Early Phase Assembly Systems Design, Automation Alternatives Description, and Optimization : A Support to Automation Decision

Anas Salmi

Communauté UNIVERSITÉ Grenoble Alpes

THÈSE

Pour obtenir le grade de

DOCTEUR DE LA COMMUNAUTÉ UNIVERSITÉ GRENOBLE ALPES

Spécialité : **Génie Industriel**

Arrêté ministériel : 7 août 2006

Présentée par

« Anas SALMI »

Thèse dirigée par **« Eric BLANCO»** et
co-encadrée par **« Pierre DAVID»** et **« Joshua D. SUMMERS »**

préparée au sein du **Laboratoire G-SCOP**
dans **l'École Doctorale IMEP2**

Aide à la Décision pour l'Optimisation du Niveau d'Automatisation durant la Conception des Systèmes d'Assemblage Industriels

Thèse soutenue publiquement le **« 05/12/2016 »**,
devant le jury composé de :

M. Bernard GRABOT
Professeur, Ecole Nationale d'Ingénieur de Tarbes, Rapporteur
Mme Olga BATTAIA
Professeur, ISAE-SUPAERO, Toulouse, Rapporteur
M. Giorgio COLOMBO
Professeur, Politecnico di Milano, Examinateur
M. Emmanuel CAILLAUD
Professeur, Université de Strasbourg, Président
M. Joshua D. SUMMERS
Professeur, Clemson University, Caroline du Sud, Co-encadrant
M. Pierre DAVID
Maître de conférences, Grenoble INP, Co-encadrant
M. Eric BLANCO
Maître de conférences, HDR, Grenoble INP, Directeur de thèse

Anas Salmi : *Aide à la Décision pour l'Optimisation du Niveau d'Automatisation durant la Conception des Systèmes d'Assemblages Industriels,* Thèse de doctorat, *05 Décembre 2016*

Early Phase Assembly Systems Design, Automation Alternatives Description, and Optimization: A Support to Automation Decision Making

Anas Salmi

Acknowledgements

I would like first to thank the jury members for their acceptance to evaluate this work Pr. Bernard Grabot, Pr. Olga Battaïa, Pr. Giorgio Colombo, & Pr. Emmanuel Caillaud

I would like then to take this opportunity to gratefully thank my advisors Dr. Eric Blanco, Dr. Pierre David, and Pr. Joshua Summers for their valuable and continuous support and advices through this PhD. They have always been available to give their help, feedbacks, and advices. Their presence was always constructive and motivating. I learned from them a lot in both research and teaching. I particularly appreciate their belief in me.

I wish to express my gratitude to the G-SCOP laboratory staff and colleagues for the favourable and pleasant atmosphere for working. I would like also to thank Industrial Engineering School of Grenoble INP staff, colleagues, and students for the exciting opportunity of teaching during the three years of the PhD in such great establishment.

I would like also to thank CEDAR laboratory staff and research members in Clemson University in the United States for hosting me during six months as a research scholar. In this context, I would like to thank "Région Rhône-Alpes" for its participation in concretising and organizing the partnership and stay. The experience was for me unique, beneficial, and encouraging to pursue with a high motivation my research career.

I wish also to particularly thank our industrial partners for their promoting welcoming, discussions, and case studies. Their contributions were significantly helpful and lightening multiple aspects in this research topic.

My special thanks go to my family members for their unwavering love and faith. I am particularly grateful to my dad who encouraged me to do the PhD.

Résumé

Le contexte industriel s'avère de plus en plus difficile dans le cadre de la mondialisation, de la compétition internationale, et de la complexification des exigences des clients. Ceci incite les industriels à réfléchir sans cesse à maitriser et optimiser leurs processus de production dans le but, en particulier, d'obtenir un meilleur dimensionnement et une exploitation des systèmes de production avec une meilleure maitrise des coûts d'industrialisation.

- ***Objectifs***

Cette thèse s'intéresse en particulier aux lignes d'assemblage. L'optimisation de ces systèmes et processus forme une réelle problématique préoccupant les industriels quels que soient leurs domaines ou leurs localisations. Une meilleure conception du processus d'assemblage avec un choix judicieux en matière d'automatisation devient crucial afin d'assurer une compétitivité sur le marché.

Ce travail a pour objectif la détermination d'une méthode d'aide à la décision pour la modélisation et la conception de systèmes d'assemblage. La détermination du niveau d'automatisation optimal en prenant en compte à la fois le design du produit à assembler, les informations stratégiques de la production, et divers facteurs impliqués dans la décision, est un point central de la contribution.

- ***Etat de l'art***

En premier lieu, une revue générale de la littérature a été menée sur le sujet de la détermination du bon niveau d'automatisation pour les systèmes d'assemblage. L'état de l'art a balayé différent axes: les tendances et les axes de recherche en matière d'automatismes, le concept de niveau d'automatisation, et les méthodologies existantes guidant la décision.

Cette revue a permis de fixer une échelle adéquate de description des niveaux d'automatisation adaptée à la phase de conception des systèmes d'assemblage. Nous avons ainsi défini l'automatisation de chaque ressource d'un processus sur une échelle à 4 niveaux : Manuel (Niveau 1), Manuel avec assistance automatisée (Niveau 2), Automatique (Niveau 3), ou Robotique (Niveau 4). La revue a également permis de souligner l'aspect multicritères de la décision. Une liste de critères à prendre en compte a été construite à partir de la littérature. Cette liste montre les critères et facteurs à prendre en compte en matière de décision du niveau d'automatisation, tels que la productivité, le coût, la qualité, la flexibilité, l'ergonomie, la stratégie et les préférences de l'industriel, la localisation et le contexte économique, la capacité d'investir, ou l'aspect social de l'entreprise.

La revue a confirmé que peu de travaux traitent le problème d'aide à la décision par une approche objective et méthodique afin de pouvoir décider de l'automatisation. Une analyse approfondie des méthodes de décision a été élaborée. Elle a confirmé que les méthodes qui existent ne fournissent pas une aide significative qui soit alignée avec les exigences complexes et multicritères du sujet. Les méthodes considèrent peu de critères et ne proposent pas d'approches concrètes d'analyse d'alternatives pour aboutir à une solution optimale en automatisation.

- ***La proposition d'une nouvelle approche***

Une nouvelle approche est donc proposée pour guider la décision. La méthode (Figure ci-dessous) a été définie pour répondre à des exigences décrivant et reflétant le besoin en matière de décision : une méthode applicable en phase de conception, objective, analytique, permettant des automatisations partielles, multicritère, prenant en compte le contexte de l'industriel, et garantissant l'optimalité.

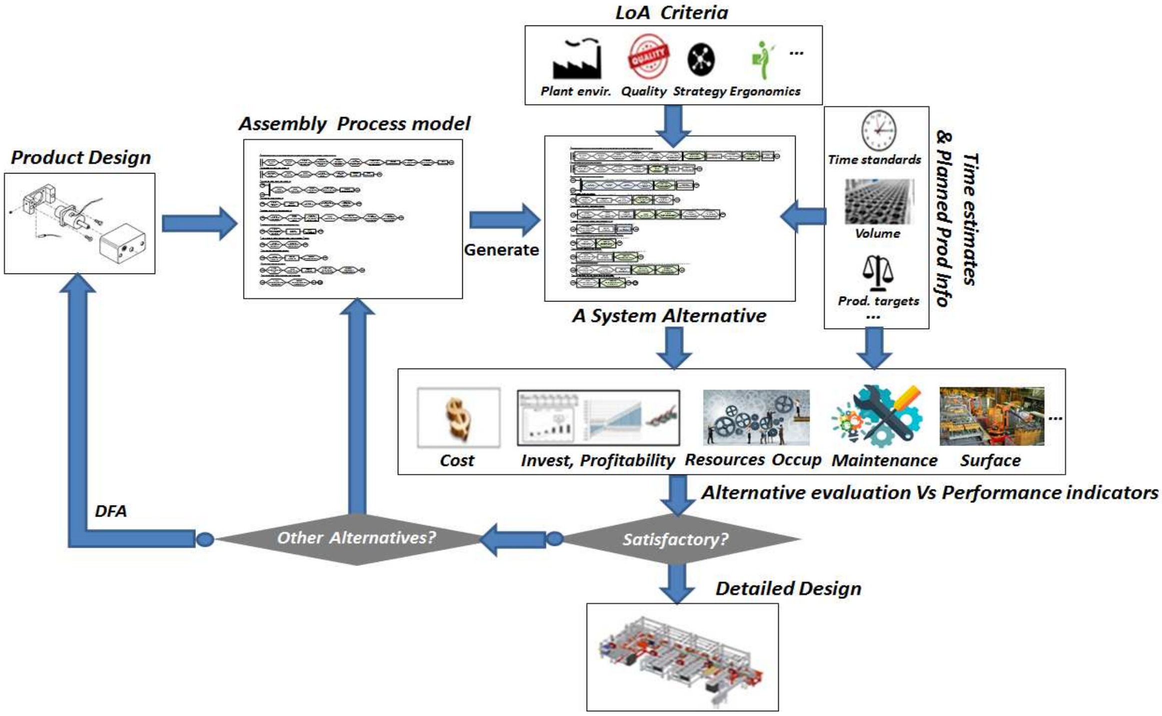


Figure: l'approche de décision du niveau d'automatisation proposée

Cette approche nécessite, pour être utilisable et efficace en pratique, plusieurs développements de modules afin de permettre son implémentation.

- ***Propositions détaillées et implémentations***

La méthodologie de décision que nous avons définie nécessite en particulier un langage standardisé de représentation des processus d'assemblage et d'allocation des ressources, une méthodologie pour la considération des critères, une approche d'estimation du temps et du coût, et une technique d'optimisation par génération de scénarios d'automatisation et leur évaluation. Des propositions ont été effectuées dans chacun de ces axes au cours de la thèse.

- ***Un langage de modélisation des processus d'assemblage***

Les méthodes graphiques existantes pouvant être utiles en matière de représentation de processus ont été revues et analysées. Ceci a conduit à la définition d'une nouvelle méthode graphique dédiée et établie par analogie et combinaison de méthodes existantes. Après la définition de ce langage baptisé *ASML* pour « *Assembly Sequences Modeling Language* », un vocabulaire standardisé de mouvements élémentaires d'assemblage a été réutilisé de la littérature. Une extension du vocabulaire a aussi été proposée pour compléter l'utilisation de l'*ASML*. Le processus d'assemblage peut être ainsi modélisé sous forme d'une succession d'actions élémentaires standardisées représentées graphiquement.

Un vocabulaire de plus haut niveau, celui des tâches et techniques d'assemblage, en lien avec le premier, a ensuite été défini. Ceci a été fait dans un souci de standardisation de l'approche et de soutien à la génération de scénarios d'automatisation. Cette vision plus macroscopique permet de faciliter la modélisation tout en la laissant indépendante du niveau d'automatisation de la ressource. Cela permet aussi de réduire l'explosion combinatoire due au nombre important de mouvements élémentaires résultant de l'utilisation du vocabulaire initial.

La connexion entre les 2 vocabulaires a également été réalisée. L'appel à une tâche selon son type, ses paramètres (les caractéristiques des composants du produit à assembler et les critères de complexité en assemblage impliqués), et le niveau d'automatisation sélectionné pour la réaliser est lié avec le premier vocabulaire. Ceci permet une obtention automatique de la séquence des opérations élémentaires possible pour chacune des tâches représentées. Ce processus, itéré à toutes les tâches du modèle, permettra ainsi l'obtention du modèle de représentation fin en opération élémentaires d'assemblage. L'approche permet donc de bénéficier du vocabulaire de tâche défini réduit ainsi que du niveau détaillé des mouvements élémentaires. Le passage à ce vocabulaire plus fin offre plusieurs avantages tels que la détermination des temps d'assemblage ou la détection des opérations répétitives comme signes propices à une éventuelle automatisation. L'ensemble ASML et vocabulaires standardisés permet une meilleure rapidité, facilité, et compacité, pour définir, organiser, et représenter un processus d'assemblage pour un produit donné.

- ***L'identification et considération des critères de décision***

En bénéficiant de la capacité du langage défini, des vocabulaires, et de la possibilité de passer d'une granularité à une autre, une démarche d'allocation adéquate des ressources a été proposée. Pour ce faire, l'identification des critères impliqués dans la décision du niveau d'automatisation a été effectuée. Une liste de 73 critères de décision a été reprise de la littérature.

Une démarche de prise en compte des critères, en cohérence avec l'approche de décision, a été définie. Elle est basée sur l'analyse des taches du modèle par rapport à des critères que le décideur pourra sélectionner. L'analyse devra être effectuée également par rapport aux différents niveaux d'automatisation possibles. Une représentation matricielle de l'analyse est proposée. Cette matrice sera ensuite intégrée dans l'approche de génération de scénarios afin d'interdire, permettre, ou imposer certains scénarios en fonction des critères pris en compte.

- ***Estimation de temps et équilibrage des scénarios***

L'allocation des ressources aux taches du modèle prend en considération également les estimations des temps des taches. On réalise l'estimation de temps par passage par la couche de la description des mouvements élémentaires. L'estimation de temps propose l'utilisation d'une base de données de temps standard pour les mouvements élémentaires. Les valeurs sont à sélectionner en prenant en compte le type de mouvement élémentaire, les paramètres des composants à assembler, et le niveau d'automatisation sélectionné pour la tâche.

Le processus d'estimation de temps des taches est réalisé pour toutes les taches du modèle. L'estimation du temps dépend de l'architecture du modèle initial qui peut présenter des taches et ressources en série, ou parallèle (divergence en 'ET'), ou en choix de sous-séquences (divergence ne 'OU'). Des règles ont été développées pour assurer le calcul des temps des ressources prenant en compte les temps des tâches assignées, les temps précédemment obtenus de ces différentes tâches et l'architecture des tâches provenant du modèle *ASML* initial. Le résultat appliqué à chacune des ressources du modèle permet de calculer les temps d'assemblages au niveau des ressources assignées aux tâches pour un scénario d'automatisation donné.

Des règles d'équilibrage du scénario en question vis-à-vis de la productivité requise ont été proposées. Ces règles ont été définies en cohérence avec les principes du « lean manufacturing », tels que la gestion du flux, la synchronisation entre les ressources et l'ajustement par rapport au takt-time. Cet ajustement ou équilibrage, lorsque nécessaire, est effectué selon 2 cas possibles obtenus dans l'allocation initiale. Dans le premier cas, la ressource est plus lente que nécessaire. Dans le deuxième, la ressource est plus rapide.

Dans la première situation, où la ressource est lente par rapport à la cadence cible, la duplication de la ressource peut être proposée. Le nombre de duplication est à déterminer selon la cadence obtenue au niveau de la ressource (à l'aide des estimations de temps des tâches correspondantes) par rapport à la cadence requise.

Dans la deuxième situation où la ressource est plus rapide que ce qui est requis, on propose, si cela est techniquement faisable, l'affectation de plus de tâches à la ressource tant que la cadence de cette dernière reste supérieure à la cadence cible. Cette réaffectation prend en compte les contraintes de compatibilité entre les tâches devant être exécutées par la même ressource dans un niveau d'automatisation donné.

- ***Estimation des coûts d'assemblage***

Une fois qu'une configuration d'automatisation valide et cohérente respectant les règles précédemment mentionnées est disponible, d'autres études peuvent être menées. Nous nous intéressons ensuite à l'intégration du critère coût vu l'‘importance des investissements pouvant être générés. Une revue exhaustive en estimation de coût a été établie. A sa suite, un modèle intégré permettant l'estimation du coût d'assemblage par produit a été défini. Ce modèle, basé sur les estimations de temps du processus ainsi que sur des informations stratégiques de la production planifiée, a vocation de prédire le coût d'assemblage par produit pour une alternative de système. Le modèle de coût à appliquer par ressources du modèle traite et détaille les coûts par familles de ressources.

On distingue dans le modèle d'estimation de coût, 2 familles selon la ressource en question : des coûts générés par l'utilisation de ressources manuelles (opérateurs), et des coûts générés par l'utilisation de machines. Selon le niveau d'automatisation en question (4 niveaux), la famille de coût est associée. Les niveaux 1 et 2 considèrent la 1ère famille de coût. Les niveaux 2, 3, 4 considèrent la seconde. Le niveau 2, considère les 2 familles car il associe à la fois travail manuel et machine.

Chaque famille a été ensuite détaillée et décomposée en classes. Trois classes ont été définies : des coûts spécifiques au manuel, des coûts spécifiques aux machines, et des coûts communs à toute ressource quel que soit son type. Chaque classe a été détaillée en éléments de coût avec proposition d'équations, justifiées par la littérature. On s'est attaché à ce que ce modèle soit applicable dès les phases amont de la conception de la ligne d'assemblage. On distingue pour la classe des coûts spécifiques au manuel : le coût de main d'œuvre, les coûts des outils ou outillages manuels et les coûts de formations. Pour la classe des coûts spécifiques aux machines, on distingue : les coûts d'investissements en machines ou robots, les coûts d'énergie et les coûts de maintenance. Pour la dernière classe concernant les coûts à prendre en compte pour toute ressource quel que soit son niveau d'automatisation, les éléments de coûts consistent en : les coûts de préparation de la station et de sa configuration ou programmation, les coûts liés à la surface occupée et les coûts liés à la qualité séparés en coûts de rejets et coûts de retravaille des pièces non conformes.

Le modèle développé permettant d'estimer le coût d'assemblage par produit généré par une ressource selon son niveau d'automatisation a été ensuite étendu. Cette extension lui permet d'intégrer le coût de toutes les ressources de la configuration d'automatisation considérée. Le résultat donne donc une estimation du coût par produit de la configuration. Des indicateurs de performances en matière de coût ont été ensuite développés utilisant les paramètres du modèle de coût. Ces indicateurs, en plus du coût par produit calculé par le modèle, ont pour objectif de comparer une alternative d'automatisation donnée par rapport à d'autres alternatives. Ces indicateurs sont: l'investissement initial total, l'investissement global (qui inclue l'investissement initial total avec des investissements intermédiaires pouvant être requis selon la durée de vie estimée des ressources et la durée de production planifiée), la surface totale de la solution et son coût mensuel associé, l'énergie mensuelle et son coût associé, la période d'amortissement et le retour sur investissement.

- ***Génération de scénarios et optimisation***

A cette phase, tous les éléments sont réunis pour l'évaluation d'une configuration d'automatisation donnée ainsi que des facteurs permettant de la comparer à d'autres. Or l'objectif est de déterminer la meilleure configuration possible, il est nécessaire d'appliquer l'approche sur plusieurs options possibles ou alternatives d'assemblage. Ce processus ne peut évidemment pas être réalisé d'une manière manuelle. Une informatisation des approches et leurs implémentation est nécessaire. Ceci demande une technique de génération de scénarios et leur évaluation. Cette implémentation a été réalisée par le développement d'un modèle d'optimisation basé sur une technique de résolution exacte par formulation mathématique en programme linéaire. Ce module permet la recherche de la solution optimale. Il permet une implémentation des approches précédentes avec saisie du modèle *ASML* initial, considération des critères de décision, considération des estimations de temps et des équilibrages optimisant la solution, et du modèle de coût représentant la fonction objectif à minimiser. L'approche a été implémentée sur un solveur commercial (*IBM ILOG CPLEX OPL 12.6*) et a donné des résultats prometteurs.

- ***Conclusions et perspectives***

Au cours de cette thèse une nouvelle approche de décision du niveau d'automatisation a été proposée. Des modules couvrant divers aspects permettant son implémentation ont également été développés. Ces modules concernent la modélisation de processus d'assemblage, l'identification et la prise en compte des critères de décision, l'estimation de temps et de coûts d'assemblage et la génération de scénarios et leur optimisation. Les différentes propositions ont été justifiées au préalable par des études et analyses de la littérature dans les domaines concernés.

Les travaux effectués ont fait objet de publications dans des revues (Salmi A. , David, Summers, & Blanco, 2014) (Salmi A. , David, Blanco, & Summers, 2016) ainsi que des communications en conférences internationales (Salmi A. , David, Blanco, & Summers, 2015) (Salmi, Dhulia, Summers, David, & Blanco, 2015) (Salmi A. , David, Blanco, & Summers, 2015.b) (Salmi A. , David, Blanco, & Summers, 2016). D'autres articles sont en soumission (Salmi A. , David, Blanco, Summers, & Briant, en revue) (Salmi A. , David, Blanco, & Summers, en soumission).

En termes de perspectives, en plus de la possibilité de servir à la modélisation et optimisation du niveau d'automatisation, l'approche développée pourra être utilisée pour la re-conception ou l'amélioration de processus d'assemblages existants. Plusieurs axes de continuité intéressants dans le sujet peuvent être proposés. On propose de les classifier en 3 catégories: axes d'amélioration ou axes de développement à court terme, axes d'élargissement ou à moyen terme, et axes d'extension à des sujets connexes ou à long terme. Pour les axes d'amélioration, on propose plus de validations industrielles, une échelle de niveaux d'automatisation plus large, une approche d'optimisation multi-objectifs, la prise en compte des fluctuations monétaires, et la prise en compte des incertitudes de données. Pour les développements à moyen terme, des heuristiques pour des problèmes de taille plus larges pourront être développées. Une automatisation de la génération des modèles du processus initial directement à partir d'outils CAO de produit pourra être développée. Une optimisation plus globale avec génération de toutes les séquences d'assemblage par génération de leurs solutions optimales pourra être développée également. Finalement, en ce qui concerne les extensions de l'approche développée, on propose un focus sur la proposition de directives «Design for Assembly» (DFA) aidant l'amélioration du concept du produit dans le cas où la solution optimale d'assemblage obtenue n'est pas satisfaisante. La prise en compte d'une production flexible ou à volume variable pourra représenter également un axe important. On propose également comme axe d'extension la modélisation de ressources complexes pouvant assurer plus d'une tache simultanément. Pour finir, une optimisation du système d'assemblage pour différentes localisations avec prise en compte des coûts et critères locaux associés pourra représenter un axe futur d'importance.

Content

List of Figures

List of Tables

Acronyms

A: Automatic assembly machine

ASML: Assembly Sequences Modelling Language

DFA: Design For Assembly

GALBP: General Assembly Line Balancing Problem

GUI: Graphic User Interface

LoA: Level of Automation

M: Manual assembly

MT: Manual with automated Tool / assistance assembly

MTBF: Mean Time Between Failures

MTM: Methods Time Measurement

MTTR: Mean Time To Repair

NP-hard: Non deterministic Polynomial computational time hard problem

PP: Process Productivity

R: Robotic assembly

RCALB: Resource Constrained Assembly Line Balancing

ROI: Return On Investment

RP: Resource Productivity

SALBP: Simple Assembly Line Balancing Problem

SOP: Sequence of OPeration language

SP: Sequence Planner tool

TT: Takt-Time

PART 1. AUTOMATION DECISION LITERATURE & PROPOSED APPROACH OUTLINE

Chapter 1. General Introduction & Research Question

Abstract

This chapter aims at introducing the research topic, context of the study, and the research question. It introduces the challenges encountered by manufacturers leading to optimize as much as possible the design of their new production systems, particularly from an automation point of view. Our interest is particularly focused on assembly processes. It is underlined that the concept of Level of Automation (LoA) is generally closely linked to the assembly system performance. Therefore, we introduce this LoA concept throughout a literature review to clarify the different but coherent existing definitions. The objectives of the thesis and the research methodology are finally explicated.

1.1. Introduction

The decision about automation in assembly systems design continues to be an important industrial challenge; even more as new and advanced automation technologies are developed. The topic is continuously debated within manufacturing steering circles and does not seem to be yet mastered, particularly in assembly manufactories. In fact, automating the process is not always worth doing or satisfactory. As an example, the results of a survey in German companies about automation show that more than a third of 355 surveyed companies planned to reduce the LoA within their plants after having experienced a high LoA (Lay & Schirrmeister, 2001) (Gorlach & Wessel, 2008). During initial meetings between our team at G-SCOP laboratory and a tier one automotive parts manufacturer, an important point was that ones can observe different LoA for similar tasks within different plants, without strong rational supporting it. Also, manual assembly is frequently found. Then, based on multiple visits to different assembly manufacturers in France and in the United-States, it was seen that in spite of the high labor rate in these countries, manual assembly is still significantly used. This is confirmed by Boothroyd (Boothroyd G. , 2005) who pointed out that many workers assembling mechanical products are still using the same basic tools as those employed at the time of the industrial revolution.

Nevertheless, the path leading to the decision of automating the production processes is not clear. Methodologies or decision support tools orienting companies to the most suitable assembly systems with an optimal LoA are still to be developed (Lindström & Winroth, 2010). Currently, the discussion on the question to automate or not is not well documented and the path that leads to the final decision is not traceable (Ross, 2002). In fact, the usefulness of automation is highly dependent on finding appropriate distribution of tasks between the human and the technical system (Frohm, Lindstrom, Winroth, & Stahre, 2006). In contrast to the voluminous technical literature in automation, generally treating how practically automating operations or new technologies development, there is limited research considering different options and automation levels for performing tasks, and analyzing the optimal way to execute tasks (Parasuraman & Sheridan, 2000).

In this chapter we start by showing the current difficult industrial context characterized by a rude competition and a necessity to master and optimize production systems. This context influences the view of manufacturers on automation (section 1.2). Then, we clarify the concept of the Level of Automation labelled "LoA" based on literature definitions (section 1.4). We introduce a LoA scale to tackle the problem of automation decision for new assembly systems design. Finally, we present the objectives of the study (section 1.5) before concluding the chapter (section 1.6).

1.2. Current Industrial context and global competition

The driving objectives of manufacturers for their assembly processes are generally to increase rapidity, reliability, and robustness. In the past, their choices were mainly made from a technical point of view. Hill (Hill, 1999) stated that "the major problem with the technology oriented literature is that it focuses on the specific applications and the potential improvements but, unfortunately, fails to explain how to select technological investments that support a business". This is making manufacturers often opting for high LoAs that may be oversized and more productive than required, and not always the absolutely optimal system which fits better their production and context. From another side, they are realizing that a maximum LoA is not a guarantee of success or better margins for products to sell. In this context, Fieldman and Slama (Feldmann & Slama, 2001) point out that to be nowadays competitive it is absolutely necessary for manufacturers to align their products and production with customer demands. The customer orientation is leading to an increasing number of variants and to shorter product life cycles requiring a high degree of flexibility. Brainbridge (Brainbridge, 1983) noted that in a time of rapidly changing technologies and shortening product life cycles, many companies are focusing on automation as a means for competing on a more demanding market. However, "an increased usage of automation does not necessary result in increased benefits". The author mentions that this may be due to the market change by mass customized and individualized products and variability of the demand. The statement made in 1983 is still valid and relevant to light the current context where new technologies are coming to market as augmented reality, reconfigurable robots, cooperative man-machine systems, and so on. Productions systems must then be designed to handle such high variety while at the same time achieving mass production quality and productivity (Hu, Zhu, Wang, & Koren, 2008). Manufacturers should also consider and take into account the product market life being shorter and shorter.

In current industries, thanks to new advanced automated and robotized solutions, automation becomes more accessible for manufacturers due to the decrease of automated processes and robots costs and related investments. They become also easier to configure, to reprogram, and to manage with possibilities of autonomous systems using vision sensors, objects recognition, and learning programming processes. This encourages more and more manufacturers with regard to automated solutions becoming more approvable. Yet, this imposes a right reasoning and a need to analyze and compare the different possible solutions to find the most appropriate LoA.

The manufacturers' environment and context becomes more and more competitive. In order to survive and to secure their position in the worldwide market, they must, inter alia, manage their investments, and opt for the best solution that fits better the planned production. Therefore, the decision on the right Level of Automation (LoA) is an issue that must be taken carefully.

1.3. Automation: a powerful tool that should be appropriately decided

One of the solutions for every manufacturer to face the cost pressure and the tough competition is to rationalize automation. In fact, automation, when appropriately designed, allows considerably reducing costs and solving multiple issues of quality, repeatability, ergonomics, and so on. Some of the major arguments of using automation are reducing operating time and cost, particularly in highly costly assembly field. Assembly operations represent between 30% and 50% of overall production time in the automotive industry, and up to 70% in other industries (Lotter & Wiendahl, 2006). Therefore it is obvious that the opportunity to reduce the global product cost from the assembly step is worthwhile, and as cited by Ross (Ross, 2002), automation can help to rationalize assembly.

During the last years, industrial automation became more and more adopted by manufacturers for production systems. As noted in the introduction, according to a survey (Lay & Schirrmeister, 2001), the use of automation within factories especially in Germany, has experienced constant growth between 1989 and 1999. Yet, more than a third of the 355 surveyed companies planned to reduce the level of automation within their plants after their experience with high automation. It can be seen based on these results that automation can be advantageous if we see the two third of satisfied companies, and disadvantageous if we look at the remaining third of the companies which are disappointed and want to reduce their LoA. In the same context Boothroyd (Boothroyd G. , 2005) noted that "although during the last few decades, efforts have been made to reduce assembly costs by the application of high speed automation and, more recently, by the use of assembly robots, success has been quite limited.

