



HAL
open science

Standardized Vocabularies for Assembly Systems Modelling and Automation Alternatives Description

Anas Salmi, Pierre David, Eric Blanco, Joshua D. Summers

► **To cite this version:**

Anas Salmi, Pierre David, Eric Blanco, Joshua D. Summers. Standardized Vocabularies for Assembly Systems Modelling and Automation Alternatives Description. ASME 2016 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference - IDETC/CIE 2016, American Society of Mechanical Engineering (ASME), Aug 2016, Charlotte, NC, United States. hal-01326201

HAL Id: hal-01326201

<https://hal.science/hal-01326201>

Submitted on 29 Oct 2016

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

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

DETC2016-60461

**STANDARDIZED VOCABULARIES FOR ASSEMBLY SYSTEMS MODELLING
AND AUTOMATION ALTERNATIVES DESCRIPTION**

Anas Salmi¹
Graduate Student
anas.salmi@grenoble-inp.fr

Eric Blanco¹
Associate Professor
eric.blanco@grenoble-inp.fr

¹Univ. Grenoble Alpes, G-SCOP, F-38000
Grenoble, France
CNRS, G-SCOP, F-38000 Grenoble, France

Pierre David¹
Assistant Professor
pierre.david@grenoble-inp.fr

Joshua D. Summers²
Professor
jsummer@clemson.edu

²Department of Mechanical Engineering
Clemson University
Clemson, SC 29634-092

ABSTRACT

The aim of this research is to support assembly lines designers in conceiving new processes with optimal automation levels selection. Several alternatives with various automation options may exist. Graphic representations and analyses of the different designs are needed. The finality is to offer a quick, exhaustive, and reliable way of modelling alternatives based on a given product design. In this sense we propose a new assembly tasks vocabulary to be combined to an existing lower layer vocabulary of elementary motions and a graphic modelling language. These developments deal with an existing automation decision approach as an extension allowing to overcome identified gaps and to ease its implementation and computerization. The proposal facilitates assembly systems alternatives generation with automation options consideration based on an initial representation. The generated alternatives are then subject to further analyses with regard to automation criteria and performance indicators considering planned production targets.

1. INTRODUCTION

Deciding about assembly process automation during the early phase of the system design is a complex problem. This complexity is basically caused by the multiple criteria involved in the decision: parts features, their impact on assembly

complexity, cost, quality, volume to produce, ergonomics, etc. The question is to decide where to automate or not, and to what extent for an optimal process configuration with regard to the criteria and planned production targets. Before reaching the stage of the final decision, multiple analyses and comparisons of alternatives should be conducted. A first concept of the system must be available as a basis or starting point of the analyses. This concept could be ideally graphical so that it can be easily analyzed and discussed between the stakeholders involved in the decision. It should also enable providing different alternatives of resources allocation with possible degrees of automation, generally labelled Levels of Automation (LoA) in this research area. The LoA options are treated using a LoA scale going from completely manual (low LoA) to high automated or robotized systems (high LoA).

The paper is organized as follows: In section 2, a literature background is presented. In section 3, a focus is performed on the related existing automation decision method to which the paper contributes. The different lags are then identified. In section 4, the proposal is presented. The way to use the vocabulary is then tackled in the automation decision issue in section 5. A simple example is given in section 6. Finally, the paper is concluded in section 7 with openings and future works proposals.

2. LITERATURE IN AUTOMATION DECISION

The purpose of this research is to select the right automation level for a future assembly production. Yet, the literature in the field of automation is predominantly technical and seeks principally to improve processes performance, technology, autonomy, adaptability, and productivity. It is less devoted to find the most appropriate system configuration to a given future production considering manufacturer's preferences and strategy (Hill, 1999). In fact, the literature focusing on deciding the suitable automation level is not abundant and a need to an efficient method to support the decision has been identified (Lindström & Winroth, 2010).

Existing literature methods tailored to support automation decision making were reviewed in (Salmi, David, Blanco, & Summers, 2015.b). Eleven decision methods were found and analyzed. Multiple critics were addressed to these methods revealed as not fully useful to be applied for early phase automation decision making. In fact, after requirements definition, no method is revealed to be completely satisfactory or fulfilling all of the requirements. These requirements are basically related to: the applicability to an early phase of new systems design, the objectivity of the method applicability, the analytic reasoning according to the method's granularity level, partial automation possibilities offering, the profitability consideration, the manufacturer's preferences involving, and the traceability of the decision. This led to a new decision approach proposal in (Salmi, David, Blanco, & Summers, 2015.b) addressing these different requirements. The proposed method which consequently consists in the most recent and promising one represents a basis for this work. Our purpose here is to concretize its implementation and ease its computerization after gaps identification. These gaps are presented in next subsection with a thorough review of the method.

3. THE AUTOMATION DECISION APPROACH: A REVIEW AND GAPS IDENTIFICATION

The purpose of this section is to describe the literature LoA decision method representing the basis of this work via a review and gaps identification.

