolerance [$]
C_p^M	Unit cost of manufacturing product p [$/pcs.]
C^M	Parts manufacturing cost [$]
C^L	Assembly lines operation cost [$]
C^I	Installation cost of new assembly lines [$]
C^P	Investment cost of upgrading the adj. prec. of assembly lines [$]
C^X	Changeover cost on assembly lines [$]
C^H	Human labor cost [$]
C^A	Automation cost [$]

- Likewise, the investment cost of automated machinery for a given assembly process can be described by a convex, piecewise linear function of the desired precision.

- Full interchangeability of parts is assumed, i.e., there are no defective items and no selective assembly is required.

- A sufficiently precise demand forecast is available for the products.

4. Solution approach

Two alternative but related solution approaches have been investigated and implemented to address the above defined problem:

- A monolithic *mixed-integer linear programming* (MILP) formulation, which is a declarative representation of the problem at hand, and which can be solved directly using commercial MILP solvers. These solvers use branch-and-bound search for solving the MILP formulation, which implies that the approach is *exact*, i.e., it constructs proven, *exact optimal* solutions if sufficient computational time is available. On the other hand, this approach can be unsuitable for very large problem instances.

- A *large neighborhood search* (LNS) algorithm based on the same MILP formulation, which combines the above branch-and-bound solution approach with local search. This combination is expected to scale up better to very large problem instances, however, like typical local search approaches, it cannot provide any guarantee on the quality of the solution found.

4.1. Monolithic MILP formulation

The above defined problem can be encoded in the form of a MILP as presented below. In addition to classical linear constraints, this formulation makes use of so-called *indicator constraints*, a modelling utility offered by various commercial MILP solvers including FICO Xpress or IBM CPLEX for expressing logical combinations of constraints. An indicator constraint of the form $x \Rightarrow c$, where x is a binary variable and c is a linear constraint, states that if x takes a value of 1, then constraint c must hold. From the conventional mathematical programming toolkit, one could use so-called *big-M constraints* to express the same logical relations, however, indicator constraints result in a more readable model, more robust behavior, and improved computational efficiency by allowing the MILP solver to calculate tight coefficients for variable x in the constraint, even during the solution process [43]. Hence, the overall problem formulation is as follows.

Minimize

$$C^M + C^H + C^L + C^X + C^I + C^P + C^A \tag{1}$$

subject to

$$\sum_{\delta \in \Delta} \tau_\delta \leq \varphi_\Delta \qquad \forall\, \Delta \in D_d^- \tag{2}$$

$$\sum_{\delta \in \Delta} \tau_\delta \; - \; \varrho_\Delta + r_\Delta \leq \varphi_\Delta \qquad \forall\, d, k, \Delta \in D_{d,k}^+ \tag{3}$$

$$r_\Delta \leq \varphi_\Delta \qquad \forall\, d, k, \Delta \in \bigcup_k D_{d,k}^+ \tag{4}$$

$$C_1^{T[x]} \leq \tau_\delta \leq C_R^{T[x]} \qquad \forall\, \delta \tag{5}$$

$$\overline{r}_{d,k} \leq r_\Delta \qquad \forall\, d, k, \Delta \in D^+_{d,k} \tag{6}$$

$$C_\delta^T \geq \left(C_{r-1}^{T[y]} \frac{C_r^{T[x]} - \tau_\delta}{C_r^{T[x]} - C_{r-1}^{T[x]}} + C_r^{T[y]} \frac{\tau_\delta - C_{r-1}^{T[x]}}{C_r^{T[x]} - C_{r-1}^{T[x]}} \right) \qquad \forall\, \delta, r \geq 2 \tag{7}$$

$$\sum_{d \,\mid\, p(d)=p} y_d = 1 \qquad \forall\, p \tag{8}$$

$$\sum_l \xi_{dlt} = y_d \qquad \forall\, d, t: \; g_{p(d)t} > 0 \tag{9}$$

$$\sum_l x_{dlt} \leq \Pi y_d \qquad \forall\, d, t: \; g_{p(d)t} > 0 \tag{10}$$

$$\xi_{dlt} \leq x_{dlt} \qquad \forall\, d, l, t: \; g_{p(d)t} > 0 \tag{11}$$

$$x_{dlt} \leq z_{lt} \qquad \forall\, d, l \in L^{\text{new}}, t \tag{12}$$

$$(1 - v_{ltk}) \;\Rightarrow\; \left(\overline{r}_{d,k} \geq b^H x_{dlt}\right) \qquad \forall\, d, l, t, k \tag{13}$$

$$v_{ltk} \;\Rightarrow\; (b_{l0}(1 - x_{dlt}) + \overline{r}_{d,k} \geq b_{ltk}) \qquad \forall\, d, l, t, k \tag{14}$$

$$v_{ltk} \geq v_{l(t-1)k} \qquad \forall l, t, k \tag{15}$$

$$u_{dlt} \geq x_{dlt} - x_{dl(t-1)} \qquad \forall\, d, l, t \tag{16}$$

$$\sum_d \left(g_{p(d)t} \; a_d \xi_{dlt} + a^X u_{dlt}\right) \leq q_l \qquad \forall\, l, t \tag{17}$$

$$z_{lt} \leq z_{l(t-1)} \qquad \forall\, l \in L^{\text{new}}, t \tag{18}$$

$$\underline{b} \leq b_{ltk} \leq b_{l(t-1)} \qquad \forall\, l, t \tag{19}$$

$$C_{ltk}^P \geq \left(C_{r-1k}^{P[y]} \frac{C_{rk}^{P[x]} - b_{ltk}}{C_{rk}^{P[x]} - C_{r-1k}^{P[x]}} + C_{rk}^{P[y]} \frac{b_{ltk} - C_{r-1k}^{P[x]}}{C_{rk}^{P[x]} - C_{r-1k}^{P[x]}} \right) \qquad \forall\, l, k, r \geq 2 \tag{20}$$