Therefore we can confirm the need to find the convenient or suitable automation level for a given company and production circumstances. The tradeoff should be found according to numerous parameters that will be later detailed in chapter 2. Selecting the appropriate automation level can avoid useless and unnecessary high investment of inappropriate solutions, eliminate unsuitable alternatives with regard to manufacturer's best practices, and key decision criteria identification that will be presented in next chapter. Fixing the suitable LoA early in the design phase can allow avoiding extremely high cost of changing the assembly system design if the unsuitability is realized late or after the line design implementation. A pre-study or early phase analysis on the line structure, possible options, and associated automation level should be then vigilantly driven before tackling the detailed design.

This introduces the research question of this thesis:

> ***"How to decide about the most appropriate Level of Automation (LoA) when designing new assembly systems, and where to automate or not and to what extent throughout the process?"***

We proceed in next section to the presentation of the concept of Level of Automation (LoA) that has been tackled in various ways in the literature.

1.4. The concept of Level of Automation (LoA) and LoA scales

In this section, our purpose is to introduce the concept of Level of Automation (LoA). There is currently no standard or unique definition of the LoA concept. Generally the concept is supported by a measuring scale, but there are obviously often different.

We proceed in this section first by presenting the definitions and LoA scales that we found in the literature. Then, we present an adapted scale we propose for our specific research case of early phase assembly systems design and automation decision.

1.4.1. Literature definitions of LoA and existing scales

The definitions around this concept are quite different but coherent. The differences concern the accuracy of the scales, the qualitative or quantitative way of description, the analytic aspect or granularity level of the description with a general description of the whole process or lower layer of description that can reach the assembly operations layer.

We organize the different definitions of LoA and associated scales to seven different classes as shown in Figure 1. Each of them is detailed independently in a separate subsection.

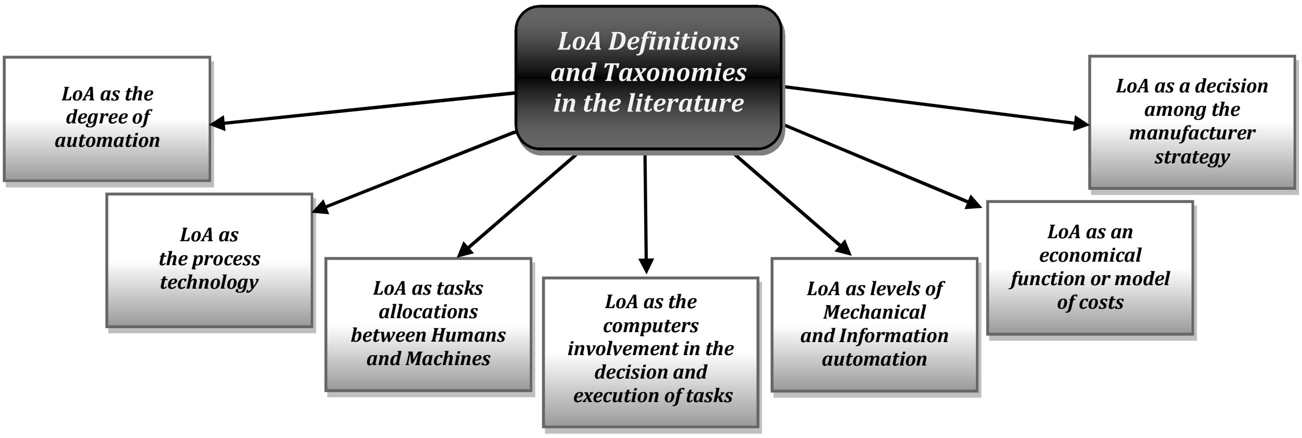


Figure 1: LoA definitions and taxonomies in the literature

- **<u>LoA as the degree of automation</u>**

The level of automation was defined in (Gorlach & Wessel, 2008) as the portion of automated operations of a system in relation to the whole operations of the system. In this same sense, it was explicitly defined as the degree or extent to which automated systems, i.e. machined processes work areas, are employed (Fasth, Stahre, & Dencker, 2008). The notion of Level of Automation (LoA) may be confused with the notion of degree of automation as mentioned in (Sheridan, 1997). This definition provides a high level view which concerns the whole process.

- **<u>LoA as the process technology</u>**

LoA can be defined as the ''process technology'' and refers to the technology of production equipment (Hill, 1999) (Slack & Lewis, 2002). This definition, however, does not take into consideration the integration of humans in the process but concerns only the technical description of automated processes. The definition is also very generic and is not previse on the concept of "technology".

- **LoA as tasks allocations between Humans and Machines**

The LoA was also defined as the relation between human and technology in terms of task and functions allocation, which can be expressed as an index between 1 (total manual work) and 9 (total automation) (Frohm, Lindstrom, Winroth, & Stahre, 2006). The graduation from 1 to 9 reflects the dominance of tasks allocated to machines in relation to those allocated to humans. In other works, LoA was described also in discrete steps but consisting in manual, semi-automatic, and automatic depending on task allocation between operators and equipment (Säfsten, Winroth, & Stahre, 2007).

This LoA was viewed in the literature as a dimension of process technology consisting of a mix of process technology and humans task by task. Optimizing this LoA consists in this point of view in optimizing the tasks allocation between humans and machines. It was mentioned that humans and machines can be complementary. The advantages of each of them have to be well interpreted and employed during tasks allocation according to the specific need. For example, technological processes are recognized for their efficiency and productivity while humans have the benefit to be flexible (Lindström & Winroth, 2010). In order to fully utilize the capabilities of both humans and machines in system, possibly semi-automated, the interaction between them needs to be well conceived (Sheridan, 2002). Such interaction has traditionally been described in human factors engineering in the terms of function allocation, implying a system design process where functions are allocated to humans or to machines, respectively. The resulting functions allocations may be described as the LoA, ranging from entirely manual operations to full automation (Sheridan, 2002). In the same sense, to help the appropriate allocation, Sheridan (Sheridan, 1995) proposes simply "allocating to the human the tasks best suited to the human, allocating to the automation the tasks best suited to it".

For more concretization of the collaboration between humans and machines in the LoA definition, Satchell (Satchell, 1998) defined automation as a task sharing approach between both the human and technology. Fasth and Stahre (Fasth & Stahre, 2008) noted that the assembly system needs to have the "right" levels of automation that they defined as the optimal mix between human and technology for each task and operation in the system.

In the same perspective of LoA optimization by optimal tasks allocation between humans and machines, Endsley and Kiris (Endsley & Kiris, 1995) underlined that when keeping the human involved in automation and tasks allocation, some intermediate LoA may provide better performances than the ones that can be obtained with highly automated systems.

- **LoA as the computers involvement in the decision and execution of tasks**

In this category, LoA is defined as the extent to which the computer is involved in achieving the task. A LoA taxonomy incorporating 10 levels was defined in (Sheridan & Verplanck, 1978). This taxonomy basically incorporates feedbacks of what the human should be told by the system, as well as relative sharing of functions determining options, selecting options and implementing. The definition introduces which agent (the human or the computer) gets or requests options, selects actions, requests or approves selection of actions, starts actions, approves start of actions, or reports actions and has been framed in terms of the teleoperation environment. The scale is detailed in Table 1.

Table 1: LoA taxonomy as used by Sheridan and Verplank (Sheridan & Verplanck, 1978)

LoA	*Desciption*
1	Human does the whole job up to the point of turning it over to the computer to implement
2	Computer helps by determining the options
3	Computer helps to determine options and suggests one, which human need not follow
4	Computer selects action and human may or may not do it;
5	Computer selects action and implements it if human approves
6	Computer selects action, informs human in plenty of time to stop it
7	Computer does whole job and necessarily tells human what it did
8	Computer does whole job and tells human what it did only if human explicitly asks
9	Computer does whole job and decides what the human should be told
10	Computer does the whole job if it decides it should be done, and if so, tells human, if it decides that the human should be told

In this same category, another similar 10 levels scale was developed in (Parasuraman & Sheridan, 2000) where a low level implies mainly manual tasks whereas a high level implies limited involvement of manual in performing the task Table 2.

Table 2: LoA Scale of different levels of automation as defined in (Parasuraman & Sheridan, 2000)

LoA	*Desciption*
1	The computer offers no assistance, humans must do it all
2	The computer offers a complete set of action alternatives, and
3	Narrows the selection down to a few, or
4	Suggest one, and
5	Executes that suggestions of humans approve, or
6	Allows humans a restricted time to veto before automatic execution, or
7	Executes automatically, then necessarily informs humans, or
8	Informs them after the execution only if they ask, or
9	Informs them after execution if it, the computer, decides to
10	The computer decides everything and acts autonomously, ignoring humans

In this definition, we appreciate the involvement of humans in describing the automation levels of the process. Yet, the level of details is too extensive and may be more appropriate for an existing system description rather than to enumerate possibilities for a new system, not existing yet, to be decided and designed. The description is also more oriented on the control part of systems than on the operative part.

- **<u>LoA as Mechanical and Information automation levels</u>**

In some works in the field of automation decision, basically in improving automation level of existing processes, LoA is defined as the allocation of 2 kinds of automation: physical or mechanical LoA and cognitive or information LoA (Frohm, 2008). The mechanical LoA is described as the level of automation for physical support or mechanical activities, and information LoA as the level of cognitive (computerized) activities (Frohm, 2008)(Granell, 2007). The ranges start from totally manual to totally automatic for each of the categories (Table 3). The intermediate levels in a seven LoA scale are used to describe the moderate automation levels. The complete LoA scale is shown in Table 3.

Table 3: LoA Levels of automation as defined in (Frohm, 2008)

LoA	*Mechanical*	*Information*
1	Totally manual	Totally manual
2	Static hand tool	Decision giving
3	Flexible hand tool	Teaching
4	Automated hand tool	Questioning
5	Static machine/workstation	Supervision
6	Flexible machine/workstation	Intervene
7	Totally automatic	Totally automatic

Concerning this definition, we appreciate the involvement of manual in the automation levels description. This scale should be applied station by station. The degree of description seems to be enough accurate. Yet, distinguishing mechanical and information automation is more appropriate to improve existing processes rather than to design new systems, for which the information scale is difficult to be used. In fact, the authors (Frohm, 2008)(Granell, 2007) ask the question about how to improve existing systems design and their automation levels. We also think that the scale is a little confusing. A level 7 corresponding to totally automatic (Table 3) can be for us flexible, such as full robotic or reprogrammable-reconfigurable stations. We think that this scale describes levels with exclusion of new advanced robotic solutions that cannot be exclusively classified to a unique category of the proposed scale. They should be eventually described by another separate level. Also, the level 5 of static machine can be in some cases confused with level 7 of totally automatic. In fact, a dedicated static machine can run autonomously and can be fully automatic. Another confusing case consists in distinguishing static hand tool (LoA=2) and totally manual (LoA=1). It seems useless, at least for our automation decision issue. For us, using a manual tool (such as a screw driver or a hammer) does not change the decision about automation. In fact, if an operator needs a manual tool to perform a task, he should have and use it systematically. The question to have it or not would not be asked. The level for us should be still just 'manual'. Another problematic situation concerns the levels 3 and 4 of respectively flexible hand tool and automated hand tool are also confusing and need clarification. In fact, a flexible hand tool can be automated, such as an automatic screw driver for which the drills calibers can be changed to handle different screwing applications and screws sizes or types.

- **<u>LoA as a cost ratio</u>**

Windmark et Al (Windmark, Gabrielson, Andersson, & Stoehl, 2012) underlined the importance of optimizing the automation level in order to realize a resource efficient manufacturing and achieving a long-term sustainable production. The optimization of this LoA is driven in the authors approaches using a LoA ratio labelled x_{af} defined as a continuous function, defined as a percentage, including the equipment costs per hour (K_{CP}) and the salary costs per hour (K_D) as follow:

$$x_{af} = \frac{K_{CP}}{K_{CP} + K_D}$$

This LoA ratio varies in value obviously between 0 and 1.0. When the ration is equal to 0, the production is entirely manual. When the equipment cost is important compared to the labor cost, the ratio is closed to 1. In this case the situation is highly automated.

The quantitative and objective way to describe LoA is for us positive. Yet, the description is too aggregated and based only on a cost criterion. Also, when changing the manufacturing location, even if we assume that the equipment cost can be approximately constant, the salary costs per hour can considerably change, particularly when moving from a low cost country to a high cost one. Then, as a result the LoA value will vary for a same system or automation configuration. The gap can be significant. In fact, the cost definition is not enough representative of the physical process automation level. There can be a process with almost operations manual, for example 5 stations, and only 1 station automated, which may have a very high cost rate because of a very expensive initial investment and high energy consumption. This machine can be more costly than the total of the 5 other manual stations, particularly in the case of a low labor cost country. Then, the resulting ratio (x_{af}), will be near to 1.0, which means a full automatic process. This significance is not reflecting the real automation level of the line where only one station among six is automatic, while the remaining five ones are fully manual.

- **LoA as a strategic decision**

The link between the Level of automation and the manufacturer's strategy was highlighted in the literature. Lindström (Lindström, 2008) underlined the need to develop tools which support alignment of both strategy and operational automation levels. Lindström and Winroth noted in (Lindström & Winroth, 2010) that research has shown that alignment between manufacturing strategy and decisions regarding automation are often of an ad hoc nature. They pointed out that an appropriate level of automation should be aligned with the manufacturing strategy of the firm. Groover mentioned in (Groover, 2015) that automation involves the long-term strategies of the company related to the level of competence and where to locate production. It also influences several output factors such as quality, delivery issues, and flexibility (Groover, 2015). In (Winroth, Säfsten, Stahre, Granell, & Frohm, 2007), the authors stated that successful decisions about automation go in line with what the company aims in the long term and the decisions are synchronized with the manufacturing strategy and capabilities. When this decision of automating is pushed without linkage to the manufacturing capabilities, such investments may become real failures (Säfsten, Winroth, & Stahre, 2007).

We agree with the fact that the automation decision is a strategic decision. Yet, we need a more concrete production oriented definition that can guide to describe conceptual future processes to be designed with optimal automation levels.

1.4.2. A proposed LoA scale for new processes design and automation decision

In the issue of LoA decision, the purpose is to find the most appropriate LoA for a new process to be designed. A suitable definition and scale should be first fixed to allow an appropriate description of possible solution during the particular phase of the decision making.

According to the different reviewed definitions of section 1.4.1, multiple suitable features were found. We appreciate defining a right automation level as the best allocation of resources and their associated automation levels throughout the process. Defining LoA should be performed station by station or resource by resource, rather than on a global description of the whole process. The automation level of each resource should vary, as commonly defined in the different scales, from full manual to full automatic with intermediary levels. These intermediate levels should be defined so that they can be enough detailed to differentiate alternatives but not too much to avoid unnecessarily complex study. The scales of Table 1 and 2 are for us not appropriate for our issue. These scales are generally used to describe existing processes providing a high accuracy of description of the use of automation. This detailed description is not suitable to describe new processes to be designed that are not completely known. The scale of Table 3 is as previously presented confusing. We totally agree with the strategic aspect of the automation decision, and we find crucial to involve the manufacturer, his choices, and preferences, in the automation decision.

To describe the different intermediate levels, we propose a LoA scale tailored to our target which addresses the different issues found in the existing scales particularly of Table 3 – mechanical LoA. We suggest then the use of a 4 LoA scale to describe resources with following automation levels: Manual (LoA=1), Manual with automated assistance (LoA=2), Automatic (LoA=3), and Robotic (LoA=4). These levels are described in details in Table 4.

Table 4: The LoA scale for automation decision in new assembly systems design

LoA	Definition	Description	Illustrative Figure
1	Manual	The task is performed completely manually with the only use of human resource using his phyical strength and manual tools when needed (manual screw driver, hammer, etc).	
2	Manual with Automated assistance	The tasks is performed in a collaborative and coherent way between a human and an external machined assistance characterized by a kind of energy (electric, hydraulic, pneumatic, etc)	
3	Automatic	A dedicated automatic machine designed to perform the specific assembly task.	
4	Robotic	An industrial robot that can handle the given task or reprogrammed for other tasks (flexible).	

This scale will be used through all the proposals that will be presented in this thesis. It will be used to describe the automation level for every resource in the process and for every assembly operation executed by that resource. For us, this scale is enough accurate for new processes design and automation decision purposes. Other important aiding information consists in the resources assignments through the process, the set of assembly tasks that every resource has to perform, and the schedule of tasks and resources.

1.5. Thesis objectives and research methodology

The aim of this work is to support deciders and assembly systems designers guiding them to the most convenient level of automation. The purpose is to provide a support to the decision about automation for a new assembly system to be designed since the early design phase. We mean by early phase the phase anticipating the detailed system design and during which the system is not yet completely decided nor designed. This phase should start in our point of view near the end of the product design phase, when the product is designed but not completely fixed. The utility of starting LoA deciding at this phase is to be able to provide some feedbacks on the product design and be able to re-design it if no satisfactory assembly system solution can be found. This coincides with DFA (Design For Assembly) approaches allowing facilitating assembly and reducing assembly costs. From another side, anticipating the phase of assembly process design will allow providing instructions, reasons for automating or not, and orientations to assembly systems designers.

The decision should be driven with consideration of relevant involved parameters and decision criteria. We proceed then as a first step by analyzing the literature, in a large term and scope, around general guidelines about automation, aiding principles, and decision criteria that are significant for manufacturers. The findings are presented in chapter 2. The chapter includes results of a general review, benchmarking, and discussions with manufacturers, 3 in France, one in Germany, and one in the United States.

Once the research area is sufficiently defined in the 2 first chapters, we perform a more tailored and specific literature review focusing on existing automation decision approaches. The review including all methods that can be found in the literature is presented in chapter 3. These methods are thoroughly analyzed from their suitability point of view, with consideration of the acquired knowledge presented in chapter 2. The analysis driven by evaluations of the methods with regard to defined requirements highlights the need to a new decision approach, which is then proposed, globally by its outline, at the end of chapter 3. The full presentation and related developments to the method is performed in the second part of the thesis.

The second part of the thesis includes then the detailed proposals allowing the new defined automation decision method implementation and computerization. Each proposal, presented in a separate chapter, includes its specific review and argued proposal.

The second part of 4 chapters, from chapter 4 to chapter 7, is organized as follows:

- A dedicated modelling language, labelled ASML as "Assembly Systems Modelling Language", for assembly processes representation using graphic rules and standardized vocabularies is presented in chapter 4.
- Assembly time estimation rules and databases allowing assembly processes time prediction with analyses and pre-balancing issues are presented in chapter 5.
- Assembly cost estimation with an integrated time-based cost model to assembly cost prediction is presented in chapter 6.
- A mathematical optimization model for assembly systems automation alternatives generation and evaluation is proposed in chapter 7. The model allows a convergence to the optimal configuration with regard to an objective function, consisting in the defined cost model, and with satisfaction of several constraints covering the multiple parameters at stake in the LoA decision.

After the presentation of detailed proposals, we present in a third and last part 2 concluding chapters with contributions sum-up in chapter 8 and future works, and openings in chapter 9.

1.6. Conclusion

This chapter shows a general initiation and introduction to the current research around assembly systems design, and more particularly automation decision making for new assembly processes to be designed. It highlights the difficult context and environment of manufacturers and their perception to automation as a powerful tool to tackle competitiveness issue and cost reduction. In this chapter we stated some feedbacks and citations showing that automation, to be beneficial, should be well designed and appropriately decided. This introduced the research question consisting in the question about how to decide about the appropriate Level of Automation (LoA) for new assembly systems design. For more clarity, we presented then definitions and scales of LoA that can be found in the literature. Then a dedicated scale for new assembly systems design and LoA decision is proposed. We finally presented the objectives of this thesis and the research methodology with the manuscript plan as well. In the next chapter, we continue in the general discovery of the field of automation decision with presentation of automation generalities, aiding principles, and involved criteria in the decision about automation.

Chapter 2. Generalities about automation and criteria identification

Abstract

This chapter belongs to the first part of the thesis around the literature and topic introduction. It aims to practically highlight the industrial interest on the topic and to present findings around general truths and best practices about automation. Advantages and drawbacks of automation and manual assembly are then studied based on the literature and industrial visits. Automation decision criteria that can be involved in the decision are exhaustively presented. The criteria identification provides a certain help to the decider as explained in this chapter. Their practical methodic consideration will be later integrated in the proposed approach through next chapters of the thesis.

2.1. Introduction

Automation is generally observed by manufacturers as an efficient tool to enhance competitiveness by reducing the cost and increasing the profitability and margins. It is also used to tackle several different problems and difficulties, such as hard environments for humans, e.g. warm, heavy, or thin parts that can be difficult to assemble. Yet, it was realized, as mentioned in the previous chapter, that using highly automated processes is not evidently the best solution. In fact, its usefulness depends on the specific case and related characteristics. Multiple criteria are involved to answer to the question, if automating the process is worth doing, and if yes, where to automate and to what extent? This decision is then complex and should be driven case by case. The complexity can be due to the multi-criteria aspect of the decision. In this chapter, we present feedbacks of experiences of manufacturers about automation justifying the fact that automation is not evidently always good solution. Then, we present based on the literature generalities about the possible advantages and drawbacks of using automation and manual that can provide a clearer idea to the reader about this research issue. We present then major impacting decision criteria that can be involved in the decision making. These criteria have to be, theoretically, entirely considered when deciding about automation.

The chapter is organized as follows: In section 2.2, we highlight in a practical way the need and importance of opting to the appropriate LoA in the process by feedbacks of manufacturers having experienced automation. In section 2.3 we present some general truths and best practices about possible advantages and limitations of automation and manual for an appropriate use. In section 2.4, we present decision criteria that should be considered in the automation decision making. The chapter is finally concluded in section 2.5.

2.2. Feedbacks of manufacturers about their experience with automation

To highlight the importance of having the appropriate automation level in the assembly process, and to orient the research to practical industrial background, we proceed by presenting some feedbacks of manufacturers having experienced automation.

According to studies presented in the first chapter, automation is much debated in industry and underlined that automation is not always satisfactory or profitable (Lay & Schirrmeister, 2001)(Gorlach & Wessel, 2008). Some case studies were performed and presented by Säfsten (Säfsten, Winroth, & Stahre, 2007) about experience of 5 companies with automation in order to study how they managed issues linked to automation decisions. The results indicate that when decisions concerning automation are made without consideration of the given manufacturer's specific context and environment, the long-term result is not satisfactory. Another study is performed in (Saunders, 2004). The author presented tests of different LoAs and processes automation for a same application in physical-chemical domain and drug research. He tested 5 technologies from the semi-automatic process to robotic and full automated with HTS (High throughput screening) technologies processes aiming the improvements in the capacity, speed and efficiency of the drug discovery processes. He presented the advantages, disadvantages, capacity and costs of each of the tested processes on his application. The results are confirming that LoA question deserve attention. In (Almannai, Greenough, & Kay, 2008), an important feedback of companies was reported indicating the need of management to be supported in improving man–machine interaction at the earliest stage of their manufacturing automation decision making process. This seeks to avoid the pitfalls of over-automation which can lead to the failure of processes to deliver cost-effective and flexible operations. It was also reported that systems in industry are rarely fully automated (Säfsten, Winroth, & Stahre, 2007).

We realize also that the need to identify the right Level of Automation and computerize the decision by several alternatives generation, testing, and comparison is also of large interest according to our industrial contacts in France, the United States, and Germany. During this project, we conducted interviews in France with an electric devices supplier and a car manufacturer. We also visited an assembly plant of painting guns in France and a plant of SUV cars assembly in the US. During previous benchmarkings and workshops in our research team, interviews were conducted with a transmissions and gearboxes manufacturer in Germany. The carried interviews were oriented to understand their decision process well as the relevant criteria for them. The results helped sketching the proposal and identifying key decision criteria. They also confirmed an explicit need to support the decision, ideally by a computerized tool.

We present in next section 2.3 a review of advantages and drawbacks of automation and manual processes as first elements that can provide a certain help around commonly admitted ideas in industry and literature concerning benefits and limitations of automation and manual.

2.3. Decision aiding principles: automation benefits and limitations

Decisions concerning automation require consideration of the possible advantages of choosing between different LoA (Säfsten, Winroth, & Stahre, 2007). In this section, we enumerate notions around possible advantages and disadvantages of automation options. These general findings and commonly shared opinions can provide a certain help to deciders. The provided knowledge may make them aware from possible risks that can be generated by automating processes, but also possible advantages of automating processes and associated appropriate situations where it can be worth doing and viable. In this point of view, we present first advantages and limitations of automated processes in first sub-section 2.3.1. Then, in section 2.3.2 advantages and drawbacks of using manual are listed. Finally, we enumerate the advantages of using a tradeoff of partial automation. This leads to hybrid automation that generally represents a good compromise for manufacturers. This is detailed in sub-section 2.3.3.

2.3.1. Advantages and limitations of automation

We proceed in this section 2.3.1 by providing benefits and limitations of automation gathered from the literature. We organize the results in Table 5.

Table 5: Advantages and limitations of automation

Advantages of automated processes	Limitations of automated processes
• **Increases productivity** and **mitigating the effects of labor shortage** (Groover, 2015)(Frohm, Lindstrom, Winroth, & Stahre, 2006) • **Improves product quality** (Groover, 2015) (Windmark, Gabrielson, Andersson, & Stoehl, 2012). • **A weapon to enhance competitiveness** on a global market due to relatively high wage costs observed in high labor countries, such as the Europe and the US (Frohm, Lindstrom, Winroth, & Stahre, 2006) (Brainbridge, 1983). • **Minimizes the number of employees** in the company (Frohm, Lindstrom, Winroth, & Stahre, 2006) and reducing labor cost (Groover, 2015). • Requires a **limited workspace** (Feldmann, Müller, & Haselmann, 1999) • Performing functions **more accurately** than human operators (Frohm, Lindstrom, Winroth, & Stahre, 2006) • **Reducing lead time** (Groover, 2015) • **Cost savings** within production (Frohm, Lindstrom, Winroth, & Stahre, 2006) • Automation gives possibilities for **higher efficiency** (Frohm, Lindstrom, Winroth, & Stahre, 2006) • **Eliminating monotonous** and **physically demanding** work situations (Frohm, Lindstrom, Winroth, & Stahre, 2006) • Execution of **impossible**, **hazardous**, or **unsafe** work, **difficult** or **unpleasant** work for humans, and **extension of human capability** (Wickens, Lee, Liu, & Gordon-Becker, 2014) (Groover, 2015). • Useful in the case of assembly of **heavy components**, **large number** of components, **different directions** in assembly operations, or in case of **high accuracy** required (Krüger, Nickolay, Heyer, & Seliger, 2005) .	• **High complexity** of the technical and organizational processes difficult to handle (Feldmann, Müller, & Haselmann, 1999)(Frohm, Lindstrom, Winroth, & Stahre, 2006)(Gorlach & Wessel, 2008). • A high LoA can lead to a **limited flexibility** leading to difficulty of customizing products (Gorlach & Wessel, 2008)(Frohm, Lindstrom, Winroth, & Stahre, 2006) (Brainbridge, 1983). • Using automation adds **maintenance** costs to the product that should be considered (Windmark, Gabrielson, Andersson, & Stoehl, 2012)(Boothroyd, Dewhurst, & Knight, 2011). • A high LoA can lead to **expensive systems** (Gorlach & Wessel, 2008) (Feldmann & Slama, 2001). • A difficulty to get **payback on investments** in automation is highlighted (Frohm, Lindstrom, Winroth, & Stahre, 2006). • A great deal of **time and cost is spent to engineer and program** robotic assembly cells (Boër, Pedrazzoli, Sacco, Rinaldi, De Pascale, & Avai, 2001). • Automated systems **robustness decreases with increasing parts tolerance** (Brainbridge, 1983). • Systems where LoA are too high present **high degree of sensitivity to disturbances** (Windmark, Gabrielson, Andersson, & Stoehl, 2012). • **Product design** should be sometimes **changed to improve feasibility** of automation (Boothroyd, Dewhurst, & Knight, 2011). • Product must be manufactured in **quite large quantities** before automation could be considered in order to be profitable (Boothroyd G. , 1987) (Boothroyd, Dewhurst, & Knight, 2011)(Windmark, Gabrielson, Andersson, & Stoehl, 2012). • **Lean awareness** should be first implemented in the company before consideration of automation (Fasth, Stahre, & Dencker, 2008)(Fasth & Stahre, 2008) • Automation is not worth doing in the case of **occasional products** over a limited time or **small batches** assembly because of a high change-over time and non-cost efficiency (Frohm, Lindstrom, Winroth, & Stahre, 2006)

It can be realized based on Table 5 that main benefits of an automated process are expected on quality, productivity, a limited space, and making possible tasks that are difficult to humans. Drawbacks can be basically summed up in a lack of flexibility, systems fragility, and the required maintenance. Concerning the cost aspect, it seems to be confusing and not clearly perceived: sometimes it is an expectation of gain and profitability margins increasing, sometimes it is the fear of expensive systems, heavy investments, and their payback risk.