3.1 THE APPROACH REVIEW

The method defined in (Salmi, David, Blanco, & Summers, 2015.b) is presented in Figure 1 by a simplified outline. The method is defined in such a way so that the whole requirements previously presented in section 2 can be satisfied. To do so, as shown in Figure 1, the idea is to support systems alternatives models definition based on a generic initial design model. The initial model has to be defined based on the product design. The systems alternatives are generated considering this initial model and other information: planned production information (plant environment, cadence, manufacturer's strategy and preferences, quality level, etc) and other possible standardized

data or historical information according to the experience or previous projects. Automation options that can be allowed according to the different gathered inputs are selected in every alternative with resources LoAs combinations and dimensioning: duplication if the cadence should be increased while the resource LoA is not productive enough or assignment to multiple tasks to maximize resources occupation (Salmi, David, Blanco, & Summers, 2015.a). Once possible alternatives available, evaluations can be conducted with regard to multiple dimensions (e.g. cycle time, cost, and ergonomics). By exploring the different possible alternatives, a best process with regard to considered dimensions can be found.

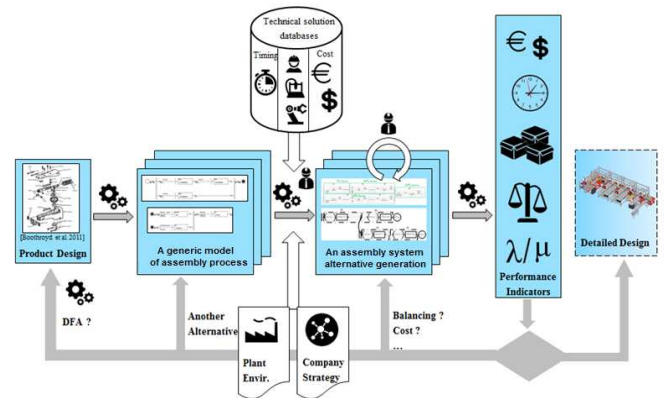


Figure 1: The automation decision making method outline (reproduced from (Salmi, David, Blanco, & Summers, 2015.b))

The focus of this research is on the second and third bloc of the outline of Figure 1, respectively the initial generic model definition and the generation of systems alternatives.

The second bloc consists in the initial generic model definition. This is based on the product design and parts features (representing then the first bloc of the diagram of Figure 1) involving complexity to assemble and resulting time estimates. The method defines this initial model using assembly motions and their sequencing using a standardized vocabulary of elementary motions from (Renu, Mocko, & Koneru, 2013). This vocabulary, inspired from "Methods Time Measurements" (MTM) (Maynard & Stegemerten, 1948) and "Design For Assembly" (DFA) methods (Boothroyd, Dewhurst, & Knight, 2011), is associated in the decision method to a dedicated modeling language labeled ASML defined in (Salmi, David, Summers, & Blanco, 2014). The combination ASML and vocabulary allows assembly process modeling by elementary motions definition based on a given assembly sequence of parts assembly. Annex Figure 1 shows an assembly process modeling using in ASML (Annex Figure 1.b) for a product design assembly sequence shown in Annex Figure 1.a.

The next step in the decision procedure, located in bloc 3 of the outline, is an iterative procedure of alternatives of processes generation. This is basically conducted by resources allocation to modeled motions of the initial generic model with

consideration of the previously discussed inputs if the planned production. As an example, Annex Figure 2 shows an alternative of resources allocations of the model of Annex Figure 1 considering a unique resource labeled R1. Lags are then mandatory following ASML rules defined in (Salmi, David, Summers, & Blanco, 2014). Such a scenario of resources assignment can be conducted in the case of a very low volume to be produced so that a unique resource can be satisfactory.

For the automation perspective, to every allocated resources a LoA is associated. These LoA selections should be performed with respect to LoA criteria (plant environment, company strategy, quality level, ergonomics, etc...), manufacturer’s preferences and best practices, and a pre-balancing performing to allow an enough productive system according to required volume to be produced. The consideration of criteria can allow, impose, or forbid certain LoAs choices to some assembly motions. For example, manual can be forbidden for motions related to welding, or robotic can be imposed for painting or automotive windshield installing throughout the whole assembly process. Possible process resources automation configurations are then subject to deeper analyses considering performance indicators (time, cost, volume, etc...). The saturation of alternatives generation and evaluations by the loop described in figure 1 should lead to an optimal alternative keeping: the best deal with regard to multiple criteria. This optimal one goes in the detailed design process. If no satisfactory alternative is found, a feedback is then provided to the product designers to try to improve its design and the easiness or cost to assemble it using approaches such as DFA. It can be seen that a key element of the method is the vocabulary of motions allowing modeling the initial model. A focus is performed in next sub-section on this vocabulary and possibility of extending it.

3.2 THE VOCABULARY OF MOTIONS: A POSSIBLE EXTENSION

As previously discussed, the method uses a vocabulary of elementary assembly motions from (Renu, Mocko, & Koneru, 2013). The vocabulary is associated to ASML to model processes. After multiple analyses of this vocabulary, we think it can be extended by adding some new motions to be more exhaustive. Table 1 shows a complete list containing the original vocabulary with some additional motions we propose to add for more standardization and to cover more operations. These extra motions are highlighted in bold in Table 1. The grey colored cells in Table 1 contain the core motions representing value added motions. The remaining motions represent unavoidable extra motions that should be done to perform the value added ones. With regard to core motion, the extra-motions can be in upstream (for example preparing the parts to be assembled, such as “identify” or “get”) or in downstream (releasing the parts, such as “place”, “move”, or

“lay”). The obtained list of Figure 1 can still be extendable by eventually adding other motions that can be particular to a specific context assembly field, such as PCB (Printed Circuit Boards), semi-conductors assembly, or other particular assembly fields.