$$C_{ltk}^P \geq C_{l(t-1)k}^P \qquad \forall\, l, t, k \tag{21}$$

$$C^P = \sum_{ltk} \frac{1}{T^D} (C_{ltk}^P - C_{lk}^{P0}) \tag{22}$$

$$C^H_{ltk} \geq c^H \left(\sum_d \left(g_{p(d)t}\ a_d \xi_{dlt} + a^X u_{dlt} \right) \ - \ (1 - v_{ltk}) q_l \right) \quad \forall\, l, t \tag{23}$$

$$C^H = \sum_{ltk} C^H_{ltk} \tag{24}$$

$$y_d \ \Rightarrow \ \left(C^M_p \geq c^{T0}_d + \sum_{\delta \in M(d)} C^T_\delta \right) \qquad \forall\, d, p = p(d) \tag{25}$$

$$C^M = \sum_{pt} (C^M_p \ g_{pt}) \tag{26}$$

$$C^L = \sum_{dlt} (c^0_l \ g_{p(d)t} \ a_d \xi_{dlt}) \tag{27}$$

$$C^X = c^X \sum_{dlt} u_{dlt} \tag{28}$$

$$C^I = \frac{c^L}{T^D} \sum_{lt} z_{lt} \tag{29}$$

$$C^A = \sum_{ltk} (v_{ltk} - v_{l(t-1)k}) \frac{c^A_k \min(T^D, T - t + 1)}{T^D} \tag{30}$$

$$x_{dlt},\ y_d,\ u_{dlt},\ z_{lt},\ v_{ltk} \ \in \ \{0, 1\} \qquad \forall\, d, l, t \tag{31}$$

$$\xi_{dlt}, b_{ltk} \geq \ 0 \qquad \forall\, d, l, t \tag{32}$$

The objective (1) stands for minimizing the total cost, composed of the parts manufacturing cost C^M, the assembly labor cost C^H, the assembly line operation cost C^L, the changeover cost C^X, the new line installation cost C^I, the lines' precision upgrade costs C^P, and the assembly line automation cost C^A.

The tolerance assignment sub-problem is addressed in constraints (2)-(7), whose solution is relevant only for the design alternatives selected for production. Constraint (2) requires that the stacked tolerance along any dimension chain without adjustment amounts to at most the design specification for the given chain. In contrast, for chains with adjustment, the adjustment mechanism can compensate an error equal to the adjustment range of the mechanism minus its adjustment precision (3). At the same time, the adjustment precision itself cannot be looser than the tolerance specification of the chain (4). Bounds for the individual tolerances must be in line with technological limits (5). The adjustment precision requirement of a design for any process k is at least as strict as the precision of the adjustments performed during that process (6). Finally, the cost related to the tolerance on an individual dimension is determined by the piecewise linear cost function $C_{\delta}^{T}(\tau_{\delta})$ (7).

The second part of the MILP focuses on the selection of the design alternatives, the assignment of the design alternatives to assembly lines, and the configuration of these lines. Equality (8) states that for each product, exactly one design alternative must be selected for production. Constraints (9) and (10) ensure that the complete demand for the selected design alternatives is distributed among at most Π assembly lines in each time period where there is nonzero demand for the given product. Moreover, a fraction of the demand for design alternative d can be assigned to a line l only if d is assigned to l in the given period (11). Products can be assigned to new lines only if the lines are already installed (12). Furthermore, the adjustment precision of the line must be at least as good as the precision required by the design alternative, both in case of human (13) and automated processes (14). Automated processes cannot be downgraded to manual (15).

Constraint (16) relates the changeover variables to the assignment variables. The capacity constraint (17) states that the sum of processing times and changeover times on an assembly line, either existing or newly built, cannot exceed the line capacity. Investments related to new line installation (18) and adjustment precision upgrade (19) are performed in a given period of time, and they cannot be undone later.

Inequality (20) calculates the adjustment precision costs of the individual lines in each time period. These adjustment precision costs increase monotonously over time (21). From these values, the total adjustment precision upgrade cost is computed by equality (22), by subtracting the cost of the initial lines from the extended lines, also accounting for depreciation. Similarly, inequality (23) calculates the per period per line labor cost, and equation (24) sums these values to compute the total labor cost.

Constraint (25) calculates the unit manufacturing cost of a product as a sum of the base manufacturing cost and the total tolerance cost on the manufactured dimensions for the selected design alternative. Then, equations (26)-(30) calculate the manufacturing, the line operation, the changeover, the new line installation, as well as the line automation costs, respectively. Finally, constraints (31) and (32) define the variables as binary or non-negative continuous.

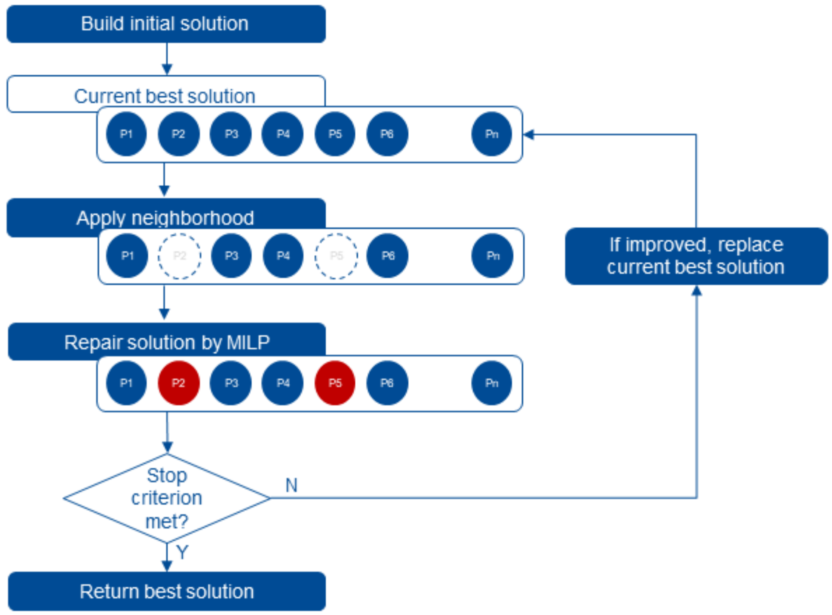

Figure 2: Flowchart of the LNS approach.