2.3.2. Advantages and limitations of full manual

As previously mentioned we present in this section some advantages justifying the use of manual processes instead of automated ones. We also present limitations of such processes. We summarize the different elements in following Table 6.

Table 6: Advantages and limitations of manual processes

Advantages of manual processes	Limitations of manual processes
• **Flexibility** of humans (Feldmann & Slama, 2001) • The possibility of producing a great **mixture of different products through a short setup time** (Windmark, Gabrielson, Andersson, & Stoehl, 2012) • Reducing **process complexity** (Feldmann & Slama, 2001) • Achieving **complex operations** (Feldmann, Müller, & Haselmann, 1999) • **Feasibility** of assembly (Feldmann & Slama, 2001)	• Limited **quality level** (Feldmann, Müller, & Haselmann, 1999) • A **non-reproducible quality** (Feldmann, Müller, & Haselmann, 1999) • A satisfactory **education and life-long learning** are required (Bley, Reinhart, Seliger, Bernardi, & Korne, 2004) • **Labor skills** and related **manufacturing location** have significant impact on the assembly efficiency and quality (Gorlach & Wessel, 2008) • Manual is not recommended in the case of **hazardous environments**, **monotonous, physically demanding** operations, or **high precision** tasks over time (Frohm, Lindstrom, Winroth, & Stahre, 2006).

The advantages and limitations of manual systems found in Table 6 are coherent and complementary to results of Table 5 concerning automated systems.

2.3.3. A compromise: mixed automation

After presenting concepts and guidelines that can encourage or discourage opting for a full automated or full manual process, previously summed up in Tables 5 and 6, we present in this section the concept of a combination of these possibilities in a same process. For this combination, we distinguish 2 possible alternatives: cooperative man-machine stations or use of partial automation throughout the process by automating work zones. These 2 alternatives are tackled, respectively, in section 2.3.3.1 and 2.3.3.2.

2.3.3.1. Cooperative man-machine or semi-automated processes

These systems involve at a same time and in the same station human(s) and machine(s) to perform the assembly task(s) and require a continuous implication of both of human(s) and machine(s). These kinds of processes correspond to the Level of automation 2 of Table 4 labelled manual with automated assistance.

These systems combine the accuracy and speed of machines with the flexibility and reliability of human workers (Krüger, Nickolay, Heyer, & Seliger, 2005). It is especially useful for complex assembly and handling tasks (Krüger, Nickolay, Heyer, & Seliger, 2005). Bley highlighted (Bley, Reinhart, Seliger, Bernardi, & Korne, 2004) that the appropriate human participation during planning and execution of assembly shall be cared. It was also reported that systems in industry are rarely fully automated and a common solution used by manufacturers is to integrate manual and automated operations into semi-automated stations (Säfsten, Winroth, & Stahre, 2007).

According to our visits in assembly companies, we realize that such systems, basically of manual workers using automated tools, machines driven by humans, or robotic arms guided by humans, are very abundant in current assembly companies. We think that it can be argued, compared to fully autonomous automatic systems, particularly by the simplicity of such processes, the limited required initial investment, and their flexibility to handle multiple variants of assemblies in the same line.

This compromise of semi-automated processes combining advantages of both manual and automated systems concern a same unique station. In next section 2.3.3.2, the compromise concerns automation by work areas that can lead to partially automated processes.

2.3.3.2. Hybrid automation, partial automation, or automation by work areas

These processes that we call hybrid, partially automated, or automated by work areas consist in processes where automation is driven, when needed, by work zones throughout the process. This can lead to areas with high automation, others manual, semi-automated zones, or possibly and if needed, to full manual or full automated process where all zones are automated. In coherence with that, we would like to remind the previously asked research question of chapter 1: "where to automate or not throughout the process, and to what extent?". In fact, it may be possible, according to some criteria and to the natures of the assembly tasks to be performed, that some of the tasks should be automated, and others not. It means that some work areas should be automated and others have to be performed more manually. The result can be a hybrid automated system or partially automated according to the case study input information or key decision criteria be taken into account. The criteria will be studied in next section 2.4.

Partially automated systems are frequently used by manufacturers. Their usefulness was realized during our visits in companies where automation is analyzed and driven by work areas. In this sense, throughout a same assembly process of a same product assembly, various LoAs can be used for tasks: some in manual (such as quality inspection tasks, raw material packages opening, triage and supply), others by robots (such as painting or windshields gluing and installation tasks), and other ones by dedicated machines (such as snap fitting tasks), or semi-automated for other tasks (such as screwing or clipping, or handling and installing heavy parts such as cars dashboards assembly where robotic systems manually guided are used). The mentioned LoAs we used to describe these presented examples between brackets coincide with the 4 LoA scales of Table 4 – chapter 1.

The usefulness of hybrid automation systems is also underlined in the literature. In this sense, the results of case studies performed in (Gorlach & Wessel, 2008) showed that fully automated as well as completely manual processes are not always the optimal ones in automotive assembly. It was also shown that the fictitiously determined levels of automation consisting of automated and manual stations is a better option for the sake of economy, flexibility, complexity decreasing, quality, and feasibility (Gorlach & Wessel, 2008)(Feldmann & Slama, 2001). Groover (2015) noted that since certain elements involved in production of many types can be particularly difficult and costly to automate, the decision is often made to introduce the use of partial automation. Lindström and Winroth (2010) stated that research has shown the importance of integrating both humans and technology in manufacturing automation, thus supporting sustainable and robust manufacturing system. Sheridan (1995) also recommended finding the best combination of human and automatic control where "best" is defined by explicit system objectives. Concerning economic allocation and for equivalent performances, if the cost of technology for automating a function is higher than hiring an operator, the function is generally not automated even in the case of existence of a technical solution (Säfsten, Winroth, & Stahre, 2007). As mentioned in (Chung, 1996), an evidence from US companies indicates the importance of including human aspects when implementing advanced manufacturing technology. In (Udo & Ebiefung, 1999) and (Mital & Pennathur, 2002) the authors noted that there continue to be reports of industrial investment failures and difficulties due to the lack of appropriate man–machine combinations in the processes.

Given the multiple advantages motivating hybrid automation, it is evident that an appropriate method has to handle the possibility to propose a hybrid solution. Offering partial automation possibilities will then represent later one of the requirements of reviewing and evaluating existing decision methods in the next chapter 3 of this thesis. Moreover, when analyzing pros and cons of full automation and manual, we realize that some criteria implicitly participate to the judgment of finding a suitable solution or not. Therefore, before reaching the methods review, we focus on identifying the automation decision criteria to make them explicit.

2.4. Decision criteria identification and consideration

In the continuity of decision making background, we tackle in this section the identification of involved criteria that should be taken into account when deciding about automation. The knowledge of the decision criteria can help deciders to identify first the most preponderant and prior criteria according to the company strategy and culture. For example, ergonomics or environmental criteria can represent prior criteria for some companies, and may be neglected by others. Major companies can care about the cost as a prior criterion, and for others the quality can represent the most significant whatever the generated cost such as for luxurious products assembly. At a second time, a more thorough analysis can be driven. In that analysis, each of the operations can be studied and analyzed with regard to each of the criteria. The suitability of each of the assembly operations can be assessed with regard to the given criteria in the different possible Levels of Automation (LoA). The LoA scale described in Table 4 can be used. As a result, the study can help deciders to eliminate some LoAs for assembly operations that cannot be appropriately performed with regard to some decision criteria, or recommend others that can be realized as most favorable to selected criteria.

In Table 7, we show an exhaustive list of identified criteria that can be involved in the automation decision. The criteria were identified based on benchmarking, workshops, and discussions performed with manufacturers during this thesis and also previous works conducted by the research team with works of (Lacouture, 2012) and (Pianne, 2012).

Table 7: Automation decision criteria identification (reworked from (Lacouture, 2012) and (Pianne, 2012))

Category	Criteria	Description
Cost	Investment cost	Invest cost of operating materials and equipment
	Payback period	The period of time allowing to recover the invested amount
	Return on Investment (ROI)	Amount of money received after the project has been completed
	Workers	The number of workers required on the line/station
	Running costs	Cost regularly spent to run the line (equipment, tools, rent, etc.)
	Set-up cost	Cost spent for setting up the line
	OEE	Overall equipment effectiveness
	Expected production volume	The scheduled number of products on the assembly line
	Labor cost	The cost of workers per hour
Productivity	Lot size	The size of a batch of products
	Expected volume	The scheduled number of products on the assembly line
	Life time of product model	The time the product will be manufactured before new version
	Takt-time	Pace of the assembly line reflecting the production cadence
Quality	Number of rejects	The ratio of defects to the whole production volume
	Number of quality rejects	The ratio of undetected quality rejects (meaning found after selling)
	Number of rework of rejects	The ratio of the rework of the defects to the whole production volume
	Sensibility of the joining components	The possibility to damage the product
Robustness	Impact of defective component to be assembled in the task	The risk associated with defective components on the product
	Reliability	The failure rate of the task
	Repeatability	The variation of product quality
	Mean time to repair	Average time to repair a failed component or device
	Mean time between failures	Average time separating 2 consecutive failures
	Allowing production flow continuity	The ability to continue manufacturing a product in spite of a tool breakdown
	Likelihood of timing error	The likelihood that the timing of the task is not fulfilled
	Likelihood of sequence error	The likelihood that the task is not done in the proper sequence
	Production system compatibility	The compatibility of the task with ZF Production System Guidelines
Flexibility	Volume flexibility	The ability to operate profitably at different production volumes
	Machine flexibility	The ability, without human interference or long set-up times, to replace worn-out or broken tools
	Product flexibility	The ability to change over to produce a new product, within the defined part spectrum, very economically and quickly
	Process sequence flexibility	The ability to interchange the ordering of several operations for each part type
	Set-up time	The time needed to prepare the system to be ready for the assembly
	Number of possible variants for the product	The number of possible variants for the same product
	Postpone automation sensibility	The ability to postpone automation for a few month in order to fit the best
Plant environment	Training availability	The ability to have an available and good quality training procedure for the worker
	Worker skills	The available skills of worker on site
	Level of work experience	The required level of experience to perform the assembly task
	Available maintenance skills	The technology and the number of technicians available
	Suppliers of equipment	The number of available suppliers and their delivery time
	Support organisation	The available support in the plant

System integration	Flexibility towards IT changes	The ability to allow IT changes quickly and economically
	Information system to be connected with	The type of information system available at the location
Environmental footprint	Waste	The amount of waste created after the assembly task is performed
	Materials used	The amount of materials (e.g. water, oil) used for the assembly task
	Energy used	The amount of energy (e.g. electricity) used for the assembly task
Product design features	Size	The dimension of product
	Weight	The weight of the product
	Number of basic parts	The required basic parts for the joining function
	Tools to re-use	The ability to already have equipment that can be used for the assembly task
	Sufficient space for the joining operation	The available space for the joining operation
	Robustness of product	The ability for a product to be insensitive to risks
	Criticality of product specification	The fact that a task is critical
	Tolerance	The accepted task precision
	Likelihood of mix-up error	The likelihood that another task is done instead of the right one
	Physical complexity of task	The number of joining components for the whole product
	Joining force/joining moment	The required joining torque for joining operation
	Stable dimensions for the joining components	The ability to maintain its original dimension while being used for its purpose
	Orientation of the joining components	The orientation of the joining components before joining, i.e. how many axes are needed to place the joining component
	Joining aid on the basic part or/and joining components	The type of joining aid on the joining part and the basic part
	Bounding volume of the joining components	The fact that all the dimensions of the joining components have the same size or not
	Complexity of insertion movement	The type of insertion movement required to join (e.g. linear)
	Number of possible variants for the joining components	The ability to have different possibilities for the assembly tasks (different dimensions for nails for example)
	Number of contact points	The number of contact point between the joining component and the basic part
	Gripping surfaces on joining components	The type and dimension of gripping surfaces available on joining components for automated handling function and joining function
	Sensibility of the joining components	The possibility to damage the product
	Possibility for the joining components to get stuck or clung	The ability for a joining component to get stuck or clung with the same components
	Number of stable position for the joining components	The ability for joining components to stay motionless
	Symmetry of the joining components	The type of symmetry on the joining component
Workers condition	Safety regulation	The ability to meet requirements of safety regulations
	Social acceptability	The ability for the task to be accepted to be done by the worker
	Decision complexity	The type of decision that needs to be made on the assembly task
	Cognitive workload	The ability for the mental workload to suit the worker
	Worker "set-up" complexity learning process organisation	The ability for the worker to learn fast new tasks
	Ergonomics and physical constraint of the task	Physical constraint of the task against ergonomics and legal constraints

The results include 73 criteria organized into 10 categories or classes as shown in Table 7. These classes are as follows: cost, productivity, quality, robustness, flexibility, plant environment, system integration, environment, product design features, and worker conditions.

To simplify the decision and for a better support, the decider can start by eliminating the main categories that are not of important significance for him, for example, system integration category can be neglected for some companies due to internal specific culture or rigid strategy. In a second step, a more detailed study can be conducted by analysis of planned assembly operations for the future process. Such studies, as it will be later described through our decision method proposal outline in next chapter, can be driven by the product design and operations analyses. These operations can be assessed with regard to the criteria that the decider wants to keep. The analysis should be driven in the different LoAs. The set of possible automation alternatives can then be as a result restricted after elimination of unsuitable scenarios that are not favorable to some criteria. The previously proposed criteria consideration guidelines aim only at highlighting how the listed criteria table can contribute and aid the decision making. This does not consist in the core proposal of the thesis. Yet, it will be reminded and integrated in the main decision approach that will be proposed in next chapter 3 of the thesis.

2.5. Conclusion

This chapter started by feedbacks of manufacturers about their experience about automation. The cited testimonies confirm that automation is not always worth doing and not evidently profitable. In a next step, a review is performed to mention some generalities of best practices about automation and manual. The results are organized in tables of benefits and limitations of each of them. The presented tables, basically established from literature works, can provide a certain support and first ideas about which process can be more or less appropriate to a given case. Combination of automation and manual into a same process are studied and shows multiple practical advantages confirmed by the literature and abundantly used in nowadays manufactories. However, it is still a need to guide, when and where to use full automated, full manual, or hybrid systems. In fact, each possible technology can be more appropriate than the others according to the given case. The usefulness of hybrid systems, even if it may represent a good deal of advantages, is not absolute and depends on the given case. For example, such systems cannot be as extremely productive or of a high speed as the one of full dedicated automatic lines. It can be then unsuitable particularly in cases of very high production volumes. Consequently, the question about the decision appears to be related to the given case specificities and should be driven case by case. This introduces the need to identify decision criteria that can orient the decision about automation. An exhaustive list of all possible criteria is then presented. The list itself can provide a certain help to deciders to guide them to reason and select the most important criteria for the company according to its specificities. A more complete study and analysis can be driven in a second step by analyzing the planned assembly operations, according to the product design, with regard to prior selected criteria, and with regard to the possible selectable LoAs. This can help to intuitively, when possible, select some LoAs in the future process, or eliminating some that are not favorable to some criteria. Yet, these first ideas around the guidance to automation decision have to be integrated and computerized into an objective approach. Before proposing our approach and integrating such reasoning, we first review existing literature automation decision approaches and analyze if they propose such orientations. The review is presented in chapter 3 concluded by our proposed method.

Chapter 3. A review in automation decision literature & a new approach proposal

Abstract

After the background review performed in the previous chapters, we focus in this third chapter more straightforwardly on the core subject of the issue about the automation decision making issue. The need to support the decision is highlighted. The non-abundancy of decision methods is also underlined. The few methods found concerning the decision making are then reviewed and described. The review confirms the poor support to automation decision. For an objective evaluation of their efficiency, we evaluate these methods with regard to defined requirements reflecting the concrete need from the support point of view. The results of the evaluation are analyzed and discussed. The study led to a need to define a new approach. The methodology we define to address the identified gaps is proposed by an outline presentation of a new approach and the developments to be defined, as modules, to allow its implementation. The next part of the thesis seeks to develop the different identified modules.

3.1. Introduction

The literature in automation is almost technical and seeks to develop new technologies in production rather than to select the most appropriate automation level for a given production. Even recently in 2010, it was stated that the literature about LoA decision is not abundant and the support for making automation decisions is "poor" (Lindström & Winroth, 2010). The question of automation decision is not well documented and the path that leads to the final decision is not traceable (Ross, 2002). The decision about the processes optimal automation level, as specified in previous chapters, depends on the given case specifications, manufacturer's context, and several involved criteria. The list of criteria presented in chapter 2 can help on identifying important ones for the manufacturer. Currently, few works exist to guide deciders to the most appropriate automation level. The aim of this chapter is to present a review in the automation decision making literature.

In section 3.2, we present a review of the literature decision methods that we classify and present a description of each approach. In section 3.3, the different methods are evaluated with regard to requirements we define. The evaluation led to a lack of efficient and significant support to guide the decision about automation for new assembly systems design. This confirmed the need to build a new approach for more helpful support and to address the different defined requirements. A new approach is then described in section 3.4 by its outline and needed modules to be developed. The chapter is finally concluded in section 3.5.

3.2. A review of existing methods supporting automation decision

In this section, a literature review is made presenting existing literature methods that can be used for LoA deciding. The purpose is to review how exiting methods help the decision and select optimal automation levels, and analyze their optimization strategy.

Eleven methods were found in the literature. In this review, we classify the methods according to the method structure and reasoning approach. We identified 4 types of methods: decision flowcharts, guidelines to help and guide the decision, decision tables, and cost-based decision methods. According to this classification, the review is performed through 4 subsections in which the methods are briefly described and the way to select the levels is discussed for each decision method. For more details about the methods, interested readers are invited to find the methods' full presentation in the provided corresponding original references else the detailed review of (Salmi A. , 2013) where more details about the methods are presented and analyzed. We provide in Annex A the outlines of some decision methods that we think need to be available for readers to be easily understandable.

3.2.1. Flowcharts methods

In (Ross, 2002), a decision flowchart is defined (Annex A – Fig. 1). The method considers assembly complexity and economic feasibility for process automation possibilities analyses. The decision methodology can be of interest. Yet, it is still need a framework and further developments to be applicable. Design complexity quantification is not completely tackled. Moreover, the economic analysis or cost computation are lacking in the work.

A second decision flowchart is defined in (Konold & Reger, 2003). The decisions through the flowchart are conducted by evaluations with regard to some criteria with values or thresholds (Annex A – Fig. 2). For example: "expiry date of the products" with a threshold of 3 years or "level of difficulty of the assembly operations" with values 'difficult' or not. The decision path can lead to a unique decision among 4 possibilities: "manual" process with 2 possibilities: with manual transfer or automated transfer, "automated", or "hybrid". The decision strategy is objective and concrete. Yet, the choices and threshold values are not argued and seem to be based on experience. They cannot handle all industrial cases, various contexts, or industry evolution. The final decision is also too general as it concerns the whole process (manual, hybrid, or automated) rather than work areas automation. Few criteria are also considered with only 8 criteria taken into account. The adaptation or consideration of other criteria, such as the ones defined in Table 7 of chapter 2, needs multiple other studies to make the decision multi-criteria. This would include defining thresholds as it is done for the few considered ones, making the approach standard, and generalized for different assembly fields.

3.2.2. Guidelines methods

In this category, automation decision methods consist in guidelines aiming to guide and structure the decision procedure and methodology, basically for existing processes improvement rather than for new processes design.

A first guideline is defined in (Kapp, 1997). It consists in a simple principle labelled USA as « Understand, Simplify, Automate » for processes where the question of improvement is asked. Thus, the method does not provide a concrete support or implementable tools to objectively reach the optimal process automation configuration. Moreover, it is dedicated to increase the LoA of existing processes.

A second method is defined in (Parasuraman & Sheridan, 2000). Alternatives guiding the decision about automation are proposed with consideration of human performance consequences as a prior factor (Annex A – Fig. 3). The human factor efficiency and difficulties consideration in automation is of interest. But, other criteria, considered as secondary criteria, such as 'automation reliability' or 'costs of actions outcomes', are not practically included in the description of the method application. No optimization strategy or exhaustive search is proposed in the approach.

Most recent methods in the guidelines category are defined in (Lindström & Winroth, 2010) and (Fasth & Stahre, 2008). The two similar approaches are respectively named "Dynamo" (Annex A – Fig 4) and "Dynamo++" (Annex A – Fig. 5) where Dynamo++ represents an evolution of Dynamo. These methods consist in steps to follow, through 8 steps for Dynamo and 12 for Dynamo++, structuring the decision. The procedure is ensured by measurement of current LoA and suggestion of possible improvements by discussions and interviews to be conducted with the concerned stakeholders. The methods basically propose organizing the process of deciding rather than providing a way to find alternatives, initiatives, or guidance to practical solutions. They are also conceived to existing processes analyses and improvement proposals rather than designing a new process as it is the purpose of this research. Moreover, no technical optimization strategies are tackled through Dynamo and Dynamo++ methods.

A last guideline is defined in (Almannai, Greenough, & Kay, 2008). This method is more quality oriented. It is based on the involvement of QFD (Quality Function Deployment) and FMEA (Failure Mode and Effects Analysis) methods into a decision approach (Annex A – Fig 6). The solutions are ranked using alternatives score computation that can be of interest. Yet this can be used only to compare some interesting alternatives. In fact, when the number of alternatives is too high regarding the combinatory explosion, this becomes unmanageable. In addition, the ranking strategy itself is not detailed or tackled in the paper. The approach can be also subjective because the ranking and decision are driven by the user and may depend on his expertise or own personal analyses and preferences.

3.2.3. Decision table

In this category, a unique method is defined in (Boothroyd & Dewhurst, 1983). This method proposes a decision driven by a table (Annex A – Fig. 7) leading to an automation level for the whole process. The decision is then found in a cell of the table which corresponds to some product information (such as the number of parts in the complete assembly) and planned production information (such as the annual production volume) organized in the table's rows and columns. Due to the use of such a table in the decision with parameters and threshold values, the decision approach can be considered as objective for the automation solution finding. Yet, little information is used to guide such complex and multi-criteria decision (only 10 parameters in total). The justification of the proposed automation options, with default values basically found after long experiments and representing the core of the decision process, needs argumentation and explanation. Then, the method proposes an automation solution that concerns the whole process. Partial automation is not offered by the method as described.

3.2.4. Cost model-based methods

In this category, three approaches are found.

A first approach is defined in (Windmark, Gabrielson, Andersson, & Stoehl, 2012) proposing a cost model to compute the cost in batch productions for existing processes. The cost driven method defines the optimal LoA as a percentage computed as a ratio of the equipment costs per hour and the sum of the equipment cost and the salary costs per hour. The work proposes an exhaustive cost model. The model can help in supporting multi-criteria automation decision methods by consideration of the cost criterion computation. Yet, the unique consideration of the cost cannot be sufficient as a driver to a system automation decision. In addition, following only the cost criterion can lead to unfeasible or unsatisfactory solutions with regard to other criteria (quality, ergonomics, etc). A decision method should handle multi-criteria dimensions of the decision to be pragmatic and lead to industrially feasible and implementable processes configurations.

A second cost-based method is defined in (Boothroyd, Dewhurst, & Knight, 2011). This method associates cost equations to compute the cost for different levels of automation of processes: manual, dedicated automatic machines, and robotic assembly. The approach follows DFA analyses with time estimations for the different possible automation levels using handling and insertion operations standardized time databases. The approach is promising but multiple gaps are identified. Only independent evaluations for separate cases with different LoAs are proposed with no procedure guiding the optimization or cost minimization. No partial automation can be handled: the studies and solution concern the integral process. The way to analyze the process using only handling and insertion motions is also limited. The consideration of multiple other lacking operations, tasks, or techniques (soldering, riveting, clipping, etc) can enhance the approach applicability. The multi-criteria aspect of the decision is absent and cannot be handled in the presented methodology.

A last cost-based approach is defined in (Gorlach & Wessel, 2008) allowing, using a simple model, the assembly cost computation for the future process. This allows predicting the cost of different solutions and opting for the solution minimizing this criterion. The approach mentions the consideration of other criteria: quality, productivity, and flexibility. However, multiple limits are found in the approach. Although a 7 LoA scale is used, the approach does not tackle how the model takes into account a given level, e.g. in the cost computation. Concerning the other criteria, productivity and flexibility are integrated into one criterion. We think flexibility and productivity criteria are not evidently correlated and could be better to be dissociated. Also, the final decision making, when confusion exist (e.g. different LoAs suggested according to different criteria consideration) is not tackled. The way to conduct the final decision after analyses with regard to the decision criteria is not explained in the proposed method. The final decision seems to be manually conducted by analyses and assessments with regard to the different criteria in the various possible LoAs. This process can be feasible for few criteria as the ones considered in the paper. Yet, when considering more criteria, this quickly becomes unfeasible and needs a methodic and objective way to guide the decision. It is thus needed to structure and offer the possibility to manage the decision process, at best by a computerized strategy.

3.3. Literature decision methods analysis with regard to requirements

The aim of this section is to evaluate the LoA methods from applicability and efficiency point of views in early phase automation decision for new assembly design. To do so, we define requirements reflecting the specificities of this need then evaluate the methods with regard to the requirements. The requirements, labelled Ri, are defined and justified in a first sub-section 3.3.1 followed by their use to evaluate LoA decision methods in section 3.3.2.

3.3.1. Requirements

As our goal is to find a method for automation deciding in assembly systems design, an appropriate method should be **applicable** during the **early phase of new assembly systems design** where the system is not existent and is to be designed **(R1)**. In fact, some methods are dedicated to manufacturing and not applicable in assembly. Some of the methods are dedicated to improve LoA of existing processes, while others are dedicated to the design of new processes. A good decision method should also be **objective (R2),** with a decision rather driven by the method itself than by expert intuition. The method should be **analytic (R3)**: a low level of granularity analysis with tasks and resources detailed in order to propose sufficiently accurate solutions. It should **allow partial automation (R4),** informing where to automate or not throughout the process. The method should consider **cost** computing and minimizing because the cost is one of the most preponderant decision criteria for every manufacturer **(R5)**. The method should involve the **manufacturer context and capabilities** within the decision **criteria (R6)**. Finally, the path leading to the final decision should be **traceable** and **justifiable (R7)**.

3.3.2. Decision methods analysis

In Table 8, the LoA methods are evaluated with regard to the defined requirements. The methods are ranked in the same order as in the review of section 3.2. We also mention in the first column of the table the previously defined categories used to classify each of the methods. Class C1 corresponds to decision flow charts class, C2 to guidelines, C3 to decision tables, and C4 to cost-based methods.

Table 8: Literature decision methods requirements fulfillment

	LoA Methods		LoA methods analysis						
			R1	R2	R3	R4	R5	R6	R7
C1	***M1***	(Ross, 2002)	✓	✓	✓	✓	✓		
	M2	(Konold & Reger, 2003)	✓	✓				✓	
C2	***M3***	(Kapp, 1997)			✓	✓			
	M4	(Parasuraman & Sheridan, 2000)	✓			✓	✓	✓	
	M5	(Lindström & Winroth, 2010)			✓	✓		✓	✓
	M6	(Fasth & Stahre, 2008)			✓	✓		✓	✓
	M7	(Almannai, Greenough, & Kay, 2008)	✓			✓		✓	✓
C3	***M8***	(Boothroyd & Dewhurst, 1983)	✓	✓			✓	✓	
C4	***M9***	(Windmark, Gabrielson, Andersson, & Stoehl, 2012)		✓		✓	✓		✓
	M10	(Boothroyd, Dewhurst, & Knight, 2011)	✓	✓	✓		✓		✓
	M11	(Gorlach & Wessel, 2008)		✓	✓	✓	✓	✓	

3.3.3. Discussion: the methods requirements fulfillment

Based on Table 8, it can be seen that no method is fulfilling all requirements. Most promising ones are methods M1, M10, and M11. Method M1 is fulfilling 5 requirements. Yet, it is only an outline (belongs to the flowcharts class – section 3.2.1) and the way to apply it is not presented. In fact, it is based on cost minimization to assess the effort to automate operations. Nevertheless, the cost model to be used is not detailed. It is also involving too few criteria (R6). Method M10 is interesting because of its analytic way of analyzing assembly operations with time estimation. But, it neglects providing suggestions for partial automation (R4) and it is not involving criteria which concern the manufacturer itself and his capabilities (e.g. potential for investing, expertise, or technical preferences). For method M11, we guess it is of interest even if it is dedicated to existing processes. It is valuable in deploying the idea of computing the cost for different alternatives with a simple model. It supports an objective evaluation and adopts an analytic way to combine several parameters involved in the cost related to product and production, with some manufacturer criteria such as the location, labor skills, experience, and resulting quality. Yet, only four decision criteria are considered: cost, productivity, quality, and flexibility. And only the cost analysis is well defined. In fact, the quantification, evaluation, and integration of the three remaining ones are not explained.