Table 1: The standardized vocabulary of assembly motions
(Adapted from (Renu, Mocko, & Koneru, 2013))

Align	Exchange	Inspect	Press	Scan
Apply	Get	Lay	Read	Solder
Clean	Handstart	Move	Remove	Tighten
Connect	Hit[Hammer]	Open	Restock	Unscrew
Disengage	Identify	Operate	Restrict	Wait
Engage	Insert	Place	Rivet	Walk

3.3 THE IDENTIFIED METHOD’S GAPS

The placement of the decision approach, just in between the finalization of the product design and the early phase of assembly system design, is challenging from the utilized data point of view, basically for data availability issues, product design maturity, and important decisions not yet taken and that should be fixed. One of the important inputs for assembly system design is the product to be assembled design and parts features. With a CAD model analysis, the obtained data on the assembly are expressed through alignments of components, insertion between one and another, or elementary actions as getting, insertion, align, tighten, etc. On the other hand, the needed information for assembly sequence description or for assembly resources allocation is different in terms of granularity level. Even if the vocabulary described in section 3.2 can be of interest for the automation possibilities analyses and LoA criteria involving (e.g. motions analyses with regard to ergonomics, repetitive motions as good signs for possible automation, quality, etc...), a too detailed granularity level does not seem to be the most suitable for the step of initial generic model building for multiple reasons. These reasons are discussed through the following sub-sections followed by a proposed solution in next section to address the different gaps.

3.3.1. THE GENERICITY OF THE INITIAL ASSEMBLY PROCESS MODEL

As described in ASML modelling language (Salmi, David, Summers, & Blanco, 2014) and in the automation decision method (Salmi, David, Blanco, & Summers, 2015.b), the first representation of the assembly process is performed using a standardized vocabulary, the one shown in Table 1. Annex Figure 1.b shows an example of an ASML modelled process for a product design of Annex Figure 1. To represent such an assembly process based on product design using this kind of motions can be a long work with a high risk of error. Also, this first ASML definition should be independent from the assembly technology so that automation alternatives with different LoAs

consideration can be later considered. This cannot be possible using the discussed vocabulary of motions. For example, an operator needs to “get” a tool (LoA=2) while it is already mounted for an automatic system (LoA=3). A second example can be the description in motions of the screwing task of two parts (P1) and (P2) by a resource R shown in the appendix table. It can be observed that the description depends on the selected LoA for the 4 different LoAs. The number of motions is also significant for a simple task as screwing. Such a low level with as much detail does not seem to be the most appropriate to handle assembly system design and automation decision making problem.

3.3.2. DEPENDENCIES BETWEEN ELEMENTARY MOTIONS FROM RESOURCE POINT OF VIEW

Another issue is to be able to assign, in the initial model assumed here to be generic, separate resources (e.g. workstations) to the different process activities. As explained, resources can be then assigned to assembly motions in the initial model using the possibility that offers the ASML language (Salmi, David, Summers, & Blanco, 2014).. Yet, the way to systematically generate alternatives of resources assignment to motions cannot be easily performed. We realize that dependencies exist between elementary assembly motions and a resource executing them, whatever is its automation level. For example, when there is an ‘Align’ of parts made by a resource, an ‘Insert’ action that follows should be executed by the same resource. This makes the problem complex when managing independent motions that should be grouped by an executing resource, and when considering a high number of motions and different possibilities of LoAs.

3.3.3. MOTIONS COMPATIBILITY

To solve the motions dependencies discussed in 3.3.2, one can try to allocate as much motions as possible to the same resource. Yet, some sequential motions can be incompatible between them because of their natures or resource selected LoA. They can belong to different assembly activities (some of the motion belongs to screwing, others to soldering, riveting, etc.). A same resource can be consequently technically not able to execute those heterogeneous activities if assigned to them, especially when inflexible (dedicated machines, specific tools required, etc). By contrast a more flexible resource as a manual can be able to execute them sequentially. Meaningful group of elementary motions can be useful to define before the resources assignment step. This meaning is expressed by the main activity or purpose of a group formed by complementary or dependent motions that can be deduced from core motions figuring in that group (grey colored cells of Table 1 motions).

3.3.4. ANALYZING PRODUCTION RATE

To conduct the LoA decision, every motion of the assembly process is time estimated regarding different possible LoAs and the parts design features. The resources time estimation is mandatory to design assembly system so that it can reach the required production rate. Every resource should then run at the production cadence. The time estimation on the level of every assigned resource considering its selected automation level should satisfy the planned production takt-time. The assembly sequence representation must then enable the description of timely measurable assembly actions once an LoA is associated to given actions. When using low layer granularity level motions vocabulary, an assembly motion on the product is described with a different sequence for dissimilar LoAs. As a result, the assignments of motions to resources for selected LoAs considering required productivity target will be also different. The motions to be assigned to a resource will depend then on the selected LoA. This increases the problem resolution complexity especially for a systematic generation of alternatives. Addressing these difficulties using a low layer granularity seems to be unmanageable.