4.2. Large neighborhood search algorithm

In order to solve very large instances of the above problem, an LNS matheuristic solution approach was implemented, which combines mathematical programming for solving the above MILP representation with local search techniques. The application of LNS to the particular problem required adapting the approach in both of its two main steps: the construction of the initial solution, and the iterative exploration of the local neighborhood.

For constructing an initial solution, a so-called *Russian Doll* approach has been applied: a hierarchy of embedded time intervals is defined as $T_1 \subset T_2 \subset ... \subset T_K = T$ with $T_k = [1, k\Delta T]$. In step k of the algorithm, the optimal solution for time interval T_k is computed subject to the constraint that the head of the solution corresponding to interval T_{k-1} matches the earlier solution for T_{k-1}. During all experiments, the value of $\Delta T = 5$ and a time limit of 300 seconds was used.

In the iterative step of LNS, an improved solution is looked for by re-solving the original problem with the added constraints that, for a subset of the products ($N - 2$ products in the current implementation), the selection of the design alternative and the assignment to assembly lines cannot be modified. For the remaining products (2 products in the implementation), both the selection of the design alternative and the assignment to assembly lines is reconsidered by solving the restricted version of the original MILP model to optimality (or stopping the MILP solver at a given time limit, 300 seconds in the experiments). The algorithm replaces the previous best solution by the current iterative solution if and only if the current solution is an improvement over the previous best solution. LNS terminates when all neighborhoods have been searched or it reaches a pre-defined time limit, 3600 seconds in the experiments reported. The flowchart of the algorithm is presented in Figure 2.

It is noted that various alternative algorithms have been implemented and evaluated both for constructing the initial solution and for the iterative step, but a decision has been made for the above procedures due to their simplicity and efficiency, as it will be shown below in the experimental evaluation.

5. Experimental evaluation

In this case study, the viability of the proposed method in multi-product high-precision assembly is investigated from the viewpoints of computational efficiency and industrial effectiveness. The following subsections first introduce the sample product and the production environment. Then, the computational efficiency of the two solution approaches, monolithic MILP and LNS, is investigated and compared. Finally, a real industrial case study is presented in detail.

5.1. Production environment

The experimental evaluation shown below in Sections 5.2 and 5.3 are based on sample data originating from the industry, and involves a product family that contains eight different products, three of which are new. While the designs of the five existing products are given and cannot be modified, the designs for the three new products can be selected from six structurally different design alternatives, and their tolerance allocation should also be optimized. Each target product in the family consists of several parts and has two design specifications, (φ_{f2} and φ_{f3}), which are guided by dimensional tolerances between the parts. An overview of the product structure is shown in Figure 3. Structural design alternatives differ in several ways: the parts called *house* and *cover* can be integrated (d_{INT}) or separated (d_{SEP}); and it is possible to incorporate or omit the adjustment mechanisms between the *house* and the *cover* (only for d_{SEP}), or between the *house* and the *stopper* (both d_{INT} and d_{SEP}). The combinations of these choices define the six structural design alternatives.

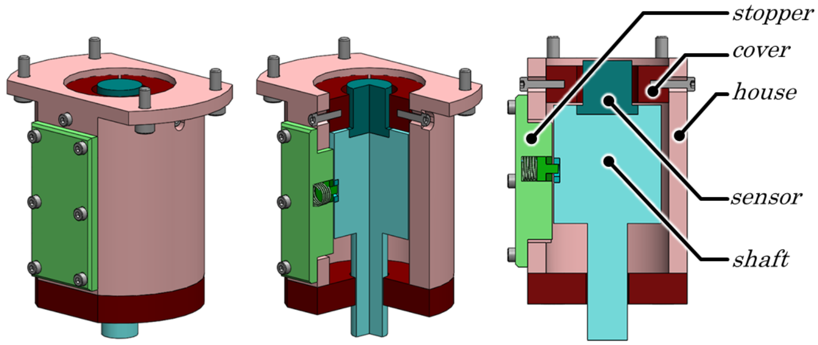

Figure 3: One structural design alternative, d_{SEP} with two adjustment mechanisms, for the sample product.

The process plan consists of three target processes, and it is presented in Figure 4 for each of the design alternatives. *Process 1* assembles the *shaft* and the *sensor*, without the option of adjustment. Hence, the resulting assembly tolerance is defined by the stacked tolerance of the parts, $\delta_{A1} = \delta_K + \delta_J$.

Process 2 involves the assembly of the *house* and the *shaft*, and it determines one of the final design specifications, φ_{f2}. Without adjustment, the design specification must be satisfied by the stacked tolerance on the involved individual dimensions, i.e., $\varphi_{f2} \geq \delta_D + \delta_{A1} = \delta_D + \delta_K + \delta_J$ for d_{INT} and $\varphi_{f2} \geq \delta_F + \delta_G + \delta_{A1} = \delta_F + \delta_G + \delta_K + \delta_J$ for d_{SEP}.

For both possible adjustment mechanisms in the sample product, the adjustment range of the mechanism is larger than the stacked tolerance on the dimensions in the same dimension chain. Therefore, in case of adjustment in *Process 2*, the design specification must be satisfied directly by the adjustment precision of the machine or a skilled worker who performs the given process, i.e., $\varphi_{f2} \geq r_{\Delta 2}$.