For these encountered LoA methods (M1 to M11), a lack of visibility about the physical process representation is noted. In fact, representing the assembly sequence with consideration of the product design features can help designing the assembly system (Homem De Morello & Sanderson, 1991). Moreover, no method takes into account the possibility of generating different alternatives and evaluating them. In addition, the assembly sequence may be developed essentially independently of the technology choices (Homem De Morello & Sanderson, 1991). Few LoA criteria are considered in the existing methods while we identified about 73 criteria influencing the automation decision as presented in chapter 2 and Table 7.

The reviewed methods are globally lacking of traceability of the decision process. No computerization can be possible for most of the methods, with no possibility to compare or evaluate different alternatives of assembly systems.

Yet, the review in automation decision literature shows multiple encountered principles that can be of high interest such as: product design and assembly complexity consideration in automation decision (M10) and high involvement of planned production information generally associated to cost computation and solution profitability evaluation (M9). The exhaustive evaluation of all possible alternatives (M11) to guarantee the solution optimality represents another interesting attempt. The existing LoA decision methods need also a computerized procedure to guide the generation of alternatives and their evaluation. Moreover, few criteria are considered. The need to enable handling a more extended set of criteria is highlighted.

A need to define a new method providing a way to decide and compare alternatives for new assembly systems design is arising. The method should fulfill the requirements of section 3.3.1. To our understanding, the method should use process modelling with a possibility to generate and evaluate alternatives with regard to LoA criteria to be considered during the decision. The focus is to base the reasoning on the analysis of the product design, the feasible assembly sequences and the planned production context and features. We present in next section 3.4 more details about a new method we propose, including these mentioned aspects and principles.

3.4. A new decision method proposal

The existing decision methods evaluation, as discussed in previous section 3.3, revealed that no method is fully satisfactory for an efficient and concrete support to automation decision and underlining a need to define a tailored new method addressing defined decision requirements. In this section, we define a new decision method based on the previously performed reviews and discussion results. This method aims at addressing the defined requirements reflecting how a suitable approach should be. We proceed first by proposing the decision outline of the method in a first sub-section 3.4.1. Then, the required developments for the method implementation are defined and explained in a second sub-section 3.4.2. The aim of the thesis will then be, through the second part of the manuscript, to develop the defined modules to concretize the proposed decision methodology and make it implementable and computerizable.

3.4.1. The decision method outline

In this section, our LoA decision methodology is proposed by its outline. The method is defined in such a way that the whole requirements previously presented in section 3.3.1 can be satisfied. The decision methodology is represented in the scheme of Figure 2.

The approach starts with a graphic model of the assembly process (bloc 2) based on the product design analysis (bloc 1). This model should be standard and generic to allow assembly systems automation alternatives definition or generation (bloc 3) based on the generic process model. When defining these alternatives, decision criteria have to be selected and taken into account, such as quality, plant environment, or workers conditions (bloc 4). The criteria consideration will allow authorizing, imposing, or forbidding certain automation choices. This mechanism was previously mentioned in chapter 2. It should be possible to include the several criteria defined in Table 7 of chapter 2 for suitable alternatives definition. This has a consistent link with alternatives feasibility for a given production context (e.g. quality, plant environment, or assembly cost threshold). Manufacturer choice and best practices should be also taken into account by eliminating or imposing some choices to avoid unsatisfactory solutions. The definition of a coherent alternative should be also performed with an appropriate resources dimensioning to fit productivity requirements by involvement of planned production information (such as the planned volume or production life) and alternative time prediction possibilities (such as the process cadence or takt-time computation) (bloc 5). As the cost is one of the most preponderant decision criteria, and once time estimates are available, a cost per product should be computed using an appropriate early phase cost model (bloc 6). Other indicators can be computed in this step, such as process required surface estimation, investment, energy, resources workload and margins, and so on. This process is performed in an iterative way considering designers feedbacks. The saturation of alternatives generation or in case of an enough satisfactory solution found by the means of the loop described in Figure 2 should lead to the optimal or satisfactory alternative keeping. The solution consists in a trade off with regard to multiple criteria and performance indicators (bloc 7). In the case of non-feasibility, non-profitability, or any kind of non-satisfactory solutions, a feedback is then provided to the product designers to try to improve its design and the easiness or cost to assemble (back to bloc 1) using approaches such as DFA rules.

It can be realized that for a practical implementation of the outline, different developments are required. These developments are identified and enumerated in next section 3.4.2.

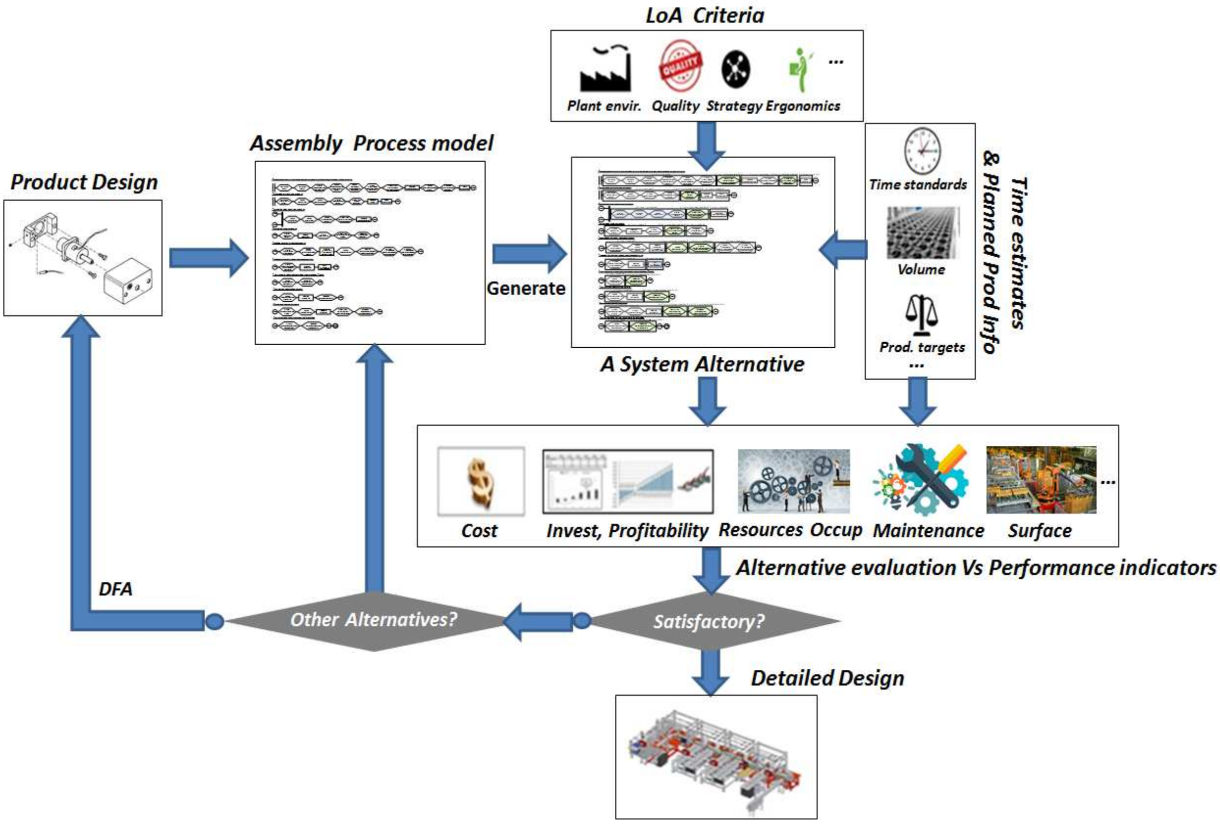


Figure 2: The automation decision making outline

3.4.2. The needed modules to implement proposed approach

We identify in this section the needed developments for the proposed approach implementation. These developments are here described as modules to be separately proposed in details through chapters of the next second part of this thesis:

- **Assembly modelling**

A modelling language is needed to represent the generic model of the process. It should allow describing automation alternatives to be defined or generated based on the initial generic one. The representations should be standard from modelling rules and vocabularies point of views to easier the computerization and unique genericity of solutions.

- **Time estimation rules and databases**

To be able to time estimate an automation alternative, rules for time estimation, consistent with the modelling language, have to be defined. These rules should be associated to databases of time estimation with consideration of the different possible automation levels corresponding to the LoA scale to be used.

- **A cost model**

An early phase cost model is needed to allow assembly cost estimation for a given alternative with selected automation options. This cost model should consider the issue of automation decision specificities and early phase constraints.

- **A model to generate automation alternatives**

To generate the different alternatives, evaluate each of them, and converge to the optimal solution, an optimization model is needed. This model should implement the whole optimization loop described in Figure 2.

3.5. Conclusion

The review in automation decision literature shows few methods guiding the decision about automation, particularly for new systems design issue. These methods seem to provide a poor support to deciders. Few criteria are considered in these methods. The need to enable handling a more extended set of criteria is underlined. The different methods were presented and evaluated with regard to defined requirements built to objectively reflect how an appropriate method should be.

The lack of a satisfactory method fulfilling all the defined requirements led to a new method proposal. The method is here proposed by its general outline. To make it practically implementable, some developments need to be performed. The identified developments at this stage concern: a modelling language to define a generic representation of the assembly based on the product design then to represent automation alternatives of assembly systems, time estimation rules and databases to estimate automation alternatives assembly time, a cost model to predict assembly cost, and an optimization model to perform the automation alternatives automatic generation and the loop computerization.

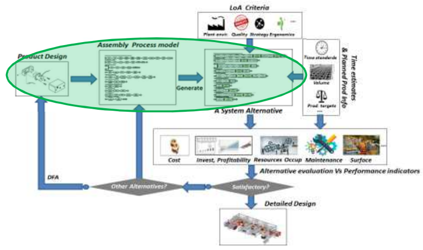
The proposed approach is defined so that it can address the different requirements that existing methods fail to fully satisfy. To anticipate this and make sure that the method, once all modules developed, can fulfill all the defined requirements, we check in upstream how the method can satisfy the requirements:

- To be applicable at the early phase (requirement R1), the approach should be coherent with this constraint.
- The modules, particularly modelling and cost estimation should take into account this constraint. The approach reasoning is a priori objective (requirement R2), the cost approach should be then also objective. Intuitive or analogical approaches should then be avoided.
- To be analytic (requirement R2), the modelling language to be used should be of a low granularity layer of description so that time and cost estimation can be analytic too.
- The modelling language should also allow partial automation (requirement R4). It should represent automation levels per resource of the process rather than a global description of the assembly. Details of resources and associated operations should then be represented.
- Once a cost model is associated to the approach, requirement R5 will be immediately fulfilled.
- The consideration of the manufacturer context and capabilities (requirement R6) should be satisfied when considering the decision criteria.
- Finally, the set of developments and the global described reasoning of optimization provide a justification and traceability (requirement R7) of the proposed solution. This is ensured by reporting all associated performance indicators and the guarantee of optimality that should be verified by the optimization module.

This upstream prediction of the proposed method performances is consequently promising and makes the development of the different identified modules worth doing. Once all modules are available, the method should then be industrially applicable.

Consequently, we develop through the next second part of the thesis the different modules to make the approach computerizable and applicable for industrial use. Because of the multidisciplinary aspect of these developments and their consistency as well, each of the modules will be independently treated in a separate chapter accompanied by a review in the associated field. Part 2 will then start by the first module around assembly modelling.

PART 2. THE PROPOSALS PRESENTATION AND IMPLEMENTATION



Chapter 4. An Assembly Modelling Language[1] with standardized Vocabularies[2]

Abstract

After presentation of the proposed decision method outline in the previous chapter concluding the first part of the thesis, we proceed in this second part by defining the identified modules allowing the implementation of the proposal. The first identified module consists in the process modelling using first a generic representation of the process and then automation scenarios generation. This chapter aims at reviewing existing modelling languages that can provide this possibility. The review led to a proposal of a new language tailored to the LoA issue labelled "ASML" as "Assembly Sequences Modelling Language". The language integrates different features of existing literature languages. To the proposed ASML, a standardized vocabulary of assembly motions is associated with a proposed extension. A second high layer vocabulary, to be connected to the first one, is also defined. The aggregation of modelling language and vocabularies addresses the need of generic standardized representation of the assembly process based on the product design analysis and automation scenarios generation.

4.1. Introduction

In the literature, few automation decision methods discuss about how explicitly to integrate the assembly sequence as a factor in the LoA decision making process. The dependency between the product design, the assembly sequence, and the assembly system design has been recognized as significant (Homem De Morello & Sanderson, 1991). In addition, the assembly sequence may be developed essentially independently of the technology choice (Homem De Morello & Sanderson, 1991). Therefore, the assembly representation can be independent of the automation levels. The sequence representation is viewed as a reliable basis to initiate the analysis of the right LoA to be implemented. Through such a representation, LoA criteria (section 2.4 – chapter 2) can be integrated. The process representation can also provide better visibility of the set of assembly tasks. Thus, we argue that a huge number of combinations of hybrid automation systems may be missed because of a lack of a suitable model to handle the problem complexity. Representing the assembly sequence with a high visibility, clarity and flexibility in showing the assembly scheduling, resources management, and allocations might enable exploring multiple or all the possible systems for a given assembly sequence. Several modeling languages used and others usable in assembly representation are found in the literature. In this chapter, we present and discuss these languages from their suitability point of view to satisfy this need of such representation for the LoA decision support issue.

[1] Assembly modelling is also presented in the following article: (Salmi A. , David, Summers, & Blanco, A modelling language for assembly sequences representation, scheduling and analyses, 2014)

[2] Vocabularies for assembly modelling are also presented in: (Salmi A. , David, Blanco, & Summers, Standardized Vocabularies For Assembly Systems Modelling and Automation Alternatives Description, 2016)

Some commercial tools in the field also exist. We can mention industrial tools for assembly systems modelling for example PLM solutions such as Tecnomatix. This tool offers a complete graphic environment to design assembly systems with advanced product information databases. Yet, such tools are used to design in details a system previously decided and for which architecture and LoA are already fixed. In our case, we are located on a step anticipating this phase of detailed design and we seek to provide the conceptual design of the line that will be later designed in details then implemented. We assume that the success of the assembly system design is highly conditioned by the ability to handle the whole assembly process and the ability to generate multiple alternatives for its implementation. This leads to the necessity to obtain, at an early stage, a flexible and technology-independent first generic representation of the assembly process then to represent resources with LoA possibilities as automation alternatives. Our purpose is then to identify or obtain a flexible modeling language to represent the assembly architecture, resources, and technologies affectation. Further, the assembly standardized representation can enable assembly time and cost estimations which will be later studied in next chapters 5 and 6. Involving the assembly sequence representation, scheduling, time estimation, and cost computation in the LoA selection process will yield decisions that are more concrete, objective, analytically supported, and optimizable. To achieve this goal, we formalize in section 4.2, the requirements for an appropriate assembly representation to the LoA decision issue. In section 4.3, a literature review of assembly modelling languages is presented with languages classifications, descriptions, and evaluations with regard to the defined requirements. The review led to a need to define a language tailored to LoA. A proposed new language, labelled ASML, is proposed in section 4.4 associating graphic rules and a literature vocabulary of standardized assembly motions. A second higher layer vocabulary of assembly tasks facilitating alternatives definition and problem solving, to be associated to the low layer one, is proposed in section 4.5. An overview with use instructions to guide the use methodology of ASML and vocabularies of tasks and motions for LoA decision issue is defined in section 4.6. Developments that are enabled by this proposals and other that can be performed are mentioned in section 4.7. A validation example using ASML model with the 2 associated vocabularies for the sake of automation decision is presented in Section 4.8. The chapter is finally concluded in section 4.9.

4.2. Requirements for a suitable modelling language to LoA decision

Analyzing the needed knowledge on the assembly sequence, we define requirements and specifications for an appropriate representation of the assembly process to the automation decision issue. A suitable modeling language should satisfy the defined requirements. Moreover, it should be coherent with the proposed LoA decision approach of chapter 3.

We define then requirements that describe an appropriate language to the issue. We organize the requirements into 3 classes: the first class (requirement 1.1) concerns the process representation form that we need in our LoA decision methodology. The second class (requirements 2.1 to 2.5) is about the data and information that should be provided by this representation. The third class (requirements 3.1 and 3.2) concerns how the representation should support the alternatives definition and their sharing between the company's actors that can be involved in the decision. This class includes also the manner of generating the different scenarios and alternatives to be analyzed. The modeling language should support the ease of switching from a scenario to another. This ability is expected regarding the high number of possible solutions, the different possible scheduling, and the combinations of LoAs affectations to be tested and assessed. The different mentioned requirements are detailed as follows:

- ***<u>Class 1: A Graphic representation for better visibility and understandability of the Assembly Process</u>***

<u>Requirement 1.1</u>: A Graphical representation of the assembly sequence

For better visibility and pattern identification to support automation selection rules, the needed assembly modeling language should be graphical. The representation should support the decider in handling the assembly sequence complexity.

- ***<u>Class 2: A representation providing primordial information needed in the LoA decision making</u>***

<u>Requirement 2.1</u>: Serial/Parallel execution representation of assembly tasks with existence/ absence of precedence constraints

The modeling language should allow showing the execution of technically nondependent assembly tasks with no precedence constraints in parallel and dependent tasks in serial.

<u>Requirement 2.2</u>: Resources Management availability and allocation representing

The modeling language should allow clearly showing the resources allocation and the technical constraints between resources (e.g. shared workspace, conflicts management, collision avoidance,.).

<u>Requirement 2.3</u>: Conditions for transiting from a task *i* to a task *j*

The modeling language should allow representing the transition conditions between a task *i* and its successor *j* in the sequencing. These conditions should consider the end of *i* and the required material/tools for executing the task *j*

<u>Requirement 2.4</u>: The final product assembly nomenclature representation

After representing the assembly parts process through the tasks sequencing, the final product assembly should be visible. The language thus should support handling the complexity of the assembly sequence.

<u>Requirement 2.5</u>: A standard representation mean, independent from product type or context

The modeling language should allow the representation of assembly sequences whatever is the product type. This representation should be unique and every person who uses this modeling language should provide a same representation for a given assembly sequence to represent.

- ***<u>Class 3: A sharable and evolutive representation in the LoA decision environment</u>***

<u>Requirement 3.1</u>: An efficient tool for Collaborative and Concurrent Engineering design process

The modeling language should offer a clear, simple, and easily understandable and analyzable description for a collaborative engineering process where different actors with heterogeneous competences are involved and collaborating within a same project.

<u>Requirement 3.2</u>: The Ease of generating different solutions, scheduling, and their required resources

The modeling language should offer the possibility to easily define or deduce the required resources for the assembly process for different scheduling and their associated allocations and assignment to assembly operations. The representation should be easy to be manipulated in order to define various solutions and scenarios.

Table 9: Literature review & classification of modelling languages used in assembly

Modelling languages used in Assembly Representation						
Purpose / Domain	***Representation*** — *Artifacts* — *Node*	*Arc*	*Type (Static/ Dynamic)*	***Assembly sequence deduction***	***Possibility of using procedures / referencing***	***For more details, description, and applications***
Product design hierarchy representation	***Scenegraph***					
	Parts, sub-assemblies, final product	Relations between parts	Static	Indirectly: effort of interpreting the graph to build assembly sequence	Possible	(Banerjee & Banerjee, 2000)
	Hierarchical Structure Model					
	Parts, sub-assemblies, final product	Relations between parts	Static	Indirectly: effort of interpreting the graph to build assembly sequence	Possible	(Dong, Tong, Zhang, & Dong, 2005) (Niu, Ding, & Xiong, 1987)
	Hierarchical Relation Graph					
	Parts, sub-assemblies, final product	Relations between parts	Static	Indirectly: effort of interpreting the graph to build assembly sequence	Possible	(Niu, Ding, & Xiong, 1987)
Product design parts relations types representation	***Relational Model***					
	Part entities in, rectangles, contact entities in circles, attachment entities in triangles	Relations between parts	Static	Indirectly: effort of interpreting the graph to build assembly sequence	Not Possible	(Homem de Mello & Sanderson, 1991)
	Graph of Connections					
	Parts to assemble	Relations between parts	Static	Indirectly: effort of interpreting the graph to build assembly sequence	Not Possible	(Homem De Morello & Sanderson, 1991)(Homem de Mello & Sanderson, 1991)
	Connection Semantics Based Assembly Relational Model (CSBARM)					
	Parts to assemble in rectangles, connectors (screws..) in ellipses	Relations parts-parts, connectors-parts	Static	Indirectly: effort of interpreting the graph to build assembly sequence	Not Possible	(Dong, Tong, Zhang, & Dong, 2005)
Assembly all possible sequences representation	***Precedence Graph***					
	Parts to assemble	Order of adding parts/ choice for arcs exiting a same node	Static	Indirectly: the graph gives differnet possible sequences	Possible	(Boothroyd G. , 2005)(Niu, Ding, & Xiong, 1987)
	AND/OR Graph					
	Parts, sub-assemblies, final product	Parts composing sub-assemblies/ final product	Static	Indirectly: the graph gives all possible sequences	Possible	(Homem De Morello & Sanderson, 1991)(Homem de Mello & Sanderson, 1991) (Homem De Morello & Sanderson, 1990)
	Parts Tree					
	Parts, sub-assemblies, final product	Decomposition or composition of sub-assemblies, final product	Static	Indirectly: the graph gives all possible sequences	Possible	(De Fazio & Whitney, 1987)
Assembly specific sequence representation with limited flexibility of sequencing	***Connection Semantics Based Assembly Tree (CSBAT)***					
	Parts	Order to adding the part to the assembled sub-product or final product	Static	Directly: the parts to add to the assembly are fixed but the order of assembling the sub-assemblies is not: no rigid sequencing	Possible	(Dong, Tong, Zhang, & Dong, 2005)
Assembly specific sequence representation	***Liaison Diagram***					
	Parts	Relations between parts	Static	Directly: the graph gives all the possible sequences but following the numbered arcs it leads to a unique sequence	Not Possible	(De Fazio & Whitney, 1987)

Modelling languages used in other fields but may be used in Assembly Representation							
Purpose / Domain	**Representation**				**Sequence deduction**	**Possibility of using procedures / referencing**	**Use/description in the literature**
	Artifacts			**Type (Static/ Dynamic)**			
	Step	**Transition**	**Arrow**				
Automatisms modeling, Programming PLCs (programmable logic controllers)	**Sequential Function Chart (SFC)**						
	Actions	Conditions	Link step-transition	Dynamic	Directly: Only one sequence is represented	Possible	(Bauer, Huuck, Lukoschus, & Engell, 2004)
Purpose / Domain	*Representation*				*Sequence deduction*	*Possibility of using procedures / referencing*	*Use/description in the literature*
	Artifacts			*Type (Static/ Dynamic)*			
	Place	*Transition*	*Arrow*				
Dynamic behavior of Processes presenting discrete events evolution: Industrial systems, transport, Telecommunications,.	***Petri Net***						
	Workstations, subsystems, functions, resources or tasks	Conditions	Link Place-transition	Dynamic	Directly: but no dedicated symbols are used to isolate the part of the model showing only the actions.	Possible (Depending on the used Petri Net Class)	(Murata, 1989) (Ahmad, Huang, & Wang, 2011) (Cecil, Srihari, & Emerson, 1992) (Zhang, Freiheit, & Yang, 2005) (Pang, Fang, Li, & Yang, 2011) (Zha, Du, & Lim, 2001) (Zha, Lim, & Fok, 1998) (Hsieh, 2006)
Purpose / Domain	*Representation*				*Sequence deduction*	*Possibility of using procedures / referencing*	*Use/description in the literature*
	Artifacts			*Type (Static/ Dynamic)*			
	Rectangle		*Arrow*				
Project Schedule	***Gantt***						
	Tasks		Precedence constraints between tasks	Static	Directly: a specific and accurate sequencing is represented	Not Possible	(Clark & Gantt, 1923)

4.3. A review in assembly representation modelling languages

In this section, we review the literature on modelling languages and study their suitability to LoA decision issue. The review and analysis aim at searching for a modeling language that fulfills the requirements and that can be consequently suitable to the LoA issue. The review results are presented in section 4.3.1 followed, in section 4.3.2, by the languages evaluation with regard to the previously defined requirements of section 4.2. We present in these sections in brief the major findings. More details can be found in (Salmi A. , David, Summers, & Blanco, 2014).

4.3.1. Literature modelling languages review and classification

In the assembly representation field, several modeling languages and methods are used. Some of them concern the product hierarchy (from which assembly sequences may be directly deductible). Others are dedicated to represent assembly sequences. We review all possible languages candidates that can be used to model processes for LoA decision.

This review includes modeling languages used in assembly and other fields that seem applicable as well. The reviewed modeling languages are shown in Table 9. The table is divided in two main sections identified by black colored cells. The first section concerns languages dedicated to assembly. The second concerns other domains representations. A second classification concerns the languages according to their purposes and domains (first column of Table 9). For each language of the table, we present in main column 2 how it represents the process: the language's artifacts (sub-column 2.1) with their roles (nodes, arcs, steps, transitions, arrows) and the representation type (sub-column 2.2) which may be a static or a dynamic. The found representation methods are also evaluated with regard to how the assembly sequence is deduced from the representation (main column 3 of Table 9). This has been the first inspected aspect since representation methods used in assembly are generally dedicate to represent the product hierarchy and how the product can be assembled in different ways. The possibility of using a referencing system for sub-assemblies representation is also shown in the table (fourth main column). This criterion is important for us because allows handling complex representations. The referencing will participate to simplify representations and reduce risk of error when modifying the representation or the sequence. Finally, for each language, references are provided in last 5th column of Table 9 for more details about the reviewed modelling languages, further descriptions, and applications.

It can be seen based on Table 9 that for almost reviewed methods, the determination of the assembly sequence from the provided representation is not always evident and may require an effort. This may be explained by the fact that the majority of the representations dedicated to assembly are built for the sake of representing the product architecture and how the parts are organized (Table 9). An exact assembly sequence with fixed order of operations, or schedule, is then deductible from these representations at a second step. Generally assembly languages are defined to represent all the possible assembly sequences. This is the case for all the found methods used in assembly except for the Connection Semantics Based Assembly Tree (CSBAT) method (Dong, Tong, Zhang, & Dong, 2005) and the liaison diagram method (De Fazio & Whitney, 1987) for the dedicated languages to assembly (Table 9).

An important principle found in most of the reviewed representation methods is the possibility of using a referencing system in representing the assembly sequence leading to the final product assembly. This system, inspired from computing principles and the use of procedures, allows referencing sub-assemblies in a graph assembly in order to define its assembly sequence independently.

4.3.2. Literature analysis with regard to the suitability to LoA issue

After the review and first feedbacks from the literature, we check how the existing languages fit the requirements that previously defined in section 4.2 to describe the expected modeling language to be successfully used for our proposed LoA decision methodology. We proceed first in section 4.3.2.1 by evaluating the different methods with regard to the defined requirements. In section 4.3.2.2 the suitability is analyzed by interpretation and most promising languages identification. A discussion is then tackled in section 4.3.2.3.

4.3.2.1. Literature languages requirements fulfillment

The reviewed methods evaluation with regard to the defined requirements is presented in Table 10. In this table the rows contains the representation methods. The requirements are shown in the table columns. The evaluation is performed as follows: the methods may strongly fulfill the requirement (filled circles), weakly fulfill (unfilled circles), or do not fulfill (empty cell).

4.3.2.2. Literature languages suitability analysis

For the first class (C1) of defined requirements, it can be seen based on Table 10 that the requirement 1.1 is fulfilled by all of the languages. This can be explained by the fact that assembly representations and models are generally graphical.

For the second class of requirements about the information we need for the purpose of LoA decision, the fulfillments are little satisfactory by the majority of the languages. Namely, the Req. 2.2, about the required resources representation, is only fully fulfilled by Petri Nets, and the Req. 2.3 about the conditions for transiting from a step i to its successor j, is only fulfilled by Petri Nets and SFC. This may be explained by the fact that they are the only ones based on transition and conditions representations (Req 2.3) and the only dynamic models. SFC and Petri nets conditions may be used in order to show the material availability to execute tasks, but the structure is not dedicated to such purpose and should be used with another meaning. These fulfilled requirements are important because they reflect primordial information we need to find in an assembly representation and represents a specificity of the LoA decision.