3.3.5. THE COMBINATORY EXPLOSION OF ASSEMBLY SYSTEMS ALTERNATIVES

In the described automation decision method, assembly systems alternatives with automation options are generated with consideration of elementary motions with resources assignments and automation possibilities combination analyses. It can be obvious that the total number of assembly elementary motions (Table 1) used to model the initial process for a complete real product assembly can be huge. As consequences, iterating all the possible alternatives of resources assignments to these elementary motions with all feasible automation options to scan can be unfeasible because of the combinatory explosion to perform, with additionally the previously discussed feasibility constraints to manage. The low layer vocabulary of assembly motions is interesting regarding its offered details, associated time estimates and analytic possibilities of automation criteria that can be considered (ergonomic motions, repetitively of actions, etc), but revealed seriously penalizing to be implemented and applied for the problem resolution. The challenge is to keep the advantages offered by the low layer vocabulary. At a same time, the multiple identified gaps have to be addressed in order to make the approach easily implementable and to be able to propose computerizable resolution issues with feasible computational time in such NP-hard problems where resolution time is generally exponential. We propose in next section a solution addressing the different encountered difficulties.

4. A VOCABULARY OF ASSEMBLY TASKS PROPOSAL

The aim of this section is to propose a solution to address the different identified weaknesses. The solution we are proposing is to build and use a standardized vocabulary of a higher layer granularity: the one of tasks. Then, to preserve the discussed benefits of the original low layer vocabulary, the proposed vocabulary should match with this original one. These features are detailed through this section.

4.1 THE VOCABULARY OF ASSEMBLY TASKS

In this proposal, we define a task as an activity of the assembly sequence that can be labelled in the same way, whatever the LoA employed. A task has same or equivalent results when performed by a resource independently from its LoA (e.g. screwing). We define an assembly tasks vocabulary following the same principles of MTM (Maynard & Stegemerten, 1948), DFA (Boothroyd, Dewhurst, & Knight, 2011), and others for assembly time analyses and standardizations (Miller, Griese, Peterson, Summers, & Mocko, 2012) (Renu, Mocko, & Koneru, 2013). In these works a low granularity vocabulary proposed for analytic studies, basically time estimation issues. For the actual issue of assembly systems design and automation decision issues, as previously discussed, a higher layer vocabulary can be more appropriate, at least during the first steps to ease process modelling and resources assignments. The use of the lower layer can be still interesting too later in the system design for the further analyses (time estimates, repetitive motions identification, ergonomics, complex motions, etc..). This invites to create a link tasks to motions to preserve their advantages.

The vocabulary of assembly tasks includes all conventional tasks that can be encountered in assembly manufactories, consisting mostly in available assembly techniques. Based on the literature and on industrial observations, we identified 20 assembly tasks. The proposed list of tasks is given in Table 2. It can be observed that major tasks match with the previously discussed core motions. This preliminary list can be refined or extended. The focus of this paper is to show that this vocabulary will help making automation decision more organized, more standardized, and more rigorously applicable within the methodology presented in 3.1. It is tailored for assembly systems modelling and automation deciding issues. To allow lower layer analyses, the vocabulary of tasks is linked to the previously presented low layer vocabulary. This is tackled in next section 4.2.

Table 2: Proposed Vocabulary of Assembly Tasks

Bolting	Clinching	Inspecting	Scanning
Bracket fixing	Clipping	Pinning	Screwing
Brazing	Feeding	Placing	Setting-up
Clamping	Gluing	Press fitting	Snap fitting
Cleaning	Hooping	Riveting	Welding

4.2 THE LINK OF ASSEMBLY TASKS TO ASSEMBLY MOTIONS

An important point to support the LoA decision process is to be able to move from one granularity level to another (higher to lower and vice-versa). The proposed high layer vocabulary of tasks matches with the level of elementary motions. Every task is then defined by a kinematic decomposition as a succession of elementary motions for the different LoA levels. This allows converting the representation to a more detailed one when it is needed (e.g. ergonomics analysis, time estimation). Depending on the task and resource automation level, the required motions can differ. For some levels, the task can be impossible to perform. For example: if a task necessitates an automated tool to be performed (such as for soldering), a low LoA defined by the only use of the worker's physical strength (full manual, LoA=1) is not valid. Also, for a same task and a same selected automation level, it can be possible to have multiple representations in motions if types in this technique can exist (tools or machines types, used technology or energy, parts features, materials, etc...). Here we talk about involved parameters to enter for every couple (task, LoA) in order to have the appropriate task type and consequently the corresponding representation in motions. For example for the task "riveting" for a selected automation level equals to 2 (manual with automated tool), different types of rivets (dimensions) can exist, or different types of riveting tools or their energy can exit (pressure, temperature, etc..). Some of the tasks can be used to be repeated 2, 3, or n times, e.g. a cleaning or welding tasks, can be single pass or multiple pass. In this case the number of passes can be managed as a parameter in the motions decomposition table or otherwise as a multiplier of the resulting motions number, and consequently the resulting time estimates. Another example is shown in the Appendix table representing possible decompositions of the "Screwing" task to the required motions to perform it for a 2 parts assembly "P1" and "P2". Every part Pi represents an object with attributes consisting in parts features impacting the time estimates and the complexity to handle or assemble the part, such as the thickness, surface, symmetry, easiness to handle, etc... The decompositions may be usable in both directions: to deduce the detailed motions for a given task, but also to identify tasks corresponding to a detailed sequence of elementary motions.