Finally, *Process 3*, which assembles the *stopper* sub-assembly to the *house*, is responsible for meeting the design specification φ_{f3}, with or without adjustment. Observe that the two dimension chains related to φ_{f2} and φ_{f3} share common dimensions, e.g., A1.

For the assessment of production costs, 15 periods demand forecast data is considered as an input. To satisfy the production volume, there are three

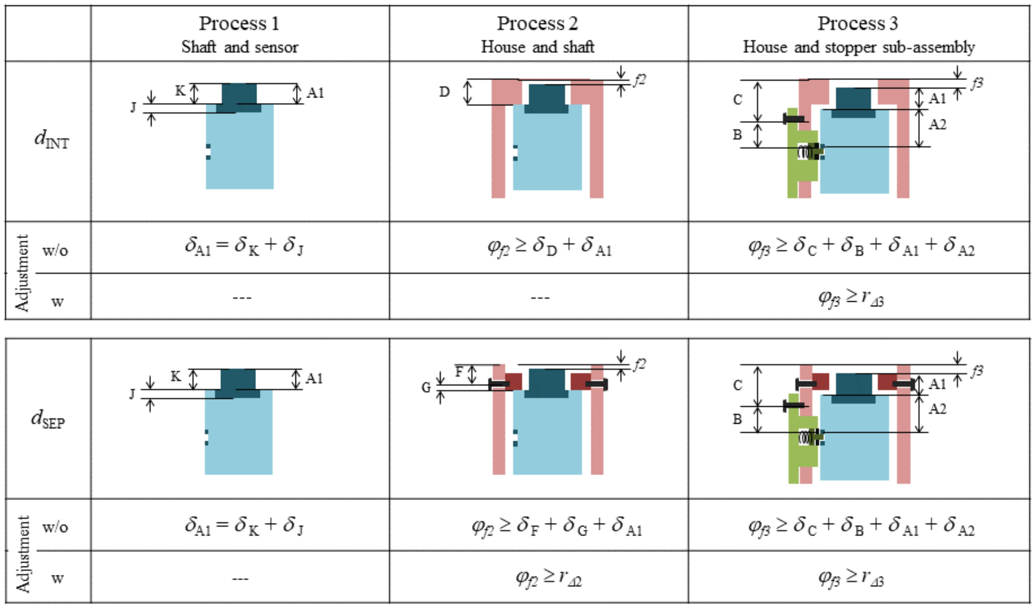

Figure 4: Assembly process of the sample product and the dimension chains related to each process. Design alternatives are defined by different combinations of the integrated (d_{INT}) or the separated (d_{SEP}) structure of the *house* and the *cover*, and designs with (w) or without (w/o) adjustment mechanisms related to each dimension chain.

existing assembly lines, with or without adjustment equipment and different adjustment precisions. As the production volume increases, two additional assembly lines have to be built by the end of the horizon.

5.2. Assessment of computational efficiency

In order to compare the computational efficiency of the two proposed mathematical models on a large set of instances with controllable sizes, artificially generated problem instances were derived from the above original dataset by applying random perturbations. The problem size was controlled by two parameters: the length of the planning horizon $|T| \in \{10, 20, 40\}$ and the number of products $|P| \in \{4, 8, 16\}$, with half of the products being new, while the other half existing products. The number of structural design alternatives per products was fixed to 6, with 4 alternatives requiring adjustment. These settings resulted in $3 \cdot 3 = 9$ combinations of the parameters, and for each combination, five different random instances were generated, leading to 45 problem instances in total. Random perturbations were applied to the production volumes, maintaining realistic demand profiles, e.g., increasing production volumes for new products and decreasing volumes for some old products (Figure 5). The realistic nature of the random instances was maintained by keeping the structure, i.e., the dimension chains of the design alternatives unchanged. They are results of design engineering work that could be hardly captured by random instance generators.

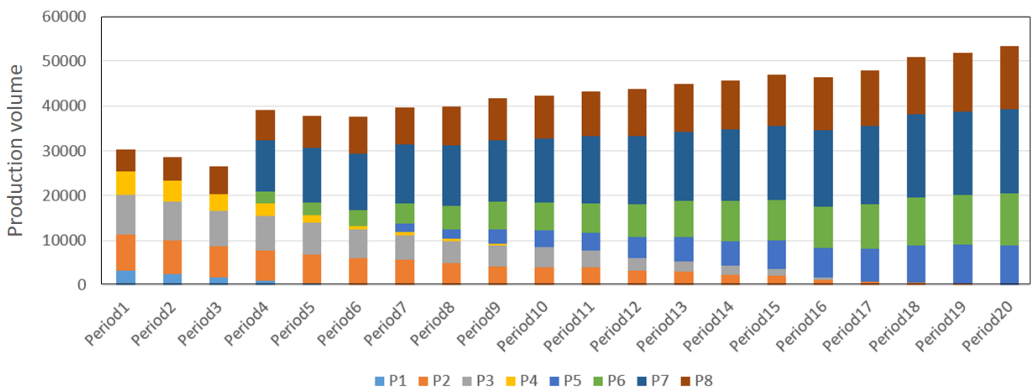

Figure 5: Demand volumes in a randomly generated instance with 8 products.

All the reported experiments were run with a time limit of one hour for both the LNS and the monolithic MILP approaches. The experiments were run on a virtual (cloud) computer with Linux operating system, using the FICO Xpress 8.2 commercial MILP solver.