Table 10: Literature modelling languages requirements fulfillment

	The Modeling Method Requirements							
	C 1	***C 2***					***C 3***	
	Req 1.1	***Req 2.1***	***Req 2.2***	***Req 2.3***	***Req 2.4***	***Req 2.5***	***Req 3.1***	***Req 3.2***
Scenegraph	●	○			●	○	●	
Hierarchical structure model	●	○			●	○	●	
Hierarchical relation graph	●	○			●	○	●	
Relational model	●	○			○	○	●	
Graph of connections	●	○			○	○	●	
CSBARM	●	○			○	○	●	
Precedence graph	●	○			●	●	●	
AND/OR graph	●	○			●	●	○	
Parts tree	●	○			●	○	●	
CSBAT	●	○			●	○	●	
Liaison diagram	●	●			○	○	●	
SFC	●	●		●		○	●	
Petri Net	●	●	●	●	●	○	○	
Gantt	●	●	○			○	●	○

- **Petri Nets as the most promising: thorough review of the language**

Regarding Table 10, Petri Net seems to be the most suitable candidate for an assembly process representation according to our requirements. This modeling method is fully fulfilling the first and the second class of requirements, except to the requirement 2.5. This may explain the extensive use of this graphic tool in modeling and analyzing Manufacturing systems (Ahmad, Huang, & Wang, 2011). Requirement 2.5 is partially fulfilled by Petri Nets because it is hard to guarantee a unique representation of a process with Petri Nets. For the third class of requirements, expressing our LoA decision context, Petri Nets are less efficient. In fact, it is partially fulfilling the requirement 3.1 because of its complex representation hard to handle for new users, and where resources are not always visible in the same manner (Ahmad, Huang, & Wang, 2011). It can be obviously admitted that Petri Nets are not efficient in a collaborative design and decision discussion involving people with few experience of this formalism. For a remark, we can mention that the only other language which do not fulfill this requirement as Petri nets, is the AND/OR language, which can similarly lead to complex and crowded representations, with lot of nodes and arcs especially for complex products leading to different possibilities of assembling parts.

Petri Nets are also not satisfying the requirement 3.2 because they do not offer an easy way to generate scenarios with resources modelling and scheduling. This can be due to the structural inflexibility of Petri Nets models (Zha, Du, & Lim, 2001) and the difficulty of adapting or modifying a scenario to another one.

The allocation of resources is also a combinatorial problem itself (Zhang, Freiheit, & Yang, 2005). It was noted in (Zhang, Freiheit, & Yang, 2005) that Petri Nets based scheduling is not always satisfactory because it results in a combinatorial explosion according to the problem size. This is also more complicated in our context of LoA decision if we consider that each resource could be of different LoAs (4 levels).

Resources in Petri nets are modelled using additional tokens or places. This requires generating a new graph each time we consider a new scenario of assembly alternative with different organization and resources allocations. This would add complexity in the LoA decision process where several scenarios and possibilities of resources with different technologies (Manual resource, Automated, or robotic) are considered. We admit the possibility of scheduling using Petri Nets with advanced frameworks and derived classes as timed petri nets or knowledge petri nets, even if standard Petri Nets do not allow scheduling. In fact, Petri Nets are used for the modeling of generally existent manufacturing system or sufficiently known ones from a concept point of view (Ahmad, Huang, & Wang, 2011). The main power of this language as a mathematical tool is its support for analyzing the dynamic behavior of a manufacturing alternative or scenario (Ahmad, Huang, & Wang, 2011). Petri Nets have been already used for describing process plans (Cecil, Srihari, & Emerson, 1992) (Srihari & Emerson, 1990), but this is useful for a given unique scenario to study. Moreover, the scheduling is not graphically visible.

Many attempts have been made to extend and modify conventional Petri nets to enhance their modeling power (Zha, Du, & Lim, 2001) and its convenience for modeling production systems (Zha, Lim, & Fok, 1998). This resulted in numerous variations such as colored Petri nets, control nets, timed Petri nets, object Petri nets (Zha, Du, & Lim, 2001)(Zha, Lim, & Fok, 1998), and parallel process nets with resources (Ahmad, Huang, & Wang, 2011). The 'theoretical' schedule using Petri Models is generally managed by Timed Petri Nets, where time information is shown in the Places as P-timed Petri Nets or in the transitions as T-timed Petri Nets (Zhang, Freiheit, & Yang, 2005). Other frameworks managing time and sequencing are Knowledge Petri Nets (Zha, Du, & Lim, 2001)(Zha, Lim, & Fok, 1998) using object-oriented techniques where one of the attributes is the tasks starting time or/and durations. These frameworks of discrete simulation tools can be used when the assembly process is known or approximately known, when the scenario is fixed and have to be optimized, or when the process exists and has to be performed, simulated, enhanced, or balanced.

In our case such structure cannot support several scenarios generation, to decide, and to evaluate them in the LoA decision process. At our stage in the engineering process of new systems design, we need a modeling tool offering a better visibility and possibility to easily switch from a scenario to another without computing each time a completely new graph. Nevertheless, these discrete simulation tools can follow our focused stage for the sake of simulating, evaluating, and balancing the designed process and the final optimal scenario we determine.

For the selected scenario or scenarios, it would be useful to use a simulation tool as Petri nets or industrial software event simulation tools such as Arena (Kelton, Sadowski, & Zupick, 2010) in order to evaluate more accurately solutions (for example including batches and stocks management).

- **Interesting features from Gantt chart**

It can be of interest to mention that requirement 3.2 about the ease of generating alternatives, to schedule, and to assign resources is not fulfilled by the reviewed modeling languages except the Gantt method, fulfilling it partially. In fact, when considering a serial execution in an assembly tasks schedule, we can imagine a same resource executing these tasks. But when dispatching a serial execution to two sub sequences of non-dependent tasks executed in parallel and at a same time, we can consider at least two required resources in order to be able to accomplish this parallel execution. This possibility of scheduling offered by the Gantt makes it an interesting tool to quickly generate completely different scenarios according to the available resources. The Gantt method is in fact tailored to manage parallel dependent tasks execution by showing lags and delays. This dependency between the assembly representation and the corresponding resources is for us very important in the LoA decision. The Gantt principle can be very interesting for our LoA approach, but unfortunately this language is not sufficiently efficient considering the other requirements defined for the LoA issue as it can be observed in Table 10.

4.3.2.3. Discussion: the need to define a dedicated language

It can be realized, based on the review, that no existing method fulfills all the defined requirements for the stake of LoA decision. Some methods are more interesting and suitable than others, but no fully satisfactory method was found. Even if some of them may be able to satisfy partially the majority of the requirements, such as time-based Petri Nets, no tool offers a satisfactory simplicity, visibility, and adaptability in the representation, scheduling, and resources allocation.

Consequently, based on these results and literature analysis, the need to define a new modeling language for assembly sequences representation is confirmed. The modeling language should benefit from the existing modeling approaches features. It has to combine identified interesting principles characterizing the most appropriate reviewed languages, such as Petri Nets, SFC, Gantt, and precedence graphs. The purpose is to define a modeling language offering a suitable representation facilitating the issue of LoA decision for assembly systems design.

4.4. Assembly Sequences Modelling Language (ASML) proposal

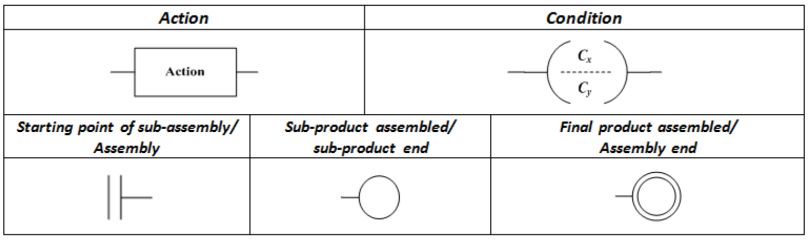
The proposed modelling language is fully defined in (Salmi A. , David, Summers, & Blanco, 2014). The language is labelled ASML, as “Assembly Sequences Modelling Language”. It is defined as a graphic modelling language dedicated to assembly sequences and processes representing. ASML is built to be dedicated to the sake of LoA decision. The language aims at making possible to easily representing different automation alternatives of systems with various possibilities of automation options based on a generic model to be defined from the product design. The model definition is helped by the previously defined requirements that it has to meet. In this thesis, we provide main features and principles of the language. The exhaustive details about the language and rules are available in (Salmi A. , David, Summers, & Blanco, 2014).

4.4.1. Basic elements of the language

ASML uses specific symbols and rules for modelling. In fact, it consists of a block scheme representation for actions and conditions (Table 11). Actions consist of elementary assembly motions using a list of a standardized vocabulary of assembly motions that will be detailed in section 4.4.3. Conditions consist in the different circumstances and transitions terms required for immediate next and previous actions through the represented sequence (Salmi A. , David, Summers, & Blanco, 2014).

An ASML sequence begins with a starting point and ends with an ending point of a sub-assembly or of the whole final product assembly (Table 11).

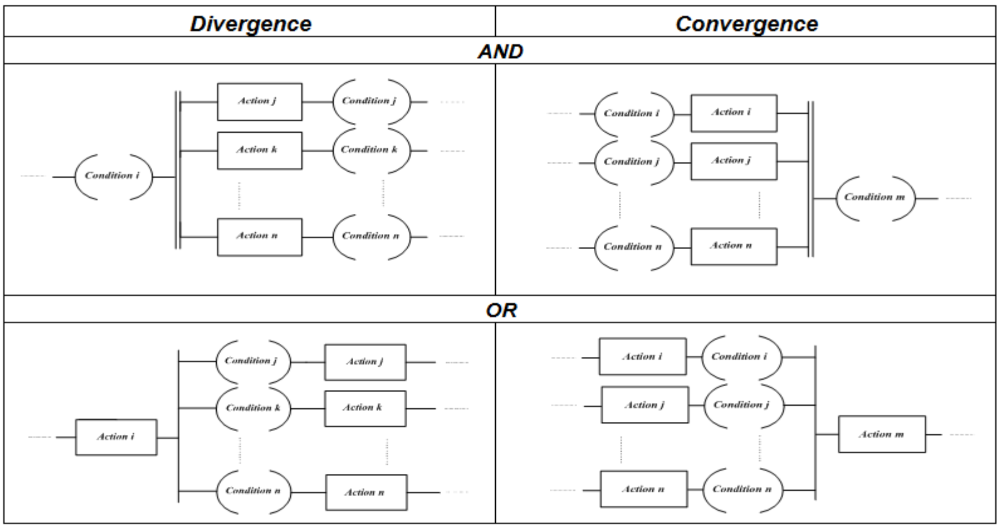
Table 11: ASML basic elements (Salmi A. , David, Summers, & Blanco, 2014)



4.4.2. Structures and architectures

ASML as a graphic language for assembly representation allows representing serial sequences, parallel sequences (AND divergence and convergence), and decision sequences (OR divergence and convergence) using specific symbols (Table 12).

Table 12: AND/OR divergences and convergences (Salmi A. , David, Summers, & Blanco, 2014)



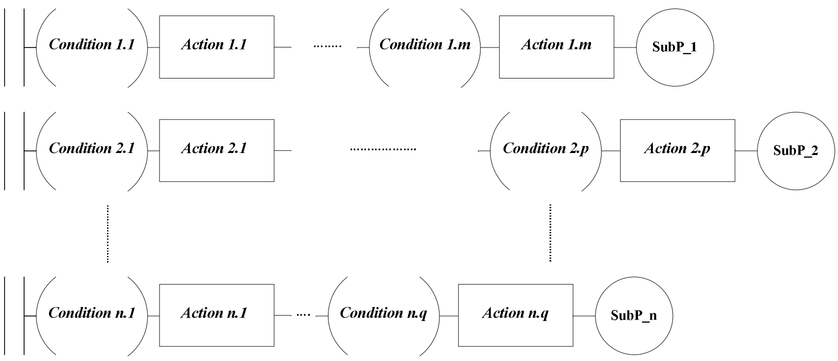
4.4.3. The referencing system

The ASML offers the possibility to represent assembly sub-sequences independently (Table 7). These sub-assembly references have to be written within the symbol circles as shown in Table 13.

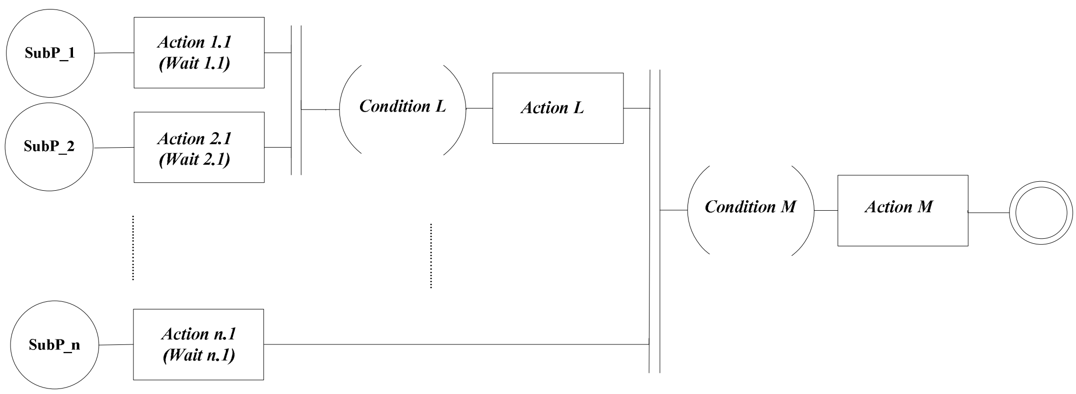
The references may be directly cited in other graph of assembly representation of the product everywhere the sub-product has to be used. The representation of the final product is possible using directly labels of sub-product in a graph leading to the final product represented by the double circle (Table 13).

Table 13: Referencing system in ASML language

References affectation to independent sub-assemblies



The final product assembly representation using sub-assemblies referencing



4.4.4. Resources representation with associated LoAs

A resource in ASML is defined by a block grouping tasks from 1 to n. A resource has a reference (as R1, R2, ... Rn) as shown Figure 3.

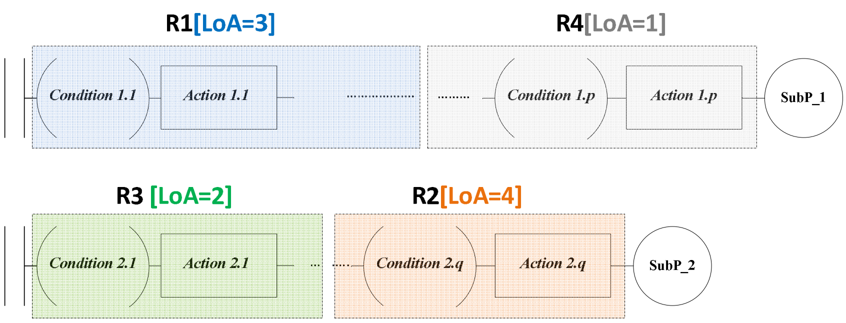


Figure 3: Resources representation in ASML

To each Resource Ri (i = 1..n) an LoA is associated. This LoA can have in our LoA approach one of the following values: Manual (LoA=1), Manual with automated assistance (LoA=2), Automatic (LoA=3), or Robotic (LoA=4). The LoAs can be shown in the representation as represented in Figure 3. Colors can be used according to the LoA. In our approach we use grey for manual, green for manual with automated assistance, blue for automatic, and orange for robotic (Figure 3).

The assignment of resources to all tasks of the graph with associated LoAs consists in one automation alternative. It also called automation scenario or configuration. For more rules about resources assignment, conflicts representation, and other aspects, interested readers are invited to find the detailed proposal (Salmi A. , David, Summers, & Blanco, 2014).

4.4.5. Time scale and actions scheduling

ASML representation uses a time scale and can include lags. These lags can be imposed for a given option of resources allocation. If a unique resource is executing the different tasks, it is evident that lags should be generated and represented for different sub-assemblies because this resource is not able to execute them in parallel (Figure 4 for example). A lag can be also caused by resource conflicts and collision avoidance management (when workspace is shared for example). It can also be imposed by the nature of the tasks or a technical choice imposing waiting the end of a task before executing a next one even if resources allocated to the different tasks are different. A lag representation can be mandatory and imposed since the generic ASML model. It can occur in other cases, dependent on the scenario of resources allocation generated. A mandatory lag is observable in the representation by an arrow between the end of the previous tasks and the beginning of its following. While in a generated lag (initially not mandatory), no arrow is used. From time scheduling, the two representations are similar and shown by the lag following a Gantt schedule.

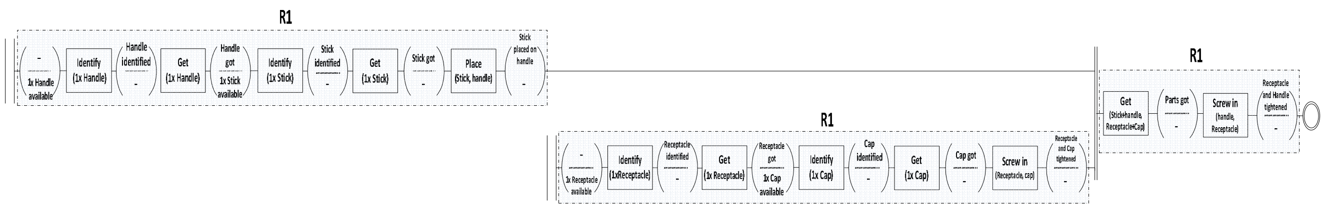


Figure 4: Lags representation in ASML

4.4.6. A standardized vocabulary of motions from the literature

As announced in the requirement 7 of section 4.2, we need a standard representation, ideally unique. This representation, when standard, will ease the assembly sequence analysis. A standard representation will make the representation process more objective. Moreover, it should become more efficient, reducing the workload of the user. It will support his thinking, interpretation and reasoning especially when decomposing the assembly motions necessary to show the sequence.

Therefore, we propose associating to the ASML language a standardized vocabulary to fill the assembly actions modelled as blocs as shown in Table 11. This is a fairly new point compared to the languages we reviewed in section 4.3. This position makes our language specific to its application context but it also improves its efficiency for our aims.

Building a standard vocabulary brings the advantage of a better and easier capacity to be computerized. This will allow for example, automatic generation of assembly sequences or setting up analysis algorithms for time estimation or assembly pattern recognition. The use of a standard vocabulary of assembly actions makes possible to estimate the assembly sequence time after developing time standards for the standardized assembly tasks. The assembly sequence time estimation is one of the important modules in the assembly cost calculation within the LoA optimization and decision support tool we develop.

We associate then to ASML a standardized vocabulary of elementary assembly motions from (Miller, Griese, Peterson, Summers, & Mocko, 2012)(Renu, Mocko, & Koneru, 2013). The vocabulary is associated to ASML to model processes. The previously shown Figure 4 uses this controlled vocabulary of motions and can be considered as an example.

After multiple analyses of this vocabulary, we think it can be extended by adding some additional motions to be more exhaustive. We propose then an extension in (Salmi A. , David, Blanco, & Summers, 2016). Table 14 shows the complete list containing the original vocabulary with some supplementary motions we propose to cover more operations. These extra motions are highlighted in bold in Table 14. The grey colored cells in the table 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 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 14: The standardized vocabulary of assembly motions

1	Align	7	Exchange	13	Inspect	19	Press	25	Scan
2	Apply	8	Get	14	Lay	20	Read	26	**Solder**
3	Clean	9	Handstart	15	Move	21	Remove	27	Tighten
4	Connect	10	**Hit[Hammer]**	16	Open	22	Restock	28	Unscrew
5	Disengage	11	**Identify**	17	Operate	23	Restrict	29	**Wait**
6	Engage	12	Insert	18	Place	24	**Rivet**	30	Walk

4.4.7. Automation alternatives definition using the motions vocabulary

In previous sections we defined the basic elements, principles, and vocabulary of ASML. These elements, with more details and rules defined in (Salmi A. , David, Summers, & Blanco, 2014); allow representing assembly processes and automation alternatives. The idea is, as proposed in LoA decision approach of chapter 3, to define an initial ASML model. Then, based on this generic model, the purpose is to be able to generate, ideally automatically, multiple alternatives of assembly systems. We present this process in the 2 next sub-sections.

4.4.7.1. The initial model representation

A first ASML model of operations, using the previously presented vocabulary, without resources can be built. The representation should be generic, otherwise, should not depend on the selected LoAs. This model should allow representing motions, sequencing the different independent sub-sequences, providing the structures or architectures of operations (serial, parallel, or decision), and showing the conditions that have to be fulfilled through the assembly including parts and tools that have to be available. An example is shown in Figure 5.

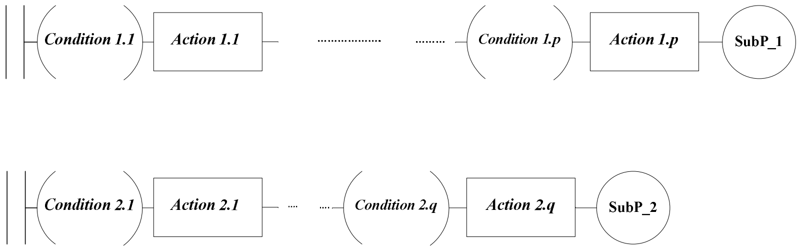


Figure 5: An example of an initial ASML model

4.4.7.2. Systems alternatives generation

Based on this generic standard ASML scheme, multiple scenarios should be proposed with resources allocation with various LoAs. Lags should be represented when required in the given automation alternative as previously mentioned. Figure 6 shows a more complex alternative with 3 resources where a lag is needed for R1 assigned to tasks belonging to 2 sub-assemblies.

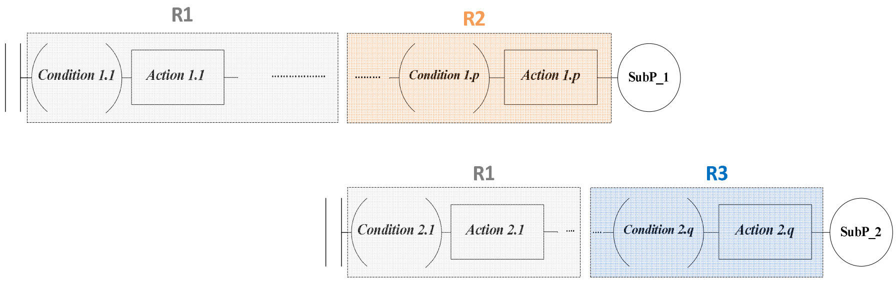


Figure 6: An example of an assembly system alternative

Each scenario with resources and lags can then be evaluated with regard to performance indicators for the sake of scenario assessment and LoA optimization. The optimization aims at finding the best system alternative with regard to different evaluating criteria that can be considered. Yet, a right system definition should follow a right dimensioning of resources with regard to the required productivity. To do so, time estimation should be performed so that resources can be assigned such as they can satisfy the required cadence according to obtained time estimations. This will be subject of chapter 5.

4.4.8. Discussion: the need to a high layer vocabulary

One of the important inputs for assembly system design is the product to be assembled design and parts features. Using 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 4.4.6 can be of interest for the automation descriptions, analyses, and LoA criteria involving (e.g. motions analyses with regard to ergonomics, repetitive motions as good signs for possible automation, quality, etc..), the too detailed granularity level of the motions vocabulary does not seem to be the most suitable for the step of initial generic model building and automatic generation of scenarios for multiple reasons that are for us signs to a need to a higher layer vocabulary. We proceed first by identifying the different gaps generated by the use of the motions vocabulary. Then, the solution we propose to address the gaps will be studied in next section.

The identified gaps concern the following 5 points:

- **The genericity of the initial assembly process model**

As previously described, the first representation of the assembly process is performed using a standardized vocabulary, the one shown in Table 14. To represent an assembly process based on product design using this kind of motions can be a long work with a high risk of error. Also, the 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). As example, we can consider the description in motions of the screwing task of two parts (P1) and (P2) by a resource R shown in Table 16. 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.

- **Dependencies between motions from resource point of view**

Another issue is to be able to assign, in the initial model, separated resources (e.g. workstations) to the different process activities. As explained, resources can be assigned to assembly motions in the initial model using the possibility that offers the ASML language. Yet, the way to systematically generate alternatives of resources assignment to motions cannot be easily performed. We realize that dependencies may 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.

- **Compatibilities between motions**

To solve the motions dependencies issue, 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 14 motions).

- **Process productivity analysis**

To conduct the LoA decision, every motion of the assembly process can be time estimated regarding different possible LoAs and the parts design features (chapter 5). 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 a required productivity target will also be different. The motions to be assigned to a resource will depend then on the selected LoA. This increases the problem resolution complexity for a systematic generation of alternatives. Addressing these difficulties using a low layer granularity seems to be unmanageable.

- **The combinatory explosion of alternatives generation**

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 is obvious that the total number of assembly elementary motions (Table 14) used to model the initial process for a complete real product assembly can be huge. As consequences, browsing all the possible alternatives of resources assignments to these elementary motions with all feasible automation options can be unfeasible because of the combinatory explosion of the problem. The low layer vocabulary of assembly motions is interesting regarding its offered details, associated time estimates (chapter 5) 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.

4.5. A high layer vocabulary 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 of 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.

4.5.1. The high layer vocabulary definition

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 Methods Time Measurement (MTM) (Maynard & Stegemerten, 1948), Design For Assembly (DFA) (Boothroyd, Dewhurst, & Knight, 2011), and the used one of motions (Table 14).

The vocabulary of assembly tasks is defined to include all possible 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 15. This preliminary list can be refined or extended. The focus of this proposal is to show that this vocabulary will help making automation decision more organized, more standardized, and more rigorously applicable within the LoA decision approach. The vocabulary is tailored to assembly systems modelling and automation deciding issues.

Table 15: 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

Using this vocabulary, ASML rules, and the approach of representing initial processes and then automation alternatives, the previously mentioned features for motions are still applicable to tasks. We only recommend to model tasks in lozenges instead of rectangles to differentiate them from motions. The representation at this granularity includes all defined rules, only conditions representation can be neglected. We show in Figure 7 an example.

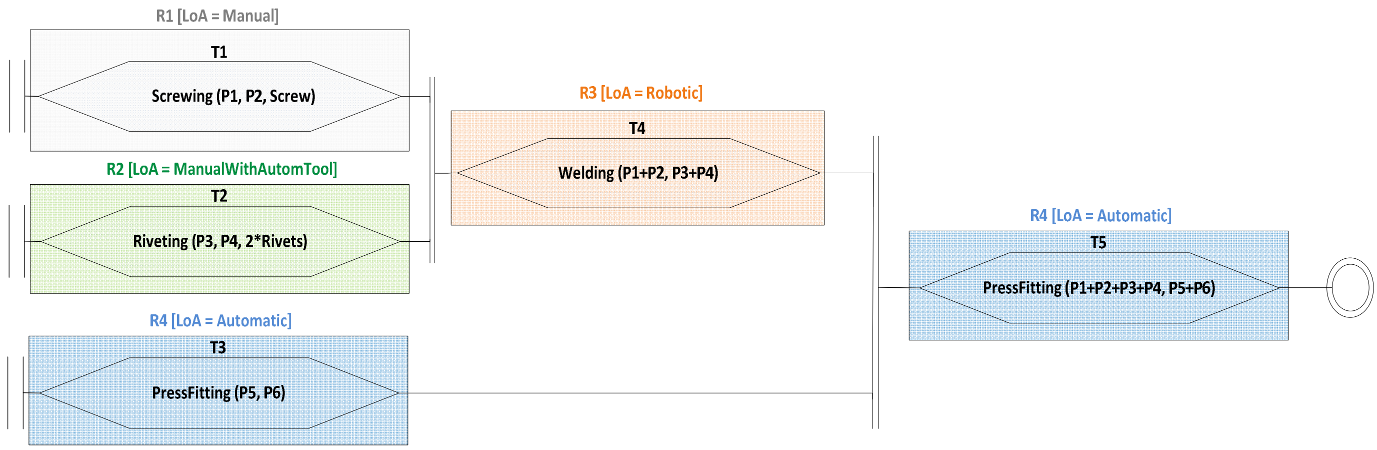


Figure 7: An example of using tasks vocabulary to represent an automation alternative

To allow motions analyses, the vocabulary of tasks is linked to the previously presented low layer vocabulary as presented in next section 4.5.2.

4.5.2. The connection of proposed vocabulary to ASML and motions vocabulary

As previously underlined, an important point to better 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 (Table 15) can match with the level of elementary motions (Table 14). In fact, every task can be defined by a kinematic decomposition as a succession of elementary motions for the different LoA levels. This allows the possibility to convert the representation to a more detailed one when it is needed (e.g. ergonomics analysis, time estimation). Depending on the task and associated resource and LoA, the required motions may be different.

Moreover, for some LoAs, the task can be impossible to be performed. For example, if a task necessitates an automated tool to be performed (such as for welding), 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 LoA, 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 (types and dimensions) can exist, or different kinds of riveting tools or associated energies 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 Table 16 representing decompositions of the "Screwing" task to the required motions for a 2 parts assembly "P1" and "P2". We define every part Pi as an object with attributes consisting in parts features impacting the time estimates for corresponding motion and task, 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.

Table 16: The Screwing task decomposing to required elementary motions in a 4 levels automation scale

	Screwing (LoA, P1, P2, Screw)			
LoA	*LoA=1* *(Manual using only physical strength)*	*LoA=2* *(Manual assisted with automated tool)*	*LoA=3* *(Automatic dedicated machine)*	*LoA=4* *(Robotic: Industrial Robot)*
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)	Identify(P1) Identify(P2) 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, ScrewingTool)	Identify(P1) Identify(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)

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 4.6, the way to use this vocabulary in assembly modelling and automation decision is more concretely presented.