It can be seen that the proposed vocabulary offers a higher layer of abstraction, but allows at a same time to keep the benefits of the lower vocabulary layer thanks to the possibility of converting every modelled task to its corresponding motions according to appropriate tasks parameters. In the next section 5, the way to use this vocabulary in assembly modelling and automation decision is more concretely presented followed by an application example in section 6.

5. THE USE OF VOCABULARIES IN ASSEMBLY MODELLING AND AUTOMATION DECISION

As mentioned, the proposed standardized vocabulary is tailored to support early phase assembly system design. This includes modelling of the assembly, analyses of automation possibilities, and optimization by alternatives evaluations and comparisons with regard to criteria and performance indicators. We follow the approach defined in (Salmi, David, Blanco, & Summers, 2015.b) described in section 3. To tackle the different identified weaknesses, we propose a little adjustment of the approach: we recommend using the new vocabulary of tasks presented in section 4.1 for the generic initial representation modelling. This is detailed in section 5.1. Then, the alternatives can be generated as explained in section 5.2 through 3 steps thanks to the link tasks to motions previously proposed in section 4.2.

5.1. A GENERIC INITIAL ASSEMBLY TASKS MODEL

The first graphic representation of the process is defined in tasks using ASML and the proposed tasks vocabulary instead of motions. Such a representation corresponds to one of the multiple possible sequences to assemble a product (Homem De Morello & Sanderson, 1991). Based on this selected sequence and the product design, tasks are defined based on how the parts are supposed to be assembled such as: screwed, soldered, riveted, snap fitted, etc. As in ASML, motions are graphically modelled in rectangles, we recommend using different features for tasks representation (such as hexagons) so that the models, the motions-based and tasks-based models, can be differentiated and confusion avoided. For automation technical feasibility and to support selecting a relevant sequence, it is recommended to schedule as much as possible similar tasks in succession to make tasks grouping easier (detailed in section 4.2). Defining this first ASML model considering the succession of similar tasks and with respect to the sequence of parts assembly can be performed using an AND/OR graph (Homem De Morello & Sanderson, 1991). We propose figuring the assembly techniques (tasks of Table 2) on every arc in the AND/OR graph to ease this process. For example, if according to the AND/OR graph, we have: screw(P1,P2), then: solder(P1+P2,P3) OR screw(P1+P2,P4). It is preferable to schedule the assembly so that the two screwing will be in succession. In this case this will lead to the following schedule: screw(P1,P2), screw(P1+P2,P4), and finally solder(P1+P2,P3).

5.2. ASSEMBLY SYSTEMS ALTERNATIVES GENERATION

Based on the tasks-based ASML model, alternatives of assembly systems can be generated. An alternative results from resources allocating to ASML tasks with different LoA options selecting. The assignments have to be performed through 3 steps as follows:

5.2.1. STEP 1: TASK LOA SELECTION AND MOTIONS DEDUCTION

Because of the possibility to decompose tasks to associated motions once an LoA is selected to a given task as described in section 3.2, it can be possible to convert a tasks-based model to a lower layer motions-based model. This aims at tasks time estimating in order to allow correct resources allocation to satisfy the required cadence as explained in Step 2. But also, motions representation can be useful to detect phenomena that can be good signs to automation, such as repetitive motions, or involve other criteria such as ergonomics (e.g. handling heavy parts) or security aspects (e.g. warm parts assembly or unhealthy environments).

5.2.2. STEP 2: TASKS TIME ESTIMATES DETERMINATION FOR SELECTED LOA

For every task, when converted to motions, a time estimate can be calculated considering the motions architecture: a sum of motions time estimates in the case of sequential, the max of the values if parallel (Salmi, David, Blanco, & Summers, 2015.a). This is enabled by the fact that every elementary motion is time estimated depending on the LoAs and the task parameter that can involve multiple features: part complexity, surface, number of times to perform the motion, etc. A table of tasks time estimates in the different selectable LoAs and appropriate parameters according to the given assembly application can be useful to be performed first in order to decrease the computation time when dealing with several alternatives generation instead of using heavy complete time standards databases.

5.2.3. STEP 3: RESOURCES ASSIGNING TO TASKS

Once the tasks time estimates are obtainable in the different LoAs thanks to the conversion to motions, resources assigning to tasks can be performed. The higher layer can then be regained: the one of tasks. The resources assignment is performed according to productivity requirement and to obtained tasks time estimates for selected LoAs. As the process cadence is obviously given by the slowest resource of the production process, we identify 3 possible cases for optimal resources assignment:

- **A resource to be assigned to a unique task**

A first scenario is to obtain, for a selected LoA to considered tasks in the process, a time estimate approximately equal to the required takt time. This means that a resource of selected LoA will be enough productive compared to the required cadence. In this case the resource is well dimensioned. It is yet recommended to keep a certain margin to prevent resources failures especially for high production volumes by resource duplicating following the second scenario principles.

- **Multiple duplicated resources for a same task**

A second scenario is to obtain, for a selected LoA to considered tasks in the process, a higher value of time estimate compared to the required takt time. This means that a resource of selected LoA will be slow if assigned to this task compared to required cadence. In this case n duplicated resources of a same type are necessary to achieve the cadence target. The factor n by which the same resource is duplicated will multiply the task cadence n times until reaching the required production cadence.