The results of the experiments are displayed in Table 4, where each row contains combined results for the 5 instances for a given $|P|$ and $|T|$. Separately for the MILP and the LNS approaches, the table shows the number of instances out of 5 where a feasible solution was found (column *Sol*), the number of instances solved to optimality (column *Opt*), the average and maximum optimality gaps (columns *Avg. gap* and *Max. gap*), and the average computation time (*Avg. time*). For each instance and solution approach, the optimality gap was calculated as (UB-LB)/LB, where UB is the upper bound (solution value) found by the given approach, and LB is the lower bound computed by MILP. Since MILP could not find any solution for some of the largest instances (though, it could always compute a lower bound), gaps are computed only for the instances with a feasible solution with the given approach. Finally, it is noted that the meaning of optimality in the table is slightly different for MILP and LNS: while MILP is an exact solution approach that can actually prove the optimality of a solution (corresponding to a gap of 0%), LNS alone cannot yield such a proof; instead, it can be observed a posteriori that the LNS solution matches the value of the MILP lower bound.

The results show that the smallest instances ($|P|$=4) were easily solvable by the MILP to proven optimality with a single exception. In contrast, for most of the medium-sized instances ($|P|$=8), MILP terminated with a sub-optimal solution after one hour of computation, with average gaps of 1.8%-6.6% and a maximum gap of 8.9%. The largest instances were indeed challenging for MILP: one third of the instances could not be solved at all, and even for the solvable instances, MILP terminated with considerable gaps (average gaps of 3.8%-37.9%, and a gap of 69.5% for one of the instances).

Two main observations can be made on the performance of the LNS. First, on the 19 instances with known optimum (all instances with $|P|$=4, and some with $|P|$=8) one can observe that LNS results in close-to-optimal solutions with an average error of 0.0034%, which is an extremely good performance from a matheuristic approach. For the 26 more challenging instances without a known optimal solution, LNS clearly outperformed the exact MILP approach. It found reasonable solutions for 5 instances where MILP could not find a solution at all. Even when MILP could find a feasible solution, LNS improved that solution by 4.5% on average, and by 56.6% in an extreme case. There was a single instance where MILP could find a somewhat better solution than LNS, by 0.16%. Moreover, LNS typically required lower computation times than MILP both for the small and the large instances.

5.3. Assessment of industrial effectiveness

The experiments presented here were carried out on a real industrial product family, similar in its size, complexity, and the involved assembly processes to the product presented in detail above. This product family consists of six products, two of which are new. There are two assembly processes, each with an independent final tolerance specification. The specifications differ for each product, and the range of specifications starts from 0.15 mm, which is also the best adjustment precision that a human operator can achieve. These specifications must be reached using parts with tolerances on individual dimensions starting from 0.05 mm. There are design alternatives with

Table 4: Comparison of the MILP and the LNS solution approaches. The best values over the different approaches are highlighted in bold.

$\|P\|$	$\|T\|$	MILP					LNS				
		Sol	Opt	Avg. gap	Max. gap	Avg. time	Sol	Opt	Avg. gap	Max. gap	Avg. time
4	10	**5**	**5**	0.0%	0.0%	**25**	**5**	**5**	0.0%	0.0%	30
	20	**5**	**4**	0.3%	1.6%	780	**5**	**4**	**0.3%**	**1.3%**	**65**
	40	**5**	**5**	0.0%	0.0%	551	**5**	**5**	0.0%	0.0%	**102**
8	10	**5**	0	6.6%	8.9%	3 600	**5**	0	**5.5%**	**6.9%**	**168**
	20	**5**	**2**	3.6%	8.3%	2 960	**5**	0	**3.3%**	**8.0%**	**385**
	40	**5**	**3**	1.8%	5.0%	1 950	**5**	2	**1.5%**	**4.2%**	**1 004**
16	10	**5**	0	3.8%	4.6%	3 600	**5**	0	**3.5%**	**4.2%**	**1 616**
	20	3	0	11.8%	22.7%	3 600	**5**	0	**6.5%**	**15.2%**	**1 998**
	40	2	0	37.9%	69.5%	3 600	**5**	0	**8.2%**	**15.2%**	**2 670**

and without adjustment mechanisms for each process, resulting in four structurally different design alternatives for each new product. This is similar to *Processes 2* and *3* in the design structure d_{SEP} in Figure 4.

The case study investigated the cross-effects of design alternative selection and the level of automation in the assembly system. Specifically, three scenarios were studied, with only machine (M), only human (H) and mixed human and machine resources (H/M), respectively. The human adjustment precision of 0.15 mm is sufficiently high to assemble any of the design alternatives manually, and the processes are not automated at the beginning of the planning horizon in any of the scenarios. This can easily lead to higher automation costs in M and H/M scenarios. The main findings of the case study are summarized in Figure 6 and Table 5, where the optimal solutions for the three scenarios are compared regarding their costs. Also, Table 6 shows the final automation status and the required precision of all production lines and processes in the three scenarios.

The overall costs are the highest in the *only machine* case, due to the high investment costs related to automation and precision upgrade: its cost is 15.56% higher than in the *only human* case, which is the baseline scenario and the current industrial practice. Total production costs are the lowest in the most generic H/M case, with a cost reduction of 13.47% compared to the baseline. This is the result of maximizing the investment efficiency by harmonizing product designs and the level of automation, i.e., using low-cost automated assembly for less precise products while assigning qualified

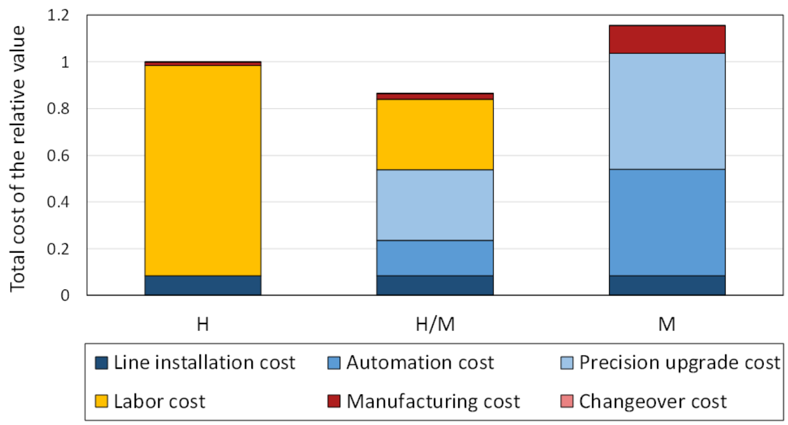

Figure 6: Results of the case study: comparison of optimal solutions under different resource selection options (H: only human, H/M: mixed human and machine resources, M: only machine).

workforce to high-precision products.