4.6. Overview: use instructions of ASML and vocabularies for LoA decision issue

In the proposed methodology, we propose using the vocabulary of tasks to represent the initial generic representation of the process. This is detailed in section 4.6.1. Then, the alternatives can be generated as explained in section 4.6.2 thanks to the link tasks to motions that we previously proposed in section 4.5.2.

4.6.1. A generic initial assembly model of tasks representation

The first graphic representation of the process is defined in tasks using ASML and the proposed tasks vocabulary instead of motions. 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 multiple initial sequences may exist (Homem de Mello & Sanderson, 1991), including generic ones, it is recommended, for automation technical feasibility and to support selecting a feasible and satisfactory sequence, to schedule as much as possible similar tasks in succession to make the possibility of tasks grouping by a same resource easier when optimizing alternatives. 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. We propose figuring the assembly techniques (tasks of Table 15) 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).

4.6.2. Systems alternatives generation with conversions of tasks to motions

Based on the tasks-based ASML model, alternatives of assembly systems can be generated. An alternative results in resources allocation to tasks with LoA selection. The assignments have to be performed through following steps:

4.6.2.1. LoA selection to tasks and motions deduction

Thanks to the possibility of decomposing tasks to associated motions once an LoA is selected, it is possible to convert a tasks-based model to a lower layer motions-based model. Motions representation are useful to detect phenomena that can be good signs for 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). Particularly, tasks conversion to motions provides a better possibility of time estimation and allows correct resources allocation to satisfy the required cadence as it will be explained in next section. Assuming every elementary motion is time estimated in the selected corresponding task LoA, tasks time estimates can be obtained. Once all the model's tasks are time estimated, resources can be appropriately assigned to satisfy the required productivity. This will be studied in more details in chapter 5 on time estimation.

4.6.2.2. Resources assignment to tasks

Once the tasks time estimates are available in the different LoAs thanks to the conversion to motions, resources assigning can be performed. To easier the procedure, it is recommended to perform the assignment in the high layer model of tasks. The resources assignment should be established 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, the assignment has to be performed with consideration of each resource independently, with consideration of the selected LoAs to tasks. This will be more detailed in next chapter 5 of the thesis.

It can be worthwhile to remind that assignment of resources to tasks, in addition to time estimation and productivity satisfying, should take into account the selected decision criteria detailed in chapter 2 – section 2.4. The consideration of criteria can impose or restrict the set of possible automation scenarios to be generated. Some tasks should or not be performed in some automation levels according to given criteria. For example, if the model contains a 'welding' task, this task can be prohibited, according to discussions with the manufacturer, in LoAs 1, and/or 2 (respectively manual, and manual with automated assistance). In this case, only automatic or robotic LoAs can be associated to this task for all alternatives or scenarios that can be generated. This will be more explained in the example of section 4.8. The consideration of criteria and automation possibilities will be more formalized for computerization in the optimization model in chapter 7.

Once a scenario is available in which all tasks have been assigned to resources, further analyses can be handled with regard to performance indicators, such as time analyses, productivity indicators, and cost estimation. Time estimation will be studied in next chapter 5 while cost estimation will be studied in chapter 6.

4.7. Enabled developments by proposed ASML and associated vocabularies

We present in this section developments already proposed in the research area and others that can be performed as possibilities powered by proposed modelling language (ASML) and its associated vocabularies. In section 4.7.1 we present a development already proposed. In section 4.7.2, we propose possible future research developments.

4.7.1. An integration of literature methods with ASML

We proposed in (Salmi, Dhulia, Summers, David, & Blanco, 2015) an enhancement of LoA methods to help the issue of analysis and improvements possibilities proposals for existing assembly systems. The method take benefit of modelling possibilities of ASML language and integrates it to DYNAMO++ method ((Fasth & Stahre, 2008); method M6 of Table 8 – chapter 3) and the decision Table method of Boothroyd and Dewhurst ((Boothroyd & Dewhurst, 1983); method M8 of Table 8 – chapter 3). The method is justified in (Salmi, Dhulia, Summers, David, & Blanco, 2015). We show the obtained proposed method in Table 17 where new proposed features allowing the methods integration are highlighted with orange color, particularly ASML modelling step highlighted with dark orange, while used existing methods in the integrated approach (M6 and M8) are green colored.

We show then this integrated guideline method (Table 17) as a development facilitated by proposed ASML and powered by the ease of generating alternatives and of describing automation possibilities using ASML graphic language. In the case of that method, the initial generic process is a model of tasks to be defined as the model of the current assembly system which exists in the company. Then, based on this model of tasks, automation alternatives should be generated and analyzed to search for the optimal configuration. If the optimal one converges to the current process, the actual automation level can be kept. Else, the current assembly system should be changed and optimized. In this case the current process should be updated to follow the optimal configuration found. This method is tailored to support the question about improving existing processes rather than designing new assembly systems as it is the main target of this thesis.

Table 17: An integration of methods with ASML to help existing processes automation improvement (Salmi, Dhulia, Summers, David, & Blanco, 2015)

Steps	Description	Provenance	Phase
Step 1	Identify the system to improve onsite	*Step 1 Dynamo++*	*Pre-study*
Step 2	Walk the process	*Step 2 Dynamo++*	
Step 3	Identify flow and time parameters by Value Stream Mapping (VSM) building	*Step 3 Dynamo++*	
Step 4	Identify the main operations and subtasks for selected area by Hierarchical Tasks Analysis (HTA) designing	*Step 4 Dynamo++*	*Measurement*
Step 5	Measure LoA using the LoA mechanical and information scales	*Step 5 Dynamo++*	
Step 6	Results documentation	*Step 6 Dynamo++*	
Step 7	Process ASML modelling with resources corresponding to different workstations identified	*ASML modelling*	*Process modelling*
Step 8	Apply B&D to the different workstations one by one independently	*B&D*	*Prel. Solution*
Step 9	Decide min and max LoA for the different tasks by Workshop considering B&D as preliminary solutions	*Step 7 Dynamo++*	*Analysis*
Step 10	Design Square of Possible Improvements (SoPI) based on workshop results	*Step 8 Dynamo++*	
Step 11	SoPI analysis	*Step 9 Dynamo++*	
Step 12	Write / visualize the suggestions of improvements	*Step 10 Dynamo++*	
Step 13	Try other reorganizations/ reconfigurations of the workstations by other resources allocations in the ASML model if other feasible alternatives exist, else Go to step 15	*New*	*Other reconfigurations and alternatives*
Step 14	Loop: Go to step 8	*New*	
Step 15	Discuss the different alternatives and SoPIs (workshop with experts) and keep the best	*New*	*Discussion*
Step 16	Implementation of the decision suggestions	*Step 11 Dynamo++*	*Implementation*
Step 17	Follow-up when the suggestions have been implemented and analyses their effects on time and flow	*Step 12 Dynamo++*	

4.7.2. Possible other developments using ASML and vocabularies

In this section we propose other possible applications as developments that can be of interest using defined ASML language and its associated vocabularies are as follows:

- **Assembly workers' instructions generation**

Modelling the process using tasks vocabularies and associating LoAs can be organized, once each task can be converted to corresponding motions according to selected LoAs, in work instructions that can be helpful to assist workers during assembly. The description of assembly operations in motions or tasks can be useful to explain to workers, especially novices, how to successfully perform operations with their description and sequencing.

- **Automatisms dimensioning and sensors selection**

The representation of the process using motions can help in dimensioning and designing machines by identifying sensors and actuators. For example, to each 'identify' motion, should correspond one or multiple sensors. The architecture provided by ASML model can also help on identifying antecedents and dependents of the automatism agents thank to the AND and OR divergences and convergences that can be modelled in ASML models. The conditions in the ASML model can also help identifying sensors or detectors, such as conditions of next task or motions to be performed (condition C_y of Table 11) of type 'Part got', 'Part available', 'Parts' press fitting completed, etc.

- **Robotic systems programming**

Similarly, the representation of the process with detailed motions can help on designing kinematics of future robotic systems and help on defining the program with the robot motions and implementing it in the initial robot program. This can be helpful for robot suppliers companies in designing initial drivers programs of robots according to their customers' requirements specifications described as a sequencing of motions that can be consequently described with the vocabulary of motions to be used with ASML representation.

These different developments, basically related to other issues than the one of this thesis, can be of interest to be developed as future researches. In this thesis, we focus on the issue of supporting automation decision for new assembly systems design.

4.8. A validation example: assembly modelling and LoA criteria consideration

The aim of this section is to validate the proposals around modelling and the use of standardized vocabularies on a simple example. We show the product to be assembled in Figure 8 where Figure 8.a shows the different parts to be assembled (4 parts: 'Cap', 'Stick', 'Receptacle', and 'Handle') and Figure 8.b shows an AND/OR graph of how these parts should be assembled as well as the corresponding tasks to be performed.

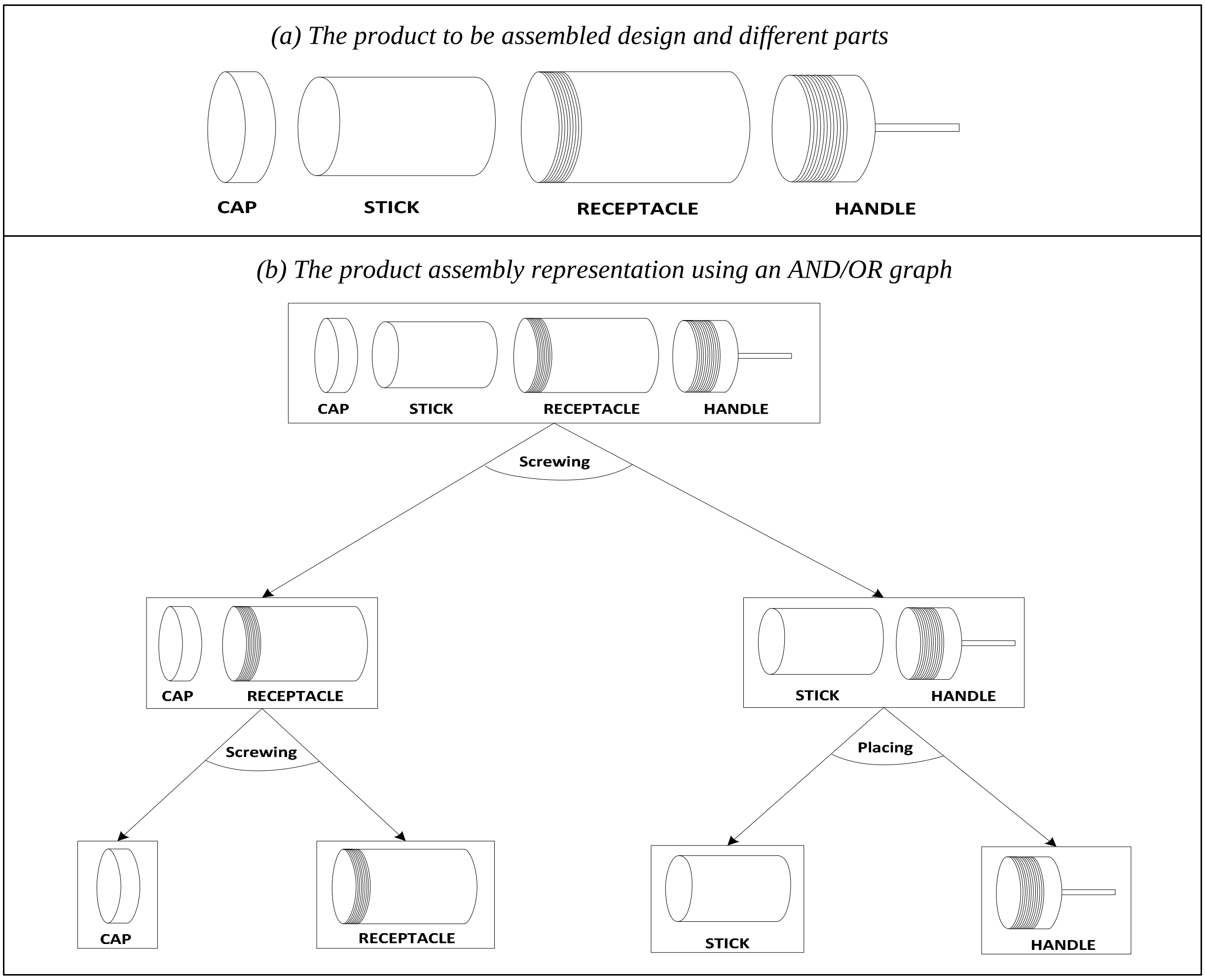


Figure 8: An example of an assembly system alternative

4.8.1. Step 1: Product design analysis

The first step consists in analyzing the design and identifying how the parts have to be assembled. In the example, Cap should be screwed in Receptacle to give the first sub-assembly; Stick should be placed inside Handle to give the second sub-assembly; and finally, the 2 sub-assemblies should be grouped by screwing Handle in Receptacle to give the final assembly. Once the assemblies identified and well understood, the initial generic ASML model can be built as a 2nd step as detailed in next section 4.8.2.

4.8.2. Step 2: The generic ASML model

Based on the given design and assembly sequence shown in the AND/OR graph (Figure 8 - b), the initial model of the assembly process can be built. The representation should follow ASML rules and use the vocabulary of assembly tasks (Table 15). For the case of this example, the obtained generic model of tasks is shown in Figure 9.

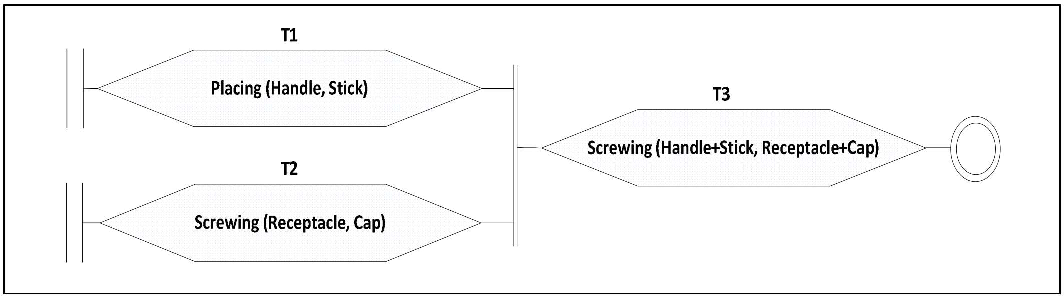


Figure 9: The initial assembly tasks generic representation for the given example

As shown in Figure 9, to each assembly operation correspond a tasks number (from T1 to T3). Then, the purpose is to generate automation alternatives based on this generic representation of tasks and search for the best or best alternative. Yet, before that, analyses of the obtained tasks should be performed with regard to the LoA decision criteria to limit the set of possible solution and to lead to satisfactory and/or feasible solutions. This is tackled in next step described in section 4.8.3.

4.8.3. Step 3: Tasks analyses with regard to LoA criteria

In chapter 2, LoA decision criteria to be involved in automation decision were identified. The complete list of criteria was shown in Table 7 (chapter 2). As previously explained in chapter 2 – section 2.4, to simplify the study that may involve up to 73 criteria for the initial list of criteria candidates, the decider can start first by identifying the most impacting ones. At a second stage he has to reconsider the only selected ones in the analysis and iterate them on his model's tasks.

For this example, let's assume that the significant criteria selected by the decider are:

- C1: Flexibility: multiple variants should be handled on a same line
- C2: Workers conditions: ergonomics favored as much as possible
- C3: Productivity: a high volume is required

If we consider these criteria, C1 should forbid automatic (LoA=3) for all the tasks because it is supposed to be inflexible (dedicated automatic machines by definition). Task T2 and T3 (direct parts screwing) are not ergonomic (criterion C2) if performed completely manual by hands, especially for repeatability (high volume – criterion C3). This should forbid LoA = 1 for T2 and T3. Finally, as a high productivity is required (criterion C3), we assume that every resource is assigned to no more than one task. The exact dimensioning will be more accurately studied in next chapter 5 on time estimation.

To conclude, for the different tasks, the new sets of solutions for tasks are as follows:

- T1 = {1, 2, 4}
- T2 = {2, 4}
- T3 = {2, 4}

As a result, the remaining number of possible LoA combinations is of 3*2*2 = 12 instead of the initial total of 64, so more than 5 times less.

4.8.4. Step 4: Automation scenario generation

The aim of this step is to generate the possible automation alternatives that have to be evaluated with regard the performance indicators in next step 5.

As one alternative, we consider a manual resource (LoA=1) for T1, a manual with an automated tool Automated Assistance (LoA=2) for T2, and robotic (LoA = 4) for T3. The obtained graph of the mentioned automation configuration is shown in Figure 10.

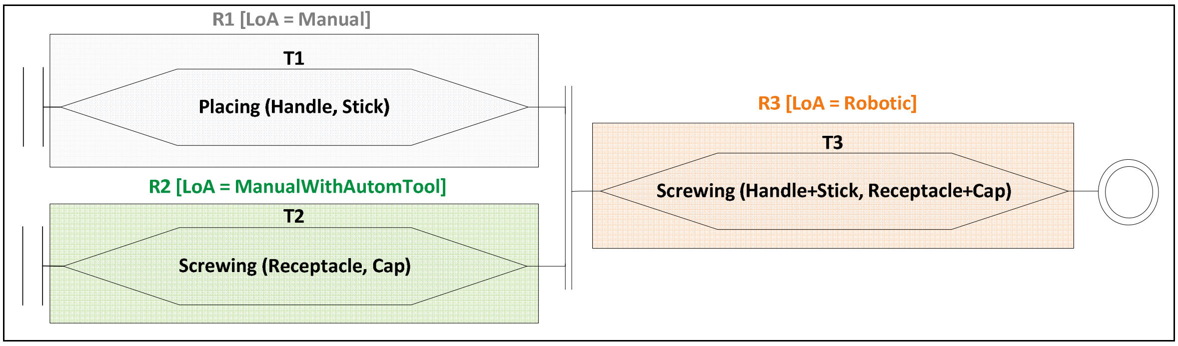


Figure 10: An automation alternative representation for the given example

4.8.5. Step 5: Conversion of tasks to motions and analyses

In this step, every task is converted to motions using the conversion mechanism shown in section 4.5.2. The example of screwing was shown in Table 16 and is still useful for this example. The conversion to motions aims at facilitating analyses of motions and detecting features that can be favorable or not to automation. Moreover, particularly the conversion to motions will make possible time estimations and then cost estimation because every elementary motion can be time estimated. Time and cost estimation will be studied in next 2 chapters 5 and 6.

The conversion of the automation scenario of Figure 10 is shown in Figure 11.

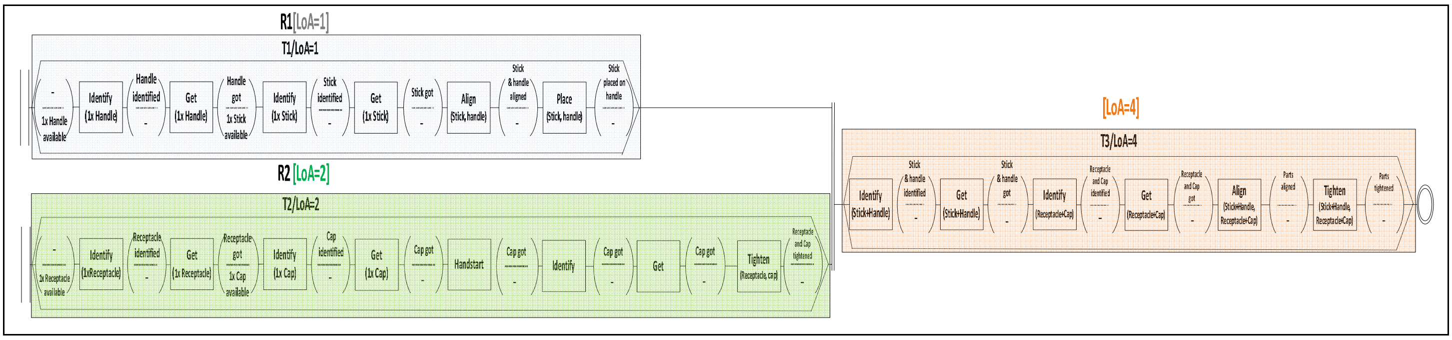


Figure 11: The decomposition of the previously considered automation configuration to motions

As it can be observed in Figure 11, task T1 is decomposed to 6 elementary motions, T2 to 8, and T3 to 6. In the case of using such motions representation instead of the tasks representation of Figure 10, this gives, with consideration of eliminated scenarios by LoA criteria (step 3 – section 4.8.3), a total of: $3^6 * 2^8 * 2^6$ = 11 943 936 combinations of LoAs compared to 12 using the tasks representation for the initial scenario. The generation of alternatives using the tasks approach allows then to avoid 11 943 924 infeasible and meaningless extra scenarios. The granularity level of tasks is then clearly more appropriate to generate automation alternatives. In addition, the motions representation, such as the one of Figure 10, cannot be generic and corresponds only to selected LoAs. By contrast, tasks model is generic and standard.

4.9. Conclusion

In this chapter, a new assembly modeling language named ASML is defined. ASML is built to offer an appropriate language to represent assembly sequences and automation alternatives for the issue of automation decision in assembly. This representation language is proposed after a formalization of the automation decision specification into requirements (section 4.2) that a modelling language should fulfill to be appropriate to LoA decision issue. The proposal is also the consequence of a literature review and analysis (section 4.3) giving important insights for its definition.

ASML exploits multiple principles of the literature methods and integrate them into a same language in order to fulfill the entire listed requirements. The result led to a graphical sequential representation method (requirement 1.1) in which sub-sequences are modeled to be executed in serial or parallel is possible (requirement 2.1), representing the required eventual conditions and precedence constraints in the level of transitions between operations (requirement 2.3), with a scheduling and corresponding resources allocation (requirement 2.2). ASML offers a visibility of the final product assembly (requirement 2.4) using its compact tasks representation, but also its referencing principle, offers an easy sharable representations (requirement 3.1) in collaborative engineering discussions, decision making and information exchange. It also provides a mean to generate easily different scenarios (requirement 3.2) of assembly systems concepts. Thanks to the associated proposed vocabulary of tasks, the initial representation and generated alternatives are standard (requirement 2.5) for a given assembly sequence in input.

ASML is associated to a standardized vocabulary of assembly elementary motions. This vocabulary even has multiple advantages (analytic description, signs to automation detection such as repetitive or non-ergonomic motions, facilitates time estimation, etc). But, it has limitations as explained in section 4.4.8. The limitations are related to a low granularity level imposing genericity issues, difficulties of assigning resources because of dependencies between motions from resource execution and motions compatibilities point of views, analyses of productivity for different LoAs for completely different and non-generic motions models, and important combinatory explosion. These limitations highlight a need to a higher layer vocabulary for alternatives generation. A high layer vocabulary of tasks is then defined and associated to the low layer one to bring benefit from both of vocabularies.

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 modelling language ASML and the use of controlled vocabularies. The lists of vocabularies of tasks and motions can be extended.

The consideration of tasks allows an easy way to consider identified LoA criteria. A guideline to consider the criteria is presented. The decider has to model the process using an ASML model of tasks, select his prior criteria, then to evaluate his model's tasks with regard to the different selected criteria in the different LoAs according to his LoA scale.

To accompany the assembly system design, the surrounding process we tailor is based on rapid creation of relevant indicators (rapid estimation of time, cost, resource enrollment etc). Therefore, different configurations shown in ASML representations should be tested and evaluated from LoAs, architectures, time, and cost point of views.

The whole approach with graphic modelling and standardized vocabularies may allow several other direct developments and useful applications. In this chapter, we proposed worker's instructions generation, automatisms dimensioning with sensors selection, and robotic systems programming.

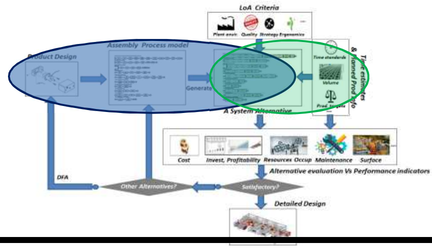
The approach can be also transposed to completely other applications, such as for manufacturing or logistics, by development of vocabularies of tasks and motions to be combined to ASML with evaluation criteria defining. Applications to Lean manufacturing by focus on core or value-added motions, on non-value-added motions, or on their time estimates, can represent one of interesting future orientations.

Some limitations of this approach can be identified. The way of considering criteria in our approach can be considered as limited. We consider criteria as favorable or not with regard to tasks and LoAs. This can be enhanced by consideration of criteria levels or scales, such as: 'not suitable', 'possible but not recommended', 'medium', 'good', and 'excellent'. Then, a score can be associated to each automation alternative using a mean value of the different criteria scores or a formula with ponderations on the criteria, such as high ponderation for most significant ones, and lower for the others. This can help in the optimization issue.

Another limitation consists in the dependency of the approach to the given assembly sequence in input, that we model in AND/OR graph. In fact, multiple initial ASML can exist because of the existence of multiple AND/OR graphs and sequences for a same product assembly (Homem de Mello & Sanderson, 1991) (Homem De Morello & Sanderson, 1990). To make the approach exhaustive, the proposal should consider the generation of all possible assembly sequences, ideally directly from the product design CAD tool, and to apply the methodology to each of them. At the end, the global optimum should be kept. In the context of this study, we consider that the initial sequence is imposed.

We present in this chapter how to represent using ASML automation alternatives with resources assignment and consideration of corresponding LoA to each resource. It is mentioned that time estimation is needed to perform correctly this assignment that must meet the required productivity. This requires rules and databases of time estimation to predict automation alternative time performances and adjust them to the required cadence. This consists in the purpose of next chapter 5 of the thesis.

Chapter 5. Automation Alternatives Time Estimation[3]



Abstract

To support appropriate resources assignment in a given assembly system automation alternative, a need to time estimation is highlighted in the previous chapters. Time estimation will allow assigning resources to tasks and motions so that the required productivity can be ensured. In this chapter, we propose an approach for assembly time estimation during the conceptual phase of new assembly systems design. The proposed approach includes ASML based time estimation rules and database of time standard values allowing time prediction with consideration of selectable automation levels. Related time applications such as processes time performances evaluation, productivity assessment, and assembly systems early phase balancing are proposed. Time estimation represents also a main basis to cost estimation that will be proposed in next chapter.

5.1. Introduction

A dedicated modelling language to automation decision issue was proposed in the previous chapter. This language labelled "ASML" was defined to address multiple requirements listed to reflect the problem exigencies. The language offers multiple advantages that can efficiently help the decision making. Important ones are: standardized graphical representations of the assembly process, automation alternatives with resources representation and their LoAs, conflicts and constraints representation, various architectures possibilities representations with serial, parallel (AND), and choice (OR) sequences and scheduling representation of assembly operations (motions and task) and resources thanks to the time scale that it uses. The language also uses standardized vocabularies: a low layer vocabulary of motions and a higher layer one of tasks. Each of the 2 layers has advantages. The migration from a layer to another is possible to bring benefit from both of the layers. These developments aggregating graphic language and standardized vocabularies make possible process time estimation. Time estimation, as previously mentioned in chapter 4, is needed to appropriately assign resources. Every automation alternative, to be appropriate, should be designed so that it can reach, with an exact fit for purpose LoA, without being over or under productive, the required productivity expressed by the expected cadence. This explicitly requires right adjustment of resource assignment with regard to a given cadence or takt-time of the future production. Time estimation is also crucial for cost estimation that will be studied in chapter 6. The aim of this chapter is then to propose a method to time estimate processes represented in ASML language to address these needs of appropriate resources assignment and cost estimation.

[3] Time estimation is presented in: (Salmi A. , David, Blanco, & Summers, Assembly Modelling and Time estimating during the early phase of Assembly Systems Design, 2015)

We proceed then for this sake, first, by presenting in section 5.2 literature approaches in time estimation in similar issues. Our time estimation procedure is then presented in section 5.3. A database of time standards to be associated to ASML models is presented in section 5.4. Rules allowing the time estimation are defined in section 5.5. Related time-based developments and applications to appropriate resources assignment and assembly systems design helped by proposed time estimation approach are proposed in section 5.6. A validation example is proposed in section 5.7. The chapter is finally concluded in section 5.8.

5.2. Time estimation literature in similar issues

Assembly time estimation standardization has already been a research debating topic since 1948 with Methods Time Measurement (MTM) (Maynard & Stegemerten, 1948). Yet, this standardization only handles manual assembly. Boothroyd et al. (Boothroyd, Dewhurst, & Knight, 2011) present a database of time standards for handling and insertion motions for the sake of assembly time estimation. In another work dedicated to cost estimation, Swift and Booker (Swift & Booker, 2013) propose the use of a similar approach consisting in handling and fitting indexes to predict the process time to estimate its corresponding cost. The indexes are used to consider the product to assemble complexity. The indexes can be considered as a sort of adjustment of time estimates according to given product complexity. These indexes are related to manual assembly. Experimental values are provided with tables and figures (abacuses) to enable retrieving the appropriate values according to the case and operations complexity.