- **A unique resource for multiple tasks**

The last possibility is to obtain for a selected LoA, a lower time estimate value compared to required planned production takt time. This means that a resource of the selected LoA will be too fast compared to the required cadence if assigned to only that task. This resource should be moderated for multiple objectives: maximizing the resource workload, minimizing the total number of resources, and consequently minimizing production costs. It is then obviously recommended that every resource performs as much as possible of tasks. Yet, this can be feasible only when concerned tasks are technically compatible. The compatibility between tasks can be related to resources or to tasks natures:

- ***Compatibility with regard to resources***

The concept of resources flexibility is here involved. We mean by resource flexibility the ability for a given LoA to handle different tasks with different natures. For example, for a 4 LoA scale, a worker using only his physical strength (LoA=1), using a tool (LoA=2), or similarly an industrial robot (LoA=4) are generally supposed to be flexible. Consequently, more than a task can be sequentially performed by a same selected flexible resource if the cadence requirement is still fulfilled. This is less commonly used to be applicable for inflexible resources, such as a dedicated automatic machine (LoA=3). Such a resource is not designed to execute tasks of completely different natures.

- ***Compatibility with regard to tasks***

In this case, compatibility is more related to the tasks. Tasks are generally compatible if they are of the same nature or if they use similar technique. For tasks natures' compatibility, a compatibility matrix including all the given assembly process modelled tasks can be useful to be build. This matrix will represent for every assembly task, the compatibility with regard to all the other tasks. To be compatible, tasks can be simply of a same type, such as screwing. Or they can be similar from required motions point of view, such as screwing and bolting. Two tasks of a same nature can be incompatible considering the parts features to assemble, such as their thickness, weight, etc. For example, screwing very thin parts can be incompatible with screwing thick parts (tools issue). These compatibilities can be more related to the given application itself or can depend on the method user analysis, reasoning, or own preferences.

To handle both compatibilities we propose a compatibility matrix taking into account both of compatibilities: with regard to resources and to tasks natures. This matrix should show the compatibilities of each task with regard to all the other model's tasks in the different possible resources LoAs. The matrix will then be a 3 dimensional matrix $M_{n \times n \times m}$ where n is the number of tasks in the assembly model and m the number of possible automation levels in the used LoA scale.

6. EXAMPLE

As an application example, we reconsider the previous example of product assembly shown in Annex Figure 1. The assembly process was modelled using motions in Annex Figure 1.b for the corresponding product design assembly shown in Annex Figure 1.a. Then resources should be assigned to these motions with automation options possibilities, with all relevant combinations generation and analysis. One assignment example was shown in Annex Figure 2 defined intuitively using a unique resource. Yet, this consists in only one of the several possible ways of system alternatives that can be defined. To scan all possible alternatives, alternatives should be generated based on Annex Figure 1.b following the original method. This model contains 12 motions. For a 4 LoA scale, this gives 16777216 possible alternatives (4^{12}) of tasks LoA combinations, and consequently of resources possible LoAs. Most of these combinations are meaningless and unfeasible because of the different reasons previously explained in section 3.3 (e.g. dependencies between motions from resource point of view).

If we model the process following the proposed approach of standardized vocabulary, we obtain following Step 1 of section 5.2.1, only 3 tasks as shown in Annex Figure 3. The number of possible LoA combinations decreases then to only 64 for the same example and a 4 LoA scale. This confirms the significant reduction of the combinatory explosion. Considering this LoA scale, a compatibility matrix of tasks $M_{3 \times 3 \times 4}$ (36 elements) should be here filled versus a $M_{12 \times 12 \times 4}$ (576 elements) of the previous motions management approach. Moreover, motions are still deducible once LoAs are selected for tasks. We show in Annex Figure 4 one possible combination with tasks and selected resources and corresponding motions representation.

The proposed configuration in Annex Figure 4 is as follows: Resource R1 selected as a manual resource (LoA=1) is executing tasks T1. The resource is duplicated 3 times to achieve the assumed planned production rate. A second resource R2 of a LoA=4 (robotic) is assigned to task T2. Finally, task T3 is executed by an automatic dedicated machine (LoA=4). Motions of every task in selected LoA are shown inside the blocks. In this example, Tasks T2 and T3 can be considered as compatible (both of them are screwing). They can be eventually allocated to a unique resource if the this can be possible considering obtainable tasks time estimates based on corresponding motions estimates (Step 2 in section 5.2.2) compared to the required cadence (Step 3 in section 5.2.3).