The design alternatives selected for the two new products are shown in Table 7. In the conventional H case with highly qualified assembly workforce, design alternatives with adjustment mechanisms are selected; this way, loose tolerances are required on the parts, and hence, manufacturing costs can be kept low. On the other hand, in the M case, the precision upgrade cost of the machines is very high, and therefore, a proposal is adopted to reduce the investments by selecting design alternatives without adjustments, yet, at the price of higher manufacturing costs for more precise parts. In the H/M case,

Table 5: Results of the case study: cost structure in percentage of the total cost of the baseline H scenario.

	H	H/M	M
Line installation cost	8.27%	8.27%	8.27%
Automation cost	0.00%	15.27%	45.81%
Precision upgrade cost	0.00%	30.23%	49.60%
Labor cost	90.21%	30.30%	0.00%
Manufacturing cost	1.48%	2.39%	11.84%
Changeover cost	0.03%	0.07%	0.03%
Total	100.00%	86.53%	115.56%

Table 6: Final automation status and adjustment precision of each process for the different scenarios (H: operated by human worker with a fixed precision of 0.15 mm; M: operated by an automated machine with precision displayed in parentheses).

		H	H/M	M
Line 1	Process 1	H(0.15)	H(0.15)	M(0.15)
	Process 2	H(0.15)	H(0.15)	M(0.20)
Line 2	Process 1	H(0.15)	M(0.30)	M(0.30)
	Process 2	H(0.15)	M(0.30)	M(0.30)
Line 3	Process 1	H(0.15)	M(0.15)	M(0.15)
	Process 2	H(0.15)	M(0.20)	M(0.20)
Line 4	Process 1	H(0.15)	H(0.15)	M(0.56)
	Process 2	H(0.15)	H(0.15)	M(0.56)
Line 5	Process 1	H(0.15)	H(0.15)	M(0.56)
	Process 2	H(0.15)	M(0.56)	M(0.56)

Table 7: Design alternatives selected for the two new products in each scenario, for each process (w: with adjustment mechanism, w/o: without adjustment mechanism).

		H	H/M	M
New product 1	Process 1	w	w	w/o
	Process 2	w	w	w/o
New product 2	Process 1	w	w	w/o
	Process 2	w	w/o	w/o

the best compromise between the above extremities is derived with partial automation, and accordingly, with adjustment in a part of the dimension chains. This shows that joint optimization of the tolerance design and the assembly resource configuration enables finding the best compromise between the costs of manufacturing precise parts, applying qualified workforce for assembly, and investing into new assembly equipment to minimize the total production cost.

6. Discussion on practical application potential

The proposed methods were implemented in a decision support system consisting of three key modules. The *optimizer* implements the mathematical model and the two proposed solution approaches. This module supports the quantitative investigation of different scenarios. Numerical results can be passed to other modules for further processing. The purpose of the *Web UI* visualization module is to display the results of the optimizer and support their analysis from all relevant aspects with the help of an interactive user interface presenting various types of charts, in an easy-to-understand format.

Finally, the design module was developed to facilitate the design workflow with the automation of design alternative generation and input data preparation, as well as result visualization, all linked to a CAD environment. In this module, a master CAD model was prepared for this particular product family. The master model contains each existing structural design alternative type (i.e., CAD assembly models that differ in geometry beyond dimension or tolerance values), with the corresponding adjustable dimensional parameters (nominal and tolerance values). When establishing a new design alternative, the designer can simply select the proper model configuration, fill in the assigned parameter values and create a new product (design alternative) instance.

As the CAD model of the new design alternative, including the tolerance model, is built up automatically, the dimension chain and tolerance parameters of the new product variant can be exported and forwarded to the optimizer. This eliminates the manual preparation of tolerance design related data, and thus reduces the possibility of faulty input for the optimizer. Furthermore, the design module is capable of reading the solution computed by the optimizer, and therefore, the resulting tolerance allocation can be displayed on the original CAD model by actual geometry modification. This provides the designer with an effective tool to ensure the feasibility and correctness of the design. The connection between the three modules is depicted in Figure 7.

7. Conclusions and future work

Despite the various approaches proposed in literature and realized in the industrial practice for each phase of the product development process, finding the best tradeoff between product design and process/production plan efficiency is becoming more challenging than ever in multi-product assembly. This paper proposed a novel method for integrating and optimizing product design and process/production planning in order to maximize the investment efficiency and reduce the overall production cost. As an early step towards this goal, the paper focused on tolerances crucial for high-precision assembled products. Accordingly, a novel optimization problem was formulated that combines tolerance design, as the relevant sub-problem of product design, with assembly resource configuration, as the corresponding sub-problem