Most of LoA decision methodologies, previously reviewed in chapter 3, are considering time aspect during their decision process. In the LoA decision approach developed by Ross (Ross, 2002), time criterion is also considered through the decision process. In fact, different time drivers and parameters such as cycle time and the period of time available to start mass production are considered in the decision approach. Windmark et Al. (Windmark, Gabrielson, Andersson, & Stoehl, 2012) also consider input time parameters in their proposed LoA calculation such as cycle time, downtime, setup time, and batch production time. Gorlach et al (Gorlach & Wessel, 2008) consider within their cost-based LoA methodology several time parameters such as cycle time, shift duration, manufacturing times for different operations: direct workers, quality control workers, re-workers, auxiliary workers. Consequently, it can be realized and confirmed based on the literature that time estimation is almost supporting LoA decision. It is generally performed to estimate the assembly or manufacturing cost. Time estimation of processes alternatives appears to be obviously one of the most important bases in the topic of LoA deciding in assembly.

Among the mentioned literature works in time estimation in assembly and LoA decision, the ones that detail the time estimation procedure and provide practical standardized techniques to time predict, and that can be consequently of interest are: MTM method (Maynard & Stegemerten, 1948), Boothroyd et al method (Boothroyd, Dewhurst, & Knight, 2011), and (Swift & Booker, 2013).

The strengths of MTM method (Maynard & Stegemerten, 1948) consists in its standardization by a use of a controlled vocabulary of assembly motions and their time estimate considering involved factors in assembly. Unfortunately, the method is only dedicated to manual assembly. The vocabulary of motions is also quite limited to only 8 motions.

The approach of Boothroyd et al (Boothroyd, Dewhurst, & Knight, 2011) presents time estimation database values for manual, automatic, and robotic. The estimations also consider the assembly complexity impact on time estimation by parts features involvement in the time estimates values. Yet, it considers only handling and insertion operations. Moreover, the estimations do not consider multiple techniques that we detail in our proposed vocabularies, such as riveting, clipping, welding, or bolting.

The approach of (Swift & Booker, 2013) presents the same advantages but also the same weaknesses of previous Boothroyd et al. approach. Moreover, it only treats manual assembly.

Based on this review, we proceed by defining an integration of principles of methods that were mentioned as of interest. We define a more dedicated procedure to LoA issue which addresses the identified weaknesses. The proposed procedure is presented in section 5.3.

5.3. The time estimation procedure

We propose in this section a time estimation procedure based on an ASML modelled process and a similar approach to MTM, Boothroyd et al, and Swift and Booker for assembly time estimation. The aim is also to support complex processes with consideration of their possible architectures (serial, parallel, choice sequences). For time estimates, the proposal considers our exhaustive list of assembly vocabularies (chapter 4 – sections 4.4.6 and 4.5.1). As the reviewed methods provides only restricted databases of time estimates with no rules for estimation, and as industrial processes can be complex (architecture), we present time estimation rules for represented processes. As a consequence, our proposal includes a database of time estimates adapted to the exhaustive defined vocabularies with time estimated in all possible LoAs (section 5.4) and rules involving processes architectures (section 5.5). The proposed time estimation procedure is described in Figure 12. We define the procedure in details by the following steps:

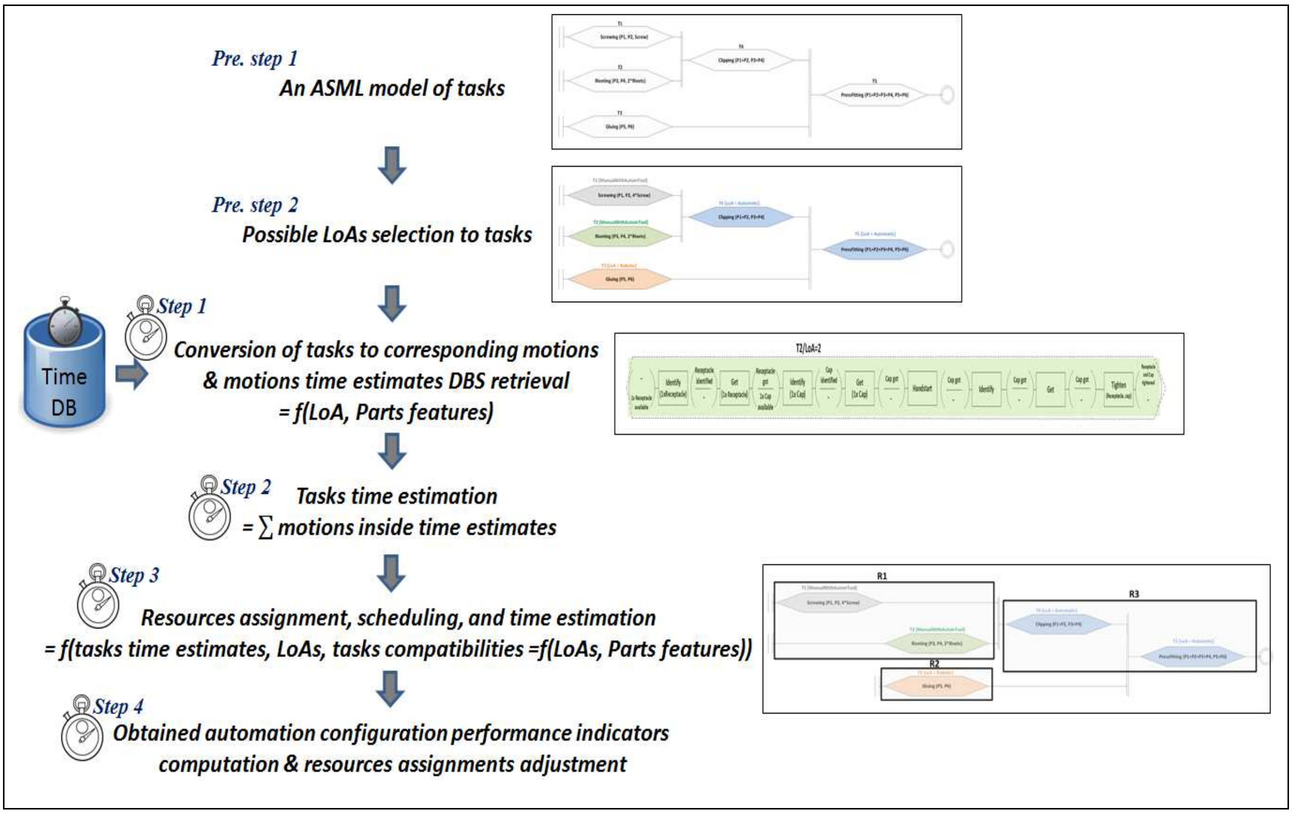


Figure 12: The decomposition of the previously considered automation configuration to motions

- **Preliminary Steps: Generic ASML tasks-based model and LoA selection to tasks**

First, the generic model of the process should be defined using the tasks vocabulary and ASML rules. This process was detailed in the previous chapter 4.

Based on the ASML generic model, an automation alternative/configuration can be defined first by LoA selection to tasks. Selected LoAs should consider the suitability by analysis of the tasks with regards to LoA criteria in the different LoAs (chapter 4 – section 4.8.3).

Once an automation alternative with tasks and their associated possible LoAs are available, time estimation procedure can begin. Time estimation is performed through 4 steps (Step 1 to 4)

- **Step 1: Conversion of tasks to motions and corresponding time estimates retrieval**

Once LoAs are selected to tasks, each task can be decomposed to its corresponding motions considering selected LoAs (chapter 4 - 4.5.2). As each motion can be time estimated in possible LoAs, thanks to a database of time estimates (section 5.4), each task can be then time estimated.

- **Step 2: Tasks time estimation**

Tasks can be time estimated as a sum of their composing motions. We use this deduction from motions time estimation databases rather than directly from tasks time estimation because it would be easier to estimate elementary motions then global tasks, also, because of the parts features number of combinations that will result. In that case, a database of all tasks should be built for all possible combinations of parts features. This would be much more time costly to establish, because of the much higher number of possible combinations. It would also present much repeatability as multiple same motions exist in different tasks (such as identify, get, or insert). The definition of motions time estimates will also be more favorable to genericity issues. It can handle possible extension by the possibility to time estimate new tasks, if exist, that can be not included in the tasks vocabulary list of Table 15. A new task that may be added to the list should be just decomposed to corresponding motions (as previously in Table 16 for screwing task). Then, motions time estimates can be automatically retrieved to deduce the new task time estimation. If a new motion is not already defined for that new task, the new time estimation will only concern this new motion. The other motions time estimates defined in the task decomposition will be retrieved from the existing database, for all possible LoAs, because there is no need to recreate and reproduce the estimates that are already available. So the process of database update will be faster than in the case of direct task time estimation. That is why we proceed in our approach to a conversion of tasks to motions to indirectly time estimate tasks.

- **Step 3: Resources assignment, tasks scheduling and time estimation**

Once all tasks are time estimated, resources can be appropriately assigned according to selected LoAs and compatibilities between tasks in given LoAs (chapter 6). The assignment of resources will allow resources scheduling and tasks scheduling inside resources according to the initial sequencing of tasks given by the generic initial ASML model. The assignment of resources to tasks should then follow, for each resource independently, the required productivity (Step 4).

- **Step 4: Productivity analysis and resources assignment adjustment**

The focus can be performed independently on the level of each resource because the process productivity is defined by the slowest resource of the process. If every resource satisfies the required cadence, the obtained configuration will be satisfactory from cadence point of view. This will be expressed and verified by the productivity performance indicators that have to be computed (section 5.6) to evaluate the coherence of the defined configuration to the required productivity. If the obtained productivity is not coherent with the required one, resources assignment should be adjusted to fit the targeted cadence.

5.4. A database of assembly motions time estimates

As presented in the time estimation approach, motions time estimation values should be retrieved (Figure 12; step 1 – section 5.3). This imposes a need to have a database of time values for the different motions that an ASML modelled process can need to be time estimated. Ideally, the database would be standard with time estimates for all elementary motions of the previously defined standardized vocabulary presented in chapter 4. Similarly to databases of manual motions (8 motions) of MTM (Maynard & Stegemerten, 1948), DFA (Boothroyd, Dewhurst, & Knight, 2011) with 'handling' and 'insertion' time estimates, and Swift and Booker (Swift & Booker, 2013) 'handling' and 'fitting' time estimates, we propose to build by analogy a database of time standards following this principle for our motions vocabulary of Table 14.

Accordingly to the mentioned approaches, to retrieve a motion's time value, these time standards should take into account the involved time impacting criteria corresponding to the given motion, but also the selected LoA. We establish then a database table of motions time standards.

The table was built following values from the mentioned literature approaches and some industrial time-keepings files. Industrial robots datasheets and information such as their speed and distance during assembly to deduce the corresponding time were also used. For some motions or LoAs, approximations were used based on determined values for similar motions, such as manual with automated assistance LoA motions are in some cases computed as mean average of manual and automatic values of corresponding motion and criteria.

The purpose of defining this database is to have default values to support time estimation during the LoA decision process. Defining a generalized, exhaustive, or extremely accurate database does not consist in the first goal of this research. In fact, such work can be much time consuming and requires several time experiments and timekeeping in different fields with plenty industrial partnerships. We provide in this work the time estimation reasoning, with the database architecture, and some values example. The finality is to show how such reasoning can handle modelling, objective methodologies, and estimations for the sake of automation decision. Enhancing the database time values or extending it by consideration of other time impacting criteria, or possibly new motions, can be proposed as a research opening. As another use, the manufacturer can associate his values and customize the database according to his historical data, eventually for restricted possibilities of automation levels as alternatives that he wants to compare. Software charged to retrieve time value should be flexible to automatically handle database values changing or extensions with no necessity to make effort to modify the programs.

According to the used standardized vocabulary of motions (Table 14 – chapter 3), a total of 30 motions exist. Correspondingly, the exhaustive time estimation database should include 30 motions. We show an extract in Table 18. The time values retrieval should be done according to:

- The given motion (1st digit): consists in the motion's code in ['1'..'30'] for the 30 motions.
- The different possible LoAs for the given motion (2nd digit). In our case we use a 4 levels scale (Table 4 – chapter 1). Consequently, the second digit can have one of 4 possible values: 'M' for Manual, 'MT' for Manual with automated Tool or automated assistance, 'A' for Automatic (LoA=3), or 'R' for Robotic.
- The time impacting criteria corresponding to the given motion (3rd digit): the time impacting criteria consideration corresponding to the complexity associated to the motion to be performed. Codes are associated to each criterion-value as letters ('a', 'b', 'c', etc). They should be determined as sub-rows once digit 1 (motion) and 2 (LoA) are selected.

Table 18: Time estimation database [sec] (extract)

		MANUAL				MANUAL WITH AUTOMATED ASSISTANCE/TOOL				AUTOMATIC				ROBOTIC			
		M				MT				A				R			
1	**Align[1]**	Easy to handle	Symmetric	*a*	1.717	Easy to handle	Symmetric	*a*	1.202	Easy to handle	Symmetric	*a*	0.687	Easy to handle	Symmetric	*a*	0.858
			½ sym	*b*	1.978		½ sym	*b*	1.384		½ sym	*b*	0.791		½ sym	*b*	0.989
			Non Sym	*c*	1.911		Non Sym	*c*	1.337		Non Sym	*c*	0.764		Non Sym	*c*	0.955
		Difficult to handle	Symmetric	*d*	3.24	Difficult to handle	Symmetric	*d*	2.268	Difficult to handle	Symmetric	*d*	1.296	Difficult to handle	Symmetric	*d*	1.62
			½ sym	*e*	4.932		½ sym	*e*	3.452		½ sym	*e*	1.973		½ sym	*e*	2.466
			Non sym	*f*	5.868		Non sym	*f*	4.107		Non sym	*f*	2.347		Non sym	*f*	2.934
2	**Apply[2]**	Surface S[cm2]	S<5	*a*	3.2	Surface [cm2]	S<5	*a*	2.2	Surface [cm2]	S<5	*a*	1.2	Surface [cm2]	S<5	*a*	1.5
			5≤S<50	*b*	7.15		5≤S<50	*b*	4.97		5≤S<50	*b*	2.8		5≤S<50	*b*	3.5
			50≤S<100	*c*	11.05		50≤S<100	*c*	7.72		50≤S<100	*c*	4.4		50≤S<100	*c*	5.5
			100≤S<500	*d*	16.67		100≤S<500	*d*	11.67		100≤S<500	*d*	6.67		100≤S<500	*d*	8.337
...	...		...	...	.		...		...	...	...			.	.		...
8	**Get[1,3]**	Small, medium, large parts easily grasped (thick.>2mm)		*a*	0.287	Small, medium, large parts easily grasped (thick.>2mm)		*a*	0.198	Small, medium, large parts easily grasped (thick.>2mm)		*a*	0.109	Standard gripper or no need to change to special gripper	Thickness >2mm)	*a*	0.1147
		Very small objects or tool (thickness≤2mm)		*b*	0.538	Very small objects or tool (thickness≤2mm)		*b*	0.375	Very small objects or tool (thickness≤2mm)		*b*	0.212		Thickness ≤2mm)	*b*	0.2155
		Object jumbled with other objects so that select occur		*c*	0.428	Object jumbled with other objects so that select occur		*c*	0.378	Object jumbled with other objects so that select occur		*c*	0.329		Object jumbled with others	*c*	0.171
														Need to change gripper	Thickness >2mm)	*d*	2.2147
															Thickness ≤2mm)	*e*	2.3155
															Object jumbled with others	*f*	2.271
...	...		...	...	.		...		...	...	...			...	.		...
30	**Walk[1,2,3]**	Distance d[m]	d≈4	*a*	2.8	Distance d[m]	d≈4	*a*	2	Distance d[m]	d≈4	*a*	1.2	Distance d[m]	d≈4	*a*	1.5
			4<d<7	*b*	3.5		4≤d<7	*b*	2.5		4≤d<7	*b*	1.5		4≤d<7	*b*	1.8
			7≤d<9	*c*	4.8		7≤d<9	*c*	3.35		7≤d<9	*c*	1.9		7≤d<9	*c*	2.35
			9<d<12	*d*	6.2		9<d<12	*d*	4.35		9<d<12	*d*	2.5		9<d<12	*d*	3.5

[1]MTM (Maynard & Stegemerten, 1948) helped definition of time impacting criteria and values for manual; [2]Timekeeping and videos from a manufacturer helped the time estimates; [3]DFA time estimated of Boothroyd et al (Boothroyd, Dewhurst, & Knight, 2011) helped the estimates; [4]MT estimates are computed as average of M and A values

The time estimation is based on the performed assembly tasks and particularly based on its corresponding motions that can be obtained by decomposition of the tasks. The motions time estimates depend on: The type of the motion to be performed, the selected LoA of the resource performing the task (manual, manual with automated assistance, automatic, or robotic), and the processed elements features intervening in the complexity of the assembly operation (e.g. size, weight, shape). This impacts consequently the time duration of the assembly motions and can increase or decrease the required time for performing the assembly operations. For example the required time to 'get' (or handle) an enough thick part easy to handle is less important than the time to do it for a thin or sliding one. This can be the same for handling a light part compared to the same motion for a heavy one. Yet, the assembly structure given by ASML architecture has to be also involved in the process time estimation. This is studied in next section 5.5.

5.5. Rules for time estimation

In the proposed time estimation approach, as previously mentioned, we use ASML modelling to handle complex processes with various resources, LoAs, assembly operations, and different structures and architectures.

To handle these complexities, we propose rules to be applied on the ASML model and the obtained tasks time estimates using previously presented database. As the first aim of the time estimation is to perform appropriate resources assignment that satisfies required productivity. As the productivity of a process is given by the slowest resource throughout the workshop, we propose rules to be applied on the level of each resource of the process model.

Inside each ASML represented resource, the model architecture influences the resource time estimation because of the possibility of using serial, parallel, or choice executions, due to product assembly sequencing or resources constraints. Consequently, we distinguish these different possible cases to compute resulting time estimation, accordingly to the different 3 possible structures, as follows:

- **Case of a serial schedule**

Time estimation of serial (or sequential) assembly motions is given by the sum of the time values of each motion in the sequential sub-sequence (Figure 13).

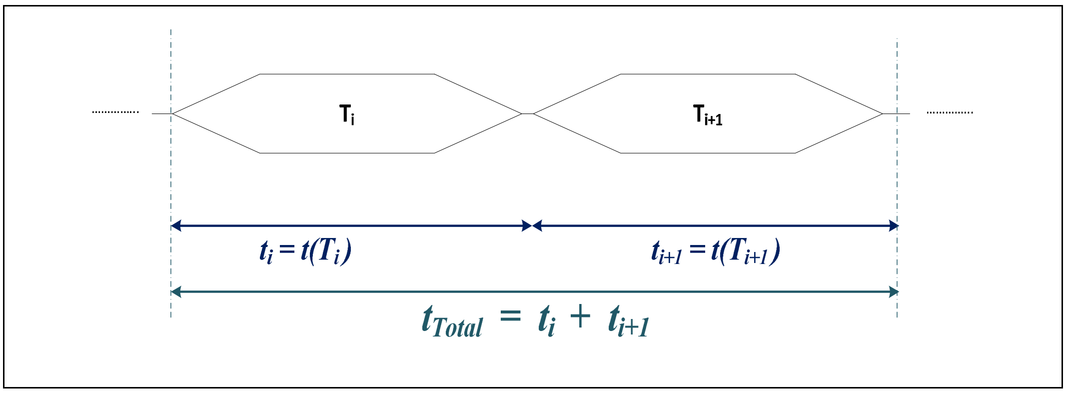


Figure 13: Time estimation for serial tasks

- **Case of parallel sub-sequences**

For the case of assembly sub-sequences to be executed in parallel, the resulting time estimation is given by the maximum among the time estimations of the different sub-sequences (Figure 14).

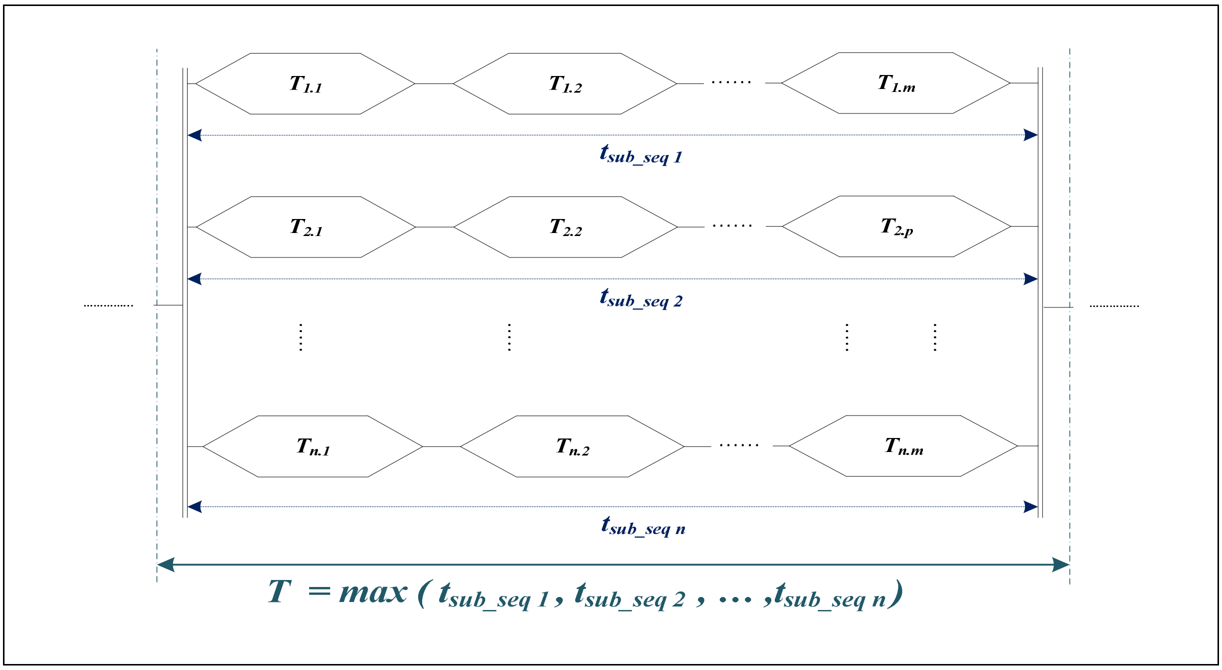


Figure 14: Time estimation for parallel sub-sequences of tasks

- **Case of choice sub-sequences**

If there are choices for executing sub-sequences, the resulting time by precaution is also the maximum among the time estimations of different sub-sequences leading consequently to a majored time estimation value ensuring the assemblies whatever are the conditions. The semantic differs considerably from the AND divergence/convergence but the time computation procedure is similar (Figure 15).

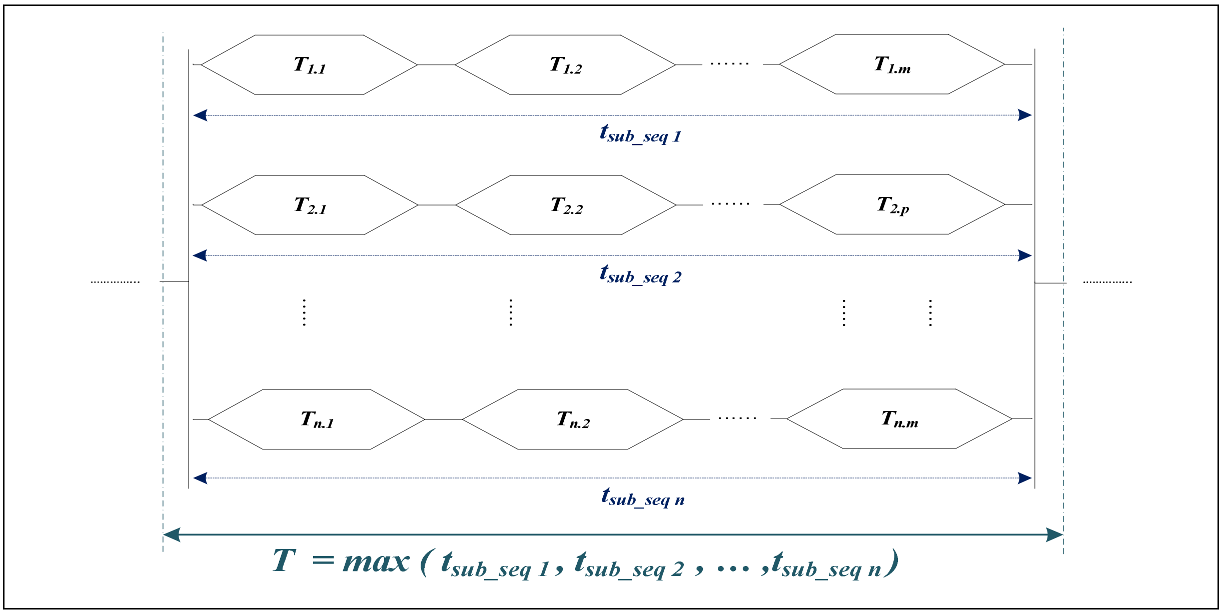


Figure 15: Time estimation for choice sub-sequences

5.6. Related Time-based applications to LoA decision

The proposed approach of time estimation is central as a framework of industrial assembly systems design, analysis, and optimization. It allows deducing multiple other indicators and important factors allowing to evaluate, adjust, and compare assembly systems alternatives with various automation options for the sake of LoA decision making.

We present in this section, first, how to compute for a given automation configuration, using the proposed approach, the corresponding takt-time, process cadence, and cycle time (section 5.6.1). These indicators allow the evaluation of the given configuration with regard to productivity targets and its adjustment to reach the required cadence. We propose then in section 5.6.2 an early phase balancing issue to ensure to appropriate dimensioning of the configuration. The resulting adjusted configuration, accompanied by the time estimates and time performance indicators, can also represent a valuable support to cost estimation. This is explained in more details in section 5.6.3.

5.6.1. Productivity performance indicators

The mentioned indicators (takt-time, cadence, and cycle time) are crucial for a production system analysis. They should be taken into account in the LoA decision approach and line dimensioning as will be detailed in section 5.6.2. In this context, we propose the computation of the indicators as follows:

- **Takt-Time computation**

The Takt Time (TT) defined as the period of time after which we have repeatedly a new assembled product leaving the assembly system, is given by the slowest resource throughout the workshop. It can be then computed as the maximum assembly time per resource considering all the resources of the ASML model. The assembly time per resource has to be computed considering the defined rules in section 5.5.

- **Process productivity or cadence computation**

When the TT is determined, the process productivity (PP) can be simply determined, as widely known, by the following formula: $PP = \frac{1}{TT}$. The quotient represents the number of parts that the process can produce by time unit. For example, if TT is in seconds, PP will inform the number of parts that can be assembled per second for the given process configuration.

PP and identically TT indicators are important in systems design. They will make possible, when considering planned production information, such as planned volume to be produced or production life, evaluating the process suitability from speed or cadence point of views. They allow anticipating the adjustment of automation alternatives and their optimization. This will be more thoroughly detailed in next sub-section 5.6.2.

- **Cycle time computation**

Based on determined assembly time estimates for all resources of the given ASML model, cycle time can be calculated. The assembly time per resource has to be computed with respect of the rules defined in section 5.5. Regarding the obtained resources time estimates, we propose using the critical path approach to compute the cycle time for the considered assembly system alternative. For basics concerning the critical path approach, interested readers can find details in (Cleland & King, 1988).

5.6.2. Resources appropriate assignment: early phase balancing

Lean manufacturing includes different principles aiming at providing best practices and optimizing production systems. Some of the principles concern production organisation and resources synchronization to minimize or eliminate intermediate production buffers between resources. Our objective in this section is to practically use the mentioned Lean principles to enhance resources assignment with a suitable and optimized adjustment of assigning resources to the process model's tasks. In this way, considering planned production information, particularly the required cadence, a designed ASML model can be optimized from resources assignment point of view with consideration of a given scenario selected LoAs.

In order to address the productivity requirements with a coherent balancing, for each resource R_i of the model and considering its selected LoA noted α_i (α_i can have the values, manual, manual with automated tool, automatic, or robotic) and corresponding time estimates, the resources allocation can be appropriately verified and adjusted. To do so, we define the aggregation "RP_i/α_i" of each resource Productivity (RP_i) and selected LoA (α_i) which represents an input information of the planned production. The resource productivity RP_i/α_i is computable similarly to the process productivity by the inverse of obtained time estimation of the resource. To ensure a correct resources assignment, productivity RP_i/α_i of each resource *i* should converge to the required productivity target PP. Three cases can be distinguished. We present the different cases and the solution to be performed to adjust the process:

5.6.2.1. Case 1: resource i is appropriate from cadence point of view

If ($RP_i/\alpha_i \approx PP_{Required}$), then the resource *i* speed is equal or approximately equal to the required cadence. This means that resource *i* will be enough productive compared to the required cadence. In this case it is already well dimensioned from productivity point of view.

5.6.2.2. Case 2: resource i is significantly slower than required

If ($RP_i/\alpha_i < PP_{Required}$), then resource *i* is much slower than required according to the LoA α_i and the assigned task. In this case, two scenarios can be explored as solutions:

- Changing to a faster LoA allowing interesting time savings, if no resource constraints exist.
- Duplicating the resource *i* in the selected LoA α_i. In this case the resource *i* has to be duplicated *n* times to reach the cadence target. The factor *n* by which the same resource is duplicated will divide the resource cadence *n* times until reaching the required cadence.

5.6.2.3. Case 3: resource i is significantly faster than required

If ($RP_i/\alpha > PP_{Required}$), meaning that the resource *i* is much faster than required for the LoA α_i compared to the required cadence. In this case, we propose the following solutions to decrease the speed for many reasons (workload maximizing, number of resources minimizing, and then their corresponding investment cost, and so on):

- Changing for a slower LoA allowing decreasing productivity and adjusting it to what required if no constraints of resources exist.
- If resource speed is adjustable, consider speed regulation especially if LoA is imposed by manufacturer for the given tasks. This can be possible for certain tasks in certain LoAs, such as robotic or automatic soldering assembly for example.
- Considering more tasks for the same resource *i*, in sequential execution, in order to increase resource cycle time occupation (sum of tasks time estimates), then decrease its cadence. To be feasible, compatibilities should be managed. We distinguish 2 types:

- **Compatibility with regard to resources LoAs**

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, even of different natures, can be sequentially performed by a same selected flexible resource if the cadence requirement is still fulfilled. This is less commonly used for less flexible resources. This flexibility that we here call compatibility with regard to resources should be checked for the different resources LoAs before allowing tasks grouping in a same resource.

- **Compatibility with regard to tasks natures**

In this case, compatibility is more related to the tasks. Tasks are generally compatible if they are of the same nature or if they are selected in same LoAs. For tasks natures' compatibility, a compatibility matrix including all the given assembly process modelled tasks can be useful. 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, for example, 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 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 $\mathrm{M}_{\mathrm{n\times n\times m}}$ where n is the number of tasks in the ASML model and m the number of possible automation levels in the LoA scale. This matrix will be used in chapter 7 (section 7.4.1) with the optimization model and the matrix is labelled $T\_Comp_{i1\times i2\times k}$.

The 3 different cases of balancing with their proposed solutions allow adjusting the configuration to the required planned production performances. The study is based on $\mathrm{RP_i}/\alpha_i$ of Resources Productivity ($\mathrm{RP_i}$) and their selected LoA (α_i).

It could be possible to similarly work directly with time estimates. In this case, resources Takt-Time ($\mathrm{TT_i}$) should be used instead of $\mathrm{RP_i}$. The purpose would be adjusting $\mathrm{TT_i}/\alpha_i$ to the required process Takt-Time ($\mathrm{TT_{Required}}$). In that situation, Case 1 would be the same ($\mathrm{TT_i}/\alpha_i \approx \mathrm{TT_{Required}}$), Case 2 would be ($\mathrm{TT_i}/\alpha_i > \mathrm{TT_{Required}}$), and Case 3 would be ($\mathrm{TT_i}/\alpha_i < \mathrm{TT_{Required}}$).

Once an appropriate configuration ensuring the planned production cadence is obtained, one of the most important issues can be to estimate the resulting cost of the configuration with resources assignment, duplications, and LoAs to better support to LoA decision. This issue is presented in section 5.6.3.

5.6.3. A basis to cost estimation

The proposed time estimation approach may support estimating the assembly cost of automation alternatives. In fact, for a given alternative, when estimating the time to assemble a product for each resource, corresponding assembly cost can be obtained. This requires a cost estimation model which should be available and applicable in this particular early phase. The resource duplication determined by balancing approach explained in sub-section 5.6.2 should be taken into account in the estimation because they represent additional resources in the process.

These developments can also help performing more advanced cost analyses and comparisons of alternatives from profitability point of view. Cost analyses are important for manufacturers as they represent according to the literature and to our discussions with manufacturers one of the most preponderant decision criteria. The topic of cost estimation will be tackled in next chapter 6 with a cost model determination for the automation decision issue.

5.7. A validation example: process time estimation and productivity analysis

To more practically explain the use of the time estimation approach and highlight its multiple advantages, we show in this section a validation on a simple example by application of the different steps and computations presented through this chapter.

❖ **Process time estimation**

We apply in this section the different steps of the proposed time estimation procedure.

- **The preliminary steps**:

As the time estimation approach starts by the initial generic ASML model of tasks, we consider a product assembly of 6 parts shown by its ASML model in Figure 16. The model contains 5 tasks: T1 consisting in a screwing task, riveting for T2, gluing for T3, clipping for T4, and press fitting for T5. The architecture and assembly sequence are shown in the ASML model shown on Figure 16.

The next step is to consider a feasible combination of tasks LoAs as an automation scenario that can be tested. The feasibility is related to the LoA criteria consideration in tasks analyses in the different selectable LoAs. In our case, we use the 4 levels LoA scale of Table 4 (chapter 1). We consider the automation scenario of Figure 17 as a possible alternative performed after analyses with regard to loA criteria. We use the same colors codes previously used in chapter 4 to graphically distinguish LoAs in the representation (Figure 17; Table 19): grey for 'M' green for 'MT', blue for 'A', and orange for 'R' (column 2 in Table 19).

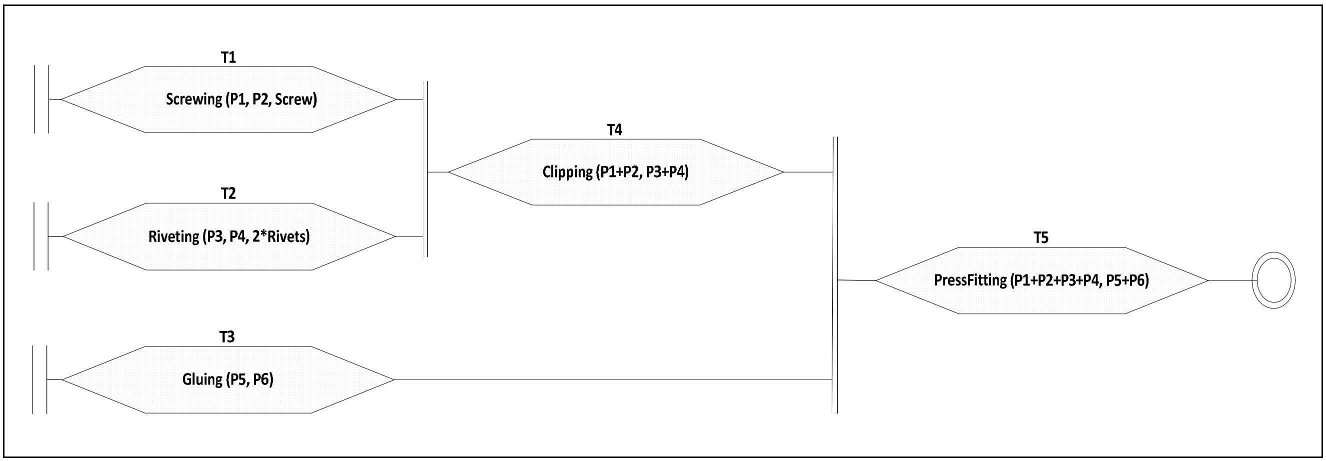


Figure 16: A product assembly example shown by its ASML tasks representation

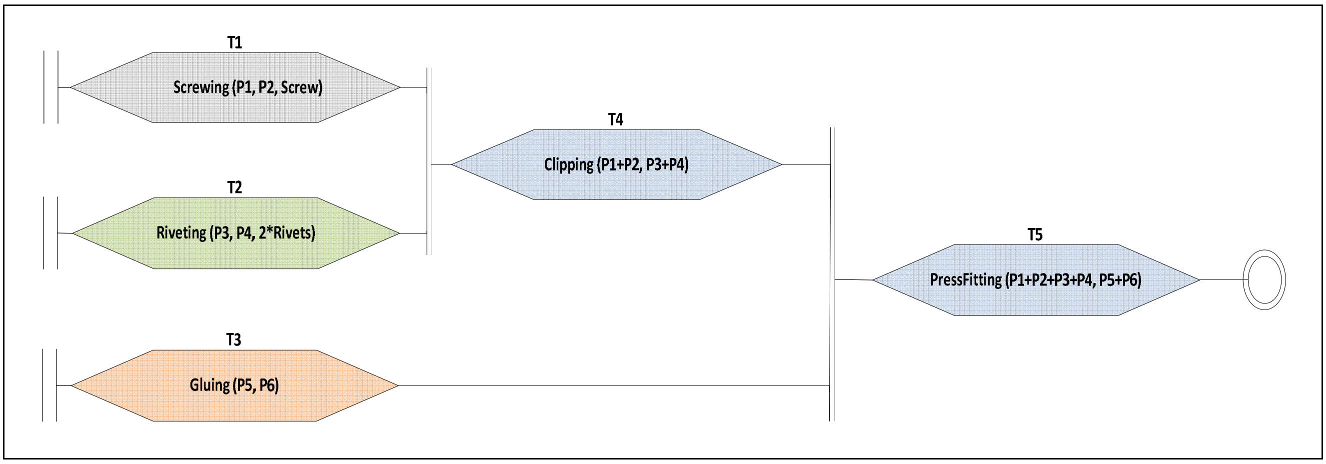


Figure 17: An automation combination to process tasks for a product assembly example

- **Step 1**:

Once tasks and LoAs are associated in the scenario by application of the preliminary steps of the time estimation approach, step 1 can be applied. This step consists in converting each task considering selected LoA to each corresponding motions (chapter 4 – section 4.5.2). Motion time estimates according to parts features can be then retrieved. The results are shown in Table 19.

Table 19: Tasks convertion to motion for the product assembly example

Task	LoA	Motions	Digits	Time [s]	
T1 Screwing(M,P1,P2,Screw)	Manual	Identify(P1)	11-M-a	1.0	44.54
		Identify(P1)	11-M-a	1.0	
		Get(P1)	8-M-c	0.428	
		Identify(P2)	11-M-a	1.0	
		Get(P2)	8-M-a	0.287	
		Identify(screw)	11-M-a	1.0	
		Get(screw)	8-M-b	0.538	
		Handstart(scw,P1,P2)	9-M-a	34.0	
		Identify(screwDrvr)	11-M-a	1.0	
		Get(screwDriver)	8-M-a	0.287	
		Tighten(screw,P1,P2)	27-M-a	4.0	
T2 Riveting(MT,P3,P4,S2*Rivets)	Manual with automated tool	Identify(P3)	11-MT-a	1.0	35.674
		Get(P3)	8- MT-c	0.378	
		Identify(P4)	11-MT-a	1.0	
		Get(P4)	8- MT-b	0.375	
		Identify(Rivets)	11-MT-a	1.0	
		2*Get(Rivets)	8- MT -a	0.396	
		Handstart(Rvt, P3, P4)	9- MT -a	24.15	
		Identify(RivetingTool)	11-MT-a	1.0	
		Get(RivetingTool)	8- MT -b	0.375	
		2*Tighten(rvt,P3,P4,ST)	27- MT-a	6.0	

Task	LoA	Motions	Digits	Time [s]	
T3 Gluing(R,P5,P6)	Robotic	Identify(P5)	11-R-a	0.25	7.514
		Get(P5)	8-R-b	0.215	
		Apply(Glue, P5, 10cm2)	2-R-b	3.5	
		Identify(P6)	11-R-a	0.25	
		Get(P6)	8-R-a	0.115	
		Align(P5,P6)	1-R-f	2.934	
		Engage(P5,P6)	6-R-b	0.25	
T4 Clipping (A,P2,P3)	Automatic	Identify(P2)	11-A-a	0.15	2.89
		Identify(P3)	11-A-a	0.15	
		Align(P2,P3)	1-A-f	2.34	
		Press (P2,P3)	19-A-a	0.25	
T5 PressFitting (A,P4,P6)	Automatic	Identify(P4)	11-A-a	0.15	2.523
		Identify(P6)	11-A-a	0.15	
		Align(P4,P6)	1-A-e	1.973	
		Press (P4,P6)	19-A-a	0.25	

- **Step 2**:

After tasks decomposition to motions and time estimates retrieval in step 1, tasks can be time estimated in this step. The tasks time estimates can be computed as the sum of their motions in the decomposition. The results are also shown in Table 19 in last column in front of each task. As shown, the results are reported as follows:

* 45.54s for T1
* 36.674s for T2
* 7.514s for T3
* 2.89s for T4
* 2.523s for T5

- **Step 3**:

This step consists in resources assignment to tasks with respect to selected LoAs to tasks and to defined compatibilities between tasks. Multiple solutions of resources assignments may exist, even for selected LoAs to task. A scenario will then represent one of the possible alternatives.

We assume in this example that all tasks are compatible with regard to resources and to tasks (section 5.6.2.3).

We consider in this example first the simplest configuration of resources assignment consisting in assigning a separate resource to each task as shown in Figure 18.

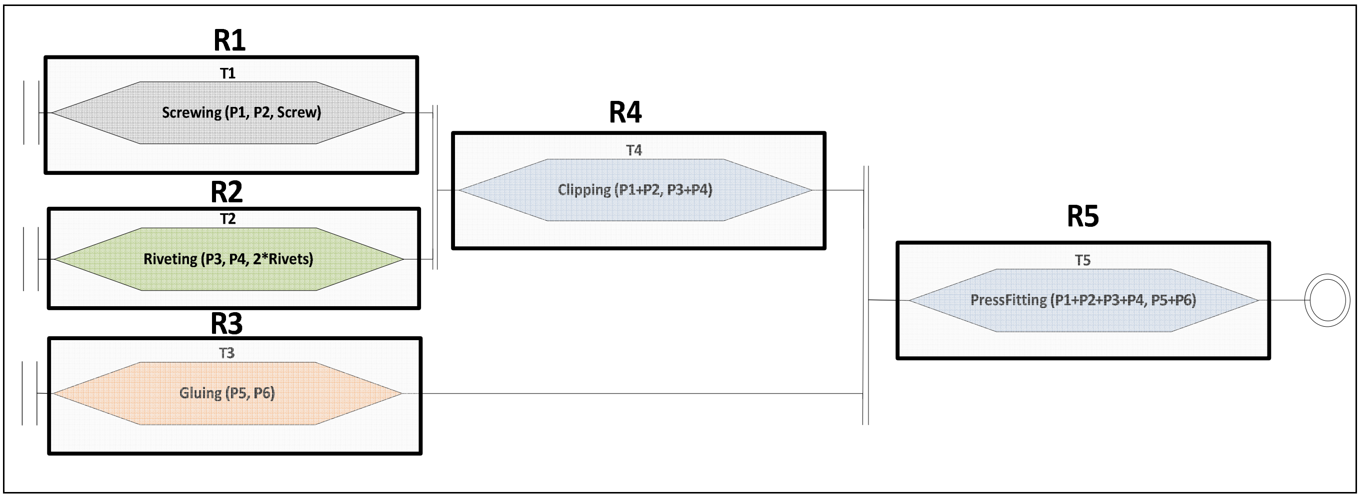


Figure 18: An alternative of resources assignments to tasks

- **Step 4**:

Step 4 consists in verifying and adjusting the obtained configuration to make sure that the proposed alternative can reach the required productivity. We perform this in 2 phases: performance indicators computation, then resources adjustment.

- **Performance indicators computation**

As planned production information and requirements, we assume a required takt-time of TT= 9 seconds, consequently a process productivity of PP=1/9 = 0.11 assemblies/sec.

To verify the suitability of the configuration to reach this required productivity, it is important to compute the presented productivity performance indicators of the obtained first configuration.

We start by computing the resource estimated time (TTi) and productivity (RPi), for each resource.

- For resource R1, TT_1= 44.54 sec ; RP_1= 1/44.54 = 0.0224 assemblies/sec < $PP_{Required}$
- For resource R2, TT_2= 35.674; RP_2= 1/35.674 = 0.028 assemblies/sec < $PP_{Required}$
- For resource R3, TT_3= 7.514; RP_3= 1/7.514 = 0.133 assemblies/sec > $PP_{Required}$
- For resource R4, TT_4=2.89; RP_4=1/2.89=0.346 assemblies/sec > $PP_{Required}$
- For resource R5, TT_5= 2.523; RP_5= 1/2.523= 0.396 assemblies/sec > $PP_{Required}$

Concerning the process cycle time for this configuration, it is of 44.54+2.89+2.523 = 49.953 seconds.

The tested process productivity is defined as the one of the slowest resource. It is then the minimum among cadences over the resources. The value is then the one of R1, and is of 0.0224 assemblies per second.

We can see according to these performance indicators computations, particularly the process productivity, that the configuration does not ensure reaching the required productivity of 0.11 assemblies/sec. This is due to one or more resources slowing the whole production workshop.

A check should be performed resource per resource to verify that each resource is enough productive.

According to the comparisons previously computed, it can be seen that the slow resources are R1(0.0224 assemblies/sec) and R2 (0.028 assemblies/sec). For R3, R4, and R4, they satisfy the productivity constraint.

- **Process balancing to reach the required cadence**

To adjust this configuration, resources duplications can be used as solution (section 5.6.2.2):

* Resource R1 should be duplicated 0.11/0.0224=4.9107 → 5 times.

* Resource R2 should be duplicated 0.11/0.028 = 3.928 → 4 times.

This duplication is sufficient to adjust the process to the required cadence

For a better optimization, it can be observed that R4 and R5 are much more productive than required. As we assumed that all tasks are assumed to be compatible with regard to resources and to tasks, and as T4 and T5 are selected in the same LoA (automatic machine), then T4 and T5 can be grouped in the same resource (case 3 – section 5.6.2.3). This allows then reducing the number of resources of the process, and then the resulting costs as one machine is saved.

If grouped, the resulting automatic resource will have a takt-time of 2.89+2.523=5.413 sec, and then a productivity of 0.184 assemblies/sec, which is still enough productive compared to the required productivity.

Consequently, the process adjustment procedure leads to the represented configuration in Figure 19.

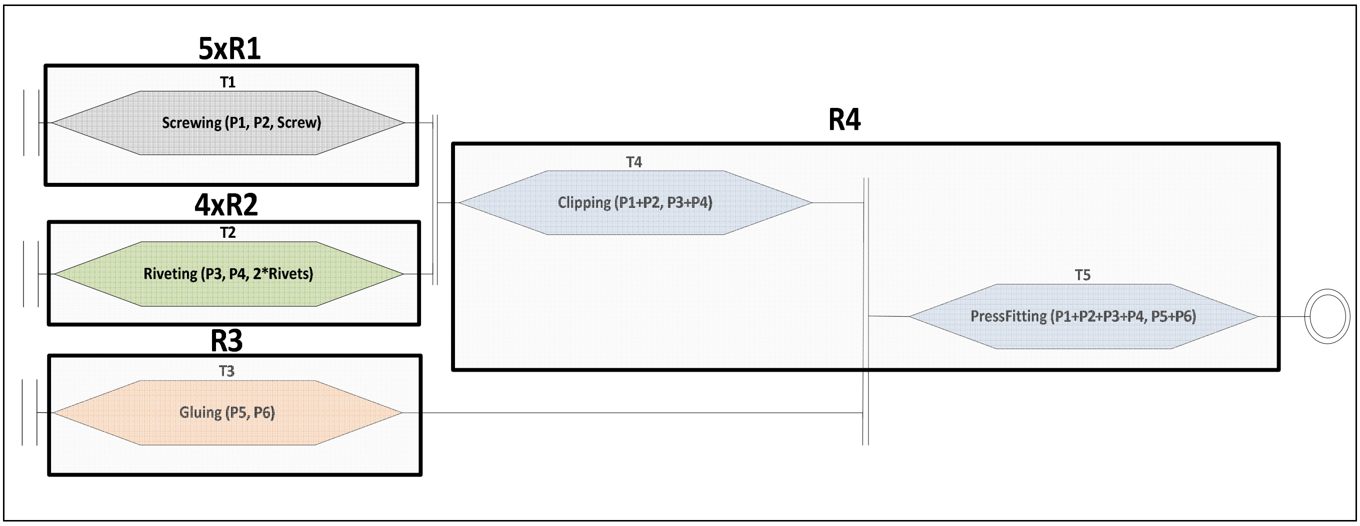


Figure 19: A configuration of automation option for a balanced process

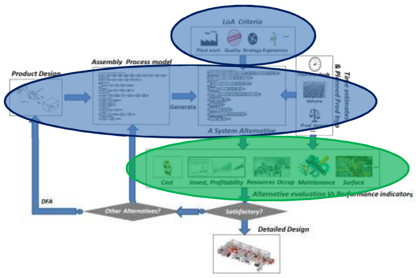
The new resulting productivity of the shown configuration in Figure 19 is then as follows:

$PP_{AdjustedProcess}$ = min {5*0.0224 ; 4*0.028 ; 0.133 ; 0.1847} = 0.112 assemblies/sec > $PP_{Required}$.

The adjusted configuration allows consequently reaching the required productivity of 0.011 assemblies per second with an optimized configuration of resources assignments adjusted to the targeted productivity.

5.8. Conclusion

The proposed time estimation approach enables one to estimate assembly time of ASML modelled modelled processes with selected automation options throughout the process. To do so, the approach proposes a database of motions time estimates using the proposed standardized vocabulary of assembly motions previously presented in chapter 4. The time values retrieval involves the motions types, the selected LoAs, the parts design features and complexity impact on the assembly time. To handle complex processes from architecture point of view, rules are defined for the different possible ASML structures. The proposals allow also to adjust the alternative and resources assignments using a defined pre-balancing approach for a better productivity suitability with regard to the planned production cadence. This is facilitated by production performances indicators computation that are made possible to be computed thanks to the proposed approach. These indicators consist in the takt-time, production cadence, and cycle time. The developments may also represent a basis to allow assembly cost estimation once a cost model, applicable during the LoA decision early phase, is available. Such cost model can be associated to these developments in order to offer the possibility to assess and compare assembly systems automation scenarios profitability. For this end, assembly cost estimation for the issue of automation decision making is tackled in next chapter 6.



Chapter 6. Automation Alternatives Cost Estimation[4,5]

Abstract

This chapter is devoted to an integrated cost estimation approach proposal aiming at supporting assembly systems design by automation alternatives profitability estimation. The finality is to help automation decision making with regard to the economic aspect. The cost criterion will contribute to objectively evaluate, analyze, and compare automation scenarios to lead to the optimal configuration. Such strategic decision has to be efficiently made because of the heavy investments and due to the long term profitability consequences that a solution can generate. This chapter presents literature findings in cost estimation for automation decision. It proposes a complete approach of estimation by a complete cost model integrating multiple literature models, features and cost drivers. The model takes into account the early design phase specificities particularly the non-abundance of input information during such phase. The result provides the possibility to predict future assembly processes costs with automation options impact consideration and long-term profitability anticipation since the early phase of assembly systems design.

6.1. Introduction

In today's difficult market environment characterized by an international competition and globalized production, being competitive becomes a complex issue. Automation represents one of the possible alternatives to keep local productions and avoid delocalization solutions particularly in high labor rate countries. However, based on our team's experience and industrial contacts and visits in leaders companies in France, Germany, and the United-States, it was seen that in spite of the high labor rate in these countries, manual assembly is still significantly in use even if advanced automated processes allowing handling almost types of assembly activities are available. Accurate cost estimations with significant consideration of several cost drivers to estimate the profitability and compare industrial implementation and future production comparisons seem are needed.

---

[4] A review in cost estimation for automation decision is published in (Salmi A. , David, Summers, & Blanco, A review of cost estimation models for determining assembly automation level, 2016)

[5] A cost model for automation decision issue is proposed: (Salmi A. , David, Blanco, & Summers, An integrated cost estimation approach to support automation decision in assembly systems design, In submission)

Cost estimation is a vital concern of every manufacturing firm and is tackled in every organization (Downs & Trappey, 1992; Eklin, Arzi, & Shtub, 2009; Winchell, 1989). Our purpose is to provide tools to support cost estimators in assembly, particularly to foresee at an early phase the profitability of different systems scenarios. The relationship between the cost and Level of Automation (LoA) decision was evoked in almost literature works in the area of automation decision making. The majority of this literature on LoA considers the cost as one of the most important and preponderant criteria (Boothroyd et al., 2011; Frohm et al., 2006; Gorlach & Wessel, 2008; Lay & Schirrmeister, 2001; Parasuraman, Sheridan, & Wickens, 2000; Windmark, Gabrielson, Andersson, & StŒhl, 2012). It was also underlined that cost estimation may be efficiently used to provide decision makers with the necessary information to make sound resource allocation choices so that economically appropriate crucial decisions can be made (Downs & Trappey, 1992; Liebers & Kals, 1997).

To predict the cost of a future assembly system alternative, a suitable cost model is needed. Based on literature models review, this model should be tailored to the issue because of the specificities of such early phase estimation and automation decision issue. The phase is basically characterized by the lack of information about the process, inexistent or not well known during this phase, with generally no clear idea, representation, or fixed architecture of the process, and unabundant production information. The challenge is then to handle all these specificities of the issue and propose an appropriate methodology to help to predict the cost of possible assembly systems alternatives with automation possibilities.

This work considers the findings of our review paper in the field of cost estimation (Salmi, et al., 2016) where literature models in the field were reviewed and important estimation aspects and cost drivers to be considered were identified. In continuity, we propose in this chapter a complete and integrated methodology allowing to predict and compare cost performances of future automation alternatives candidates.

The chapter is organized as follows: in section 6.2 we highlight the importance of cost estimation for automation decision. In section 6.3, the challenges and particularities imposed by automation decision are highlighted. In section 6.4, we provide a review in the field of cost estimation literature by most important findings presentation. The review leads to a need to define a tailored approach for automation decision. We propose in section 6.5 a cost estimation approach including: process modelling, cost model equations, and cost performances indicators computation allowing to practically compare alternatives of assembly processes. The rest of the chapter is then dedicated to cost modelling and cost performance indicators computation. We propose then in section 6.6 a parametric exhaustive definition of a cost model to be applied on a resource with consideration of possible automation levels that can be selected to it. The model is illustrated in section 6.7. The extension of the approach to handle the computation of assembly cost for multiple resources model is presented in section 6.8. Cost performance indicators are presented in section 6.9 with equations allowing their computation. In section 6.10, a validation of the approach on an industrial case study with multiple resources is provided with thorough analysis and comments. A discussion is tackled in section 6.11. Finally, the paper is concluded in section 6.12.

6.2. Importance of cost estimation for automation decision

The relationship between the cost and Level of Automation (LoA) decision was evoked in different literature works in the field of automation deciding. The majority of this literature on LoA considers the cost as one of the most important and preponderant criteria (Lay & Schirrmeister, 2001) (Frohm, Granell, Winroth, & Stahre, 2006) (Parasuraman, Sheridan, & Wickens, 2000) (Windmark, et al. 2012) (Boothroyd, Dewhurst et Knight 2011) (Gorlach et Wessel 2008). Some of them are focused on existing processes and are questioning on the possibility of improving the system. Others are focused on new systems design and are then predicting the cost of new productions. This corresponds more to our objective. These different models are reviewed through literature sections of this chapter.

Concerning cost estimation in automation decision literature, Windmark et al (2012) noted that partial automation is introduced since some elements involved in the production can be costly or particularly difficult to automate. In the same way, other researchers pointed out (Gorlach & Wessel, 2008) that a balanced combination of manual and automated processes allows reducing manufacturing costs. It was also underlined that the two dominant factors motivating automating processes are: first cost efficiency, then reducing negative effects of working environment that can represent danger to health (Windmark, Gabrielson, Andersson, & StŒhl, 2012). Concerning the cost, the authors mentioned that to be particularly profitable, high automation generally requires a high production volume. Profitability curves of three automation levels (manual, automatic, and robotic) costs with regards to number of product parts to assemble in a DFA perspective were drawn since the first DFA works (Boothroyd G. , 1987). Most important results of the study of Boothroyd show that automatic is the most profitable when the number of parts is high or medium. The profitability margins between the different technologies decrease when the number of parts decreases and converge to a same value for a two parts assembly product. Multiple critics can be addressed to these results. These interpretations are less credible nowadays because of the significant technological progress compared to manufacturing systems of that period (1987). Also, such basic experimental results without any demonstration, proof, or model, needs to be updated to the current context specificities and justified using a concrete and generic model supporting the bases of such results. The only consideration of the number of parts as an automation decision criterion or profitability parameter represents also a weakness, while multiple criteria should be taken into account (chapter 3). In another work, Gorlach and Wessel (2008) pointed out that an optimal level of automation of manufacturing systems can only be obtained if all relevant aspects of the manufacturing process are taken into account and optimum levels in terms of cost, and others consisting in productivity, quality and flexibility, are reached. It was also underlined that cost estimation may be efficiently used to provide decision makers with the necessary information to make sound resource allocation choices so that economically appropriate crucial decisions can be made (Liebers et Kals 1997) (Downs et Trappey 1992). It should be underlined that different studies indicate that some organizations experienced losses in