7. CONCLUSION & FUTURE WORKS

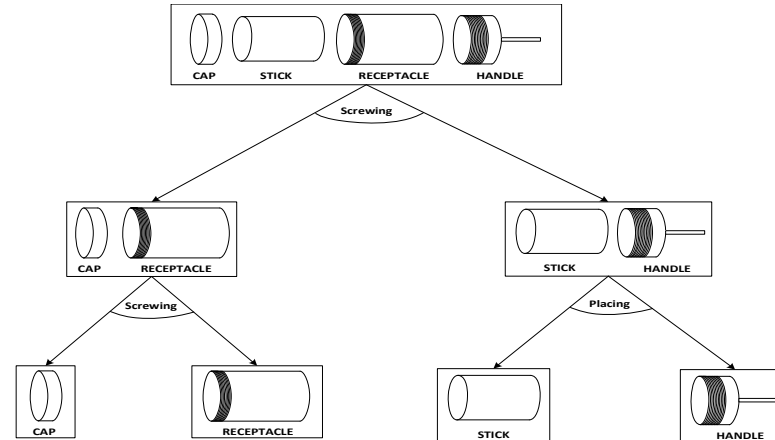
In this work a high layer vocabulary of assembly tasks is proposed. The proposal aims to support assembly system design and deals with a literature automation decision method. A solution is proposed to handle multiple identified gaps and to ease the systematic use and implementation of the decision method. The proposed vocabulary seeks to provide a more appropriate granularity level to describe the element of the assembly sequence. This enhances the creation of the first generic ASML model that should not depend on the resource LoA to use. Obtaining a lower layer representation of motions for selected LoA is still possible for specific analyses, such as for repetitive motions identifying, other signs to automation as ergonomic ones (handling of heavy parts), or tasks time estimating. This is facilitated by the link tasks to motions which allow a possible quick conversion of the tasks-based model to a motions-based one. Tasks time estimates obtaining enables efficient resources allocation considering automation options with respect of productivity targets. Managing tasks rather than motions hides the issue of dependencies of motions from executing resource point of view implicitly managed by tasks described in motion for the various LoAs. It also reduces considerably the combinatory explosion of system alternatives where the number of elements to manage is much lower. The approach is also flexible and generic because of the standardized vocabularies and the standardized modelling language ASML. The list of standardized vocabulary of tasks, also the one of motions, can be eventually more extended by new tasks definition and their decomposition to motions.

REFERENCES

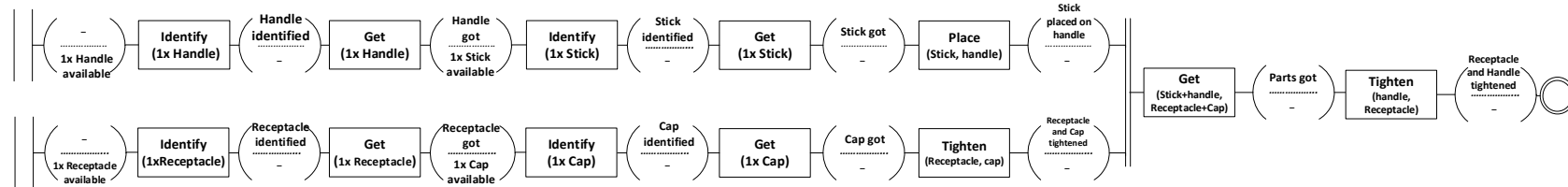
- Boothroyd, G., Dewhurst, P., & Knight, W. (2011). *Product Design for Manufacture and Assembly – Third Edition*. Boca Raton, FL: CRC Press; Taylor & Francis Group.
- Hill, T. (1999). *Manufacturing Strategy: Text and Cases, 3rd edition*. Homewood, IL: McGraw-Hill.
- Homem De Morello, L., & Sanderson, A. (1991). Representation of Mechanical Assembly Sequences. *IEEE Transactions on Robotics and Automation*, 7(2), 221-227.
- Lindström, V., & Winroth, M. (2010). Aligning manufacturing strategy and levels of automation: A case study. *Journal of Engineering and Technology Management*, 27(3-4), 148-159.
- Maynard, H., & Stegemerten, G. (1948). *Methods Time Measurements*. New York: McGRAW-Hill.
- Miller, M., Griese, D., Peterson, M., Summers, J., & Mocko, G. (2012). Reasoning: Installation Process Step Instructions as an Automated Assembly Time Estimation Tool. *Proceedings of the ASME 2012 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference, 12-15 August 2012*. Chicago, IL.
- Renu, R., Mocko, G., & Koneru, A. (2013). Use of Big Data and Knowledge Discovery to Create Data. *Procedia Computer Science*, 20, 446-453.
- Salmi, A., David, P., Blanco, E., & Summers, J. (2015.a). Assembly Modelling and Time estimating during the early phase of Assembly Systems Design. *Proceedings of the 15th IFAC INCOM Symposium on Information Control Problems in Manufacturing, May 11-13, 2015*. Ottawa.
- Salmi, A., David, P., Blanco, E., & Summers, J. (2015.b). Deciding the level of automation during the design of assembly systems: literature review of decision methods and a new approach proposal. *Proceedings of the CIE45 Conference on Computers & Industrial Engineering, October 28-30, 2015*. Metz.
- Salmi, A., David, P., Summers, J., & Blanco, E. (2014). A modelling language for assembly sequences representation, scheduling and analyses. *International Journal of Production Research*, 52(13), 3986-4006.

ANNEX
Complementary figures and tables

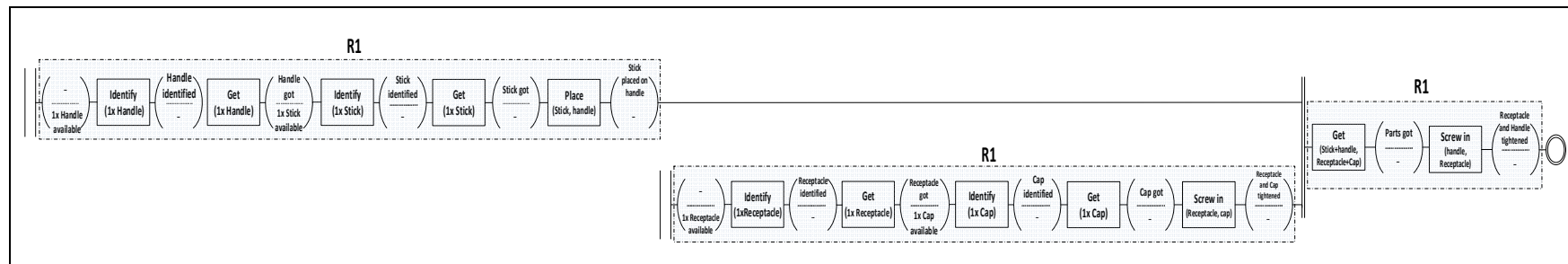
(a) Product to assemble: design and sequence modeled using an AND/OR graph with tasks representation