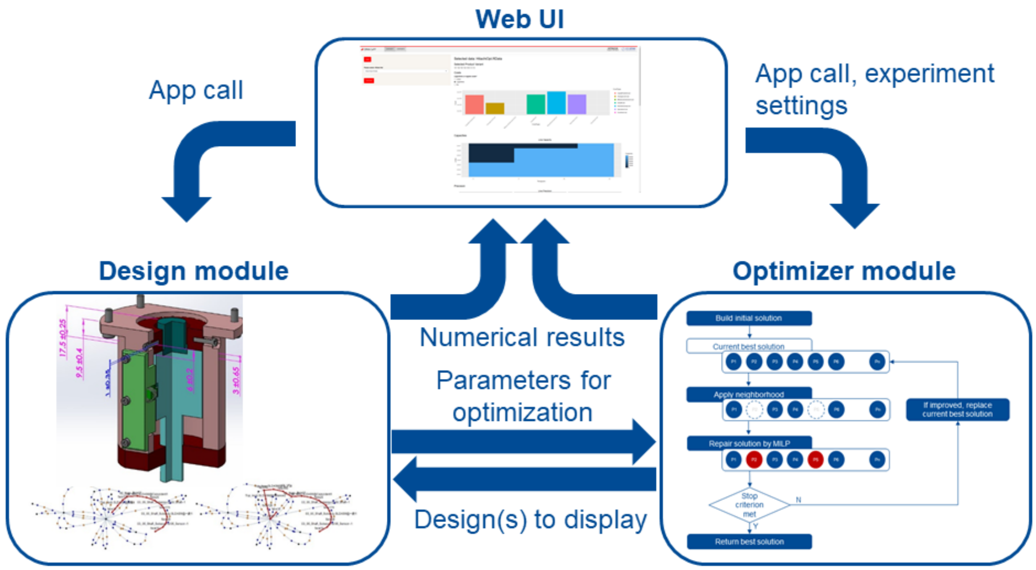

Figure 7: System architecture of the proposed decision support system.

in process and production planning.

The overall problem was formulated as a MILP, and an LNS matheuristic solution method was proposed for solving large, industrially relevant instances. The computational efficiency of LNS was demonstrated on a set of instances based on industrial data. Furthermore, the relevance of the proposed problem to industry was shown in a case study that focused on evaluating different design alternatives and different combinations of human and machine resources. It was confirmed that an appropriate combination of different resources and a corresponding selection of design alternatives can minimize the total production costs.

As a direction for future research, it is necessary to evaluate the robustness of the approach to the fluctuation of the long-term production plan and the various cost functions that define the input of the model. A relevant extension of the model will cover the case of selective assembly instead of full interchangeability of the parts. The implementation of the proposed decision support system with the optimization engine, a web-based dashboard, and 3D CAD integration is in progress, in order to promote future utilization in the industry.

Acknowledgments

The Hungarian authors of the paper were supported by the ED_18-2-2018-0006 grant on "Research on prime exploitation of the potential provided by the industrial digitalisation" and GINOP-2.3.2-15-2016-00002 grant on an "Industry 4.0 research and innovation center of excellence". A. Kovács acknowledges the support of the János Bolyai Research Fellowship.

[1] S. J. Hu, J. Ko, L. Weyand, H. ElMaraghy, T. Lien, Y. Koren, H. Bley, G. Chryssolouris, N. Nasr, M. Shpitalni, Assembly system design and operations for product variety, CIRP Annals–Manufacturing Technology 60 (2) (2011) 715–733.

[2] J. Daaboul, C. Da Cunha, A. Bernard, F. Laroche, Design for mass customization: Product variety vs. process variety, CIRP Annals–Manufacturing Technology 60 (1) (2011) 169–174.

[3] H. Wang, X. Zhu, H. Wang, S. J. Hu, Z. Lin, G. Chen, Multi-objective optimization of product variety and manufacturing complexity in mixed-model assembly systems, Journal of Manufacturing Systems 30 (1) (2011) 16–27.

[4] A.-L. Andersen, T. D. Brunoe, K. Nielsen, C. Rösiö, Towards a generic design method for reconfigurable manufacturing systems: Analysis and synthesis of current design methods and evaluation of supportive tools, Journal of Manufacturing Systems 42 (2017) 179–195.

[5] E. Lutters, F. J. van Houten, A. Bernard, E. Mermoz, C. S. Schutte, Tools and techniques for product design, CIRP Annals–Manufacturing Technology 63 (2) (2014) 607–630.

[6] T. Tolio, D. Ceglarek, H. ElMaraghy, A. Fischer, S. Hu, L. Laperrière, S. T. Newman, J. Váncza, SPECIES—Co-evolution of products, processes and production systems, CIRP Annals–Manufacturing Technology 59 (2) (2010) 672–693.

[7] H. ElMaraghy, G. Schuh, W. ElMaraghy, F. Piller, P. Schönsleben, M. Tseng, A. Bernard, Product variety management, CIRP Annals–Manufacturing Technology 62 (2) (2013) 629–652.

[8] G. Boothroyd, P. Dewhurst, Product design for manufacture and assembly, Boothroyd Dewhurst, Inc., 1989.

[9] J. He, Tolerancing for manufacturing via cost minimization, International Journal of Machine Tools and Manufacture 31 (4) (1991) 455–470.

[10] D. Tsutsumi, D. Gyulai, A. Kovács, B. Tipary, Y. Ueno, Y. Nonaka, L. Monostori, Towards joint optimization of product design, process planning and production planning in multi-product assembly, CIRP Annals–Manufacturing Technology 67 (1) (2018) 441–446.

[11] F. Hansen, Konstruktionssystematic Grundlagen für eine allgemeine Konstruktionslehre, VEB Verlag Technik, Berlin, 1965.

[12] J. Jänsch, H. Birkhofer, The development of the guideline VDI 2221–The change of direction, in: Proceedings DESIGN 2006, the 9th International Design Conference, 2006, pp. 45–52.

[13] G. Q. Huang, Design for X: Concurrent Engineering Imperatives, Springer, 1996.

[14] S. Miyakawa, T. Ohashi, The Hitachi assemblability evaluation method (AEM), Proceedings of the International Conference on Product Design for Assembly (1986).

[15] B. Miles, K. Swift, Working together, Boothroyd Dewhurst, Inc., 1992.