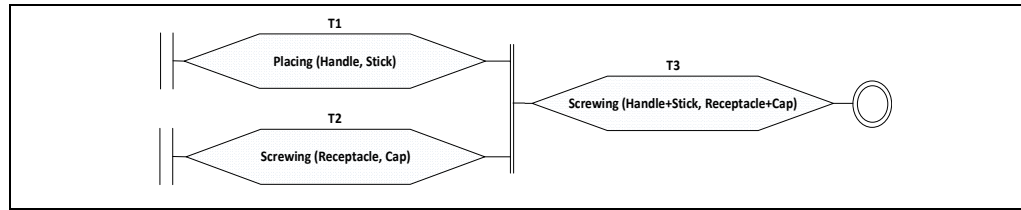
(b) Process modeling using ASML and standardized vocabulary based on product design and sequence of Figure 1.a



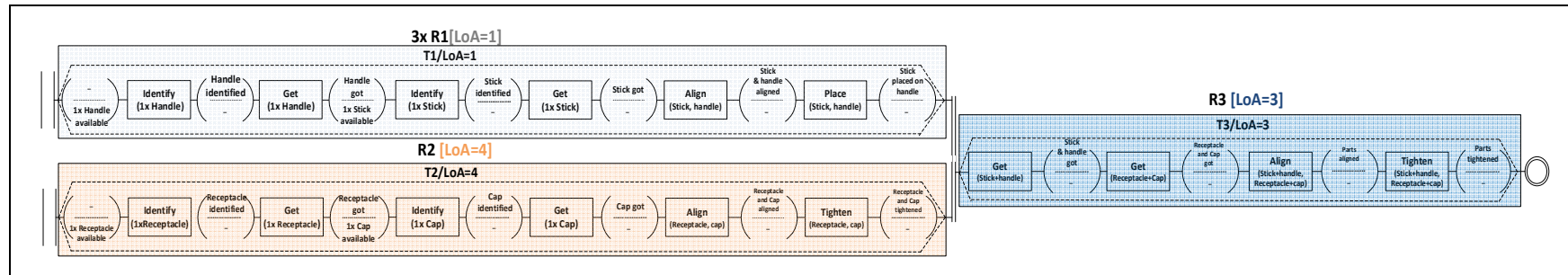
Annex Figure 1: Assembly process model representation based on product design (Reproduced from (Salmi, David, Summers, & Blanco, 2014))



Annex Figure 2: An alternative of resources assigning to motions (Reproduced from (Salmi, David, Summers, & Blanco, 2014))



Annex Figure 3: An example of assembly modelling using proposed approach of tasks vocabulary



Annex Figure 4: An example of resources automation alternatives description using proposed approach of tasks vocabulary

Annex Table: The Screwing task decomposing to required elementary motions in a 4 levels automation scale

		Screwing (LoA, P1, P2, Screw)			
		LoA=1 <i>(Manual using only physical strength)</i>	LoA=2 <i>(Manual assisted with automated tool)</i>	LoA=3 <i>(Automatic dedicated machine)</i>	LoA=4 <i>(Robotic: Industrial Robot)</i>
Corresponding Motions' decomposition	Screwing (LoA, P1, P2, Screw)				
	Identify(P1) Get(P1) Identify(P2) Get(P2) Identify(screw) Get(screw) Handstart(screw, P1, P2) Identify(screwDriver) Get(screwDriver) Tighten(LoA=1, screw,P1,P2)	Identify(P1) Get(P1) Identify(P2) Get(P2) Identify(screw) Get(screw) Handstart(screw, P1, P2) Identify(ScrewingTool) Get(ScrewingTool) Tighten(LoA=2,screw,P1,P2, ScrewingTool)	Align(P1,P2) Insert(screw,P1,P2) Tighten(LoA=3, screw,P1,P2)	Identify(P1) Get(P1) Identify(P2) Get(P2) Align(P1,P2) Identify(screw) Get(screw) Insert(screw,P1,P2) Tighten(LoA=4, screw, P1,P2)	
	Screwing (LoA, P1, P2)				
	Identify(P1) Get(P1) Identify(P2) Get(P2) Align(P1,P2) Tighten(LoA=1, P1,P2)	Identify(P1) Get(P1) Identify(P2) Get(P2) Handstart(P1, P2) Identify(ScrewingTool) Get(ScrewingTool) Tighten(LoA=2, P1,P2) Align(P1,P2) Tighten(LoA=2, P1,P2, ScrewingTool)	Get(P1) Get(P2) Align(P1,P2) Tighten(LoA=3, P1,P2)	Identify(P1) Get(P1) Identify(P2) Get(P2) Align(P1,P2) Tighten(LoA=4, P1,P2)	