[16] A. Redford, J. Chal, Design for Assembly, Principle and Practice, McGrawHill Book Company Europe, 1994.

[17] G. Molcho, Y. Zipori, R. Schneor, O. Rosen, D. Goldstein, M. Shpitalni, Computer aided manufacturability analysis: Closing the knowledge gap between the designer and the manufacturer, CIRP Annals–Manufacturing Technology 57 (1) (2008) 153–158.

[18] A. Tornow, R. Graubohm, F. Dietrich, K. Dröder, Design automation for battery system variants of electric vehicles with integrated product and process evaluation, Procedia CIRP 50 (2016) 424–429.

[19] D. Whitney, Mechanical Assemblies: Their Design, Manufacture, and Role in Product Development, Oxford University Press, 2004.

[20] B. Heling, A. Aschenbrenner, M. Walter, S. Wartzack, On connected tolerances in statistical tolerance-cost-optimization of assemblies with interrelated dimension chains, Procedia CIRP 43 (2016) 262–267.

[21] P. K. Singh, S. C. Jain, P. K. Jain, Advanced optimal tolerance design of mechanical assemblies with interrelated dimension chains and process precision limits, Computers in Industry 56 (2) (2005) 179–194.

[22] F. Schlather, V. Hoesl, F. Oefele, M. F. Zaeh, Tolerance analysis of compliant, feature-based sheet metal structures for fixtureless assembly, Journal of Manufacturing Systems 49 (2018) 25–35.

[23] L. Andolfatto, F. Thiébaut, C. Lartigue, M. Douilly, Quality- and cost-driven assembly technique selection and geometrical tolerance allocation for mechanical structure assembly, Journal of Manufacturing Systems 33 (1) (2014) 103–115.

[24] J. G. Li, Y. X. Yao, P. Wang, Assembly accuracy prediction based on CAD model, The International Journal of Advanced Manufacturing Technology 75 (5) (2014) 825–832.

[25] J.-Y. Chiang, T.-R. Tsai, Y. Lio, W. Lu, D. Shi, An integrated approach for the optimization of tolerance design and quality cost, Computers & Industrial Engineering 87 (2015) 186–192.

[26] S. Xu, J. Keyser, Statistical geometric computation on tolerances for dimensioning, Computer-Aided Design 70 (2016) 193–201.

[27] Y. Zhong, Y. Qin, M. Huang, W. Lu, W. Gao, Y. Du, Automatically generating assembly tolerance types with an ontology-based approach, Computer-Aided Design 45 (11) (2013) 1253–1275.

[28] S. Ghandi, E. Masehian, Review and taxonomies of assembly and disassembly path planning problems and approaches, Computer-Aided Design 67–68 (2015) 58–86.

[29] X. Chen, S. Gao, Y. Yang, S. Zhang, Multi-level assembly model for top-down design of mechanical products, Computer-Aided Design 44 (10) (2012) 1033–1048.

[30] S. Rachuri, Y.-H. Han, S. Foufou, S. C. Feng, U. Roy, F. Wang, R. D. Sriram, K. W. Lyons, A model for capturing product assembly information, Journal of Computing and Information Science in Engineering 6 (1) (2006) 11–21.

[31] Z. Ding, B. Hon, Constraints analysis and evaluation of manual assembly, CIRP Annals–Manufacturing Technology 62 (1) (2013) 1–4.

[32] G. Fantoni, M. Santochi, G. Dini, K. Tracht, B. Scholz-Reiter, J. Fleischer, T. K. Lien, G. Seliger, G. Reinhart, J. Franke, H. N. Hansen, A. Verl, Grasping devices and methods in automated production processes, CIRP Annals–Manufacturing Technology 63 (2) (2014) 679–701.

[33] B. W. Pearce, K. Antani, L. Mears, K. Funk, M. E. Mayorga, M. E. Kurz, An effective integer program for a general assembly line balancing problem with parallel workers and additional assignment restrictions, Journal of Manufacturing Systems 50 (2019) 180–192.

[34] A. Gomez, J. Rios, F. Mas, A. Vizan, Method and software application to assist in the conceptual design of aircraft final assembly lines, Journal of Manufacturing Systems 40 (2016) 37–53.

[35] X. Fiorentini, I. Gambino, V.-C. Liang, S. Rachuri, M. Mani, C. Bock, An ontology for assembly representation, National Institute of Standards and Technology (2007).

[36] Y. Pochet, L. A. Wolsey, Production Planning by Mixed Integer Programming, Springer, 2010.

[37] V. Maniezzo, T. Stützle, S. Voß, Matheuristics–Hybridizing Metaheuristics and Mathematical Programming, Springer, 2010.

[38] R. K. Ahuja, O. Ergun, J. B. Orlin, A. P. Punnen, A survey of very large-scale neighborhood search techniques, Discrete Applied Mathematics 123 (1-3) (2002) 75102.

[39] S. Mouthuy, P. V. Hentenryck, Y. Deville, Constraint-based very large-scale neighborhood search, Constraints 17 (1) (2012) 87–122.

[40] P. Laborie, D. Godard, Self-adapting large neighborhood search: Application to single-mode scheduling problems, in: Proceedings of

MISTA'07, Multidisciplinary International Scheduling Conference: Theory & Applications, 2007, pp. 276–284.

[41] P. M. Thompson, Local search algorithms for vehicle routing and other combinatorial problems, Ph.D. thesis, Massachusetts Institute of Technology (1988).

[42] D. G. Ullman, The mechanical design process (4th Edition), McGraw-Hill Edcuation New York, 2009.

[43] P. Belotti, P. Bonami, M. Fischetti, A. Lodi, M. Monaci, A. Nogales-Gómez, D. Salvagnin, On handling indicator constraints in mixed integer programming, Computational Optimization and Applications 65 (3) (2016) 545–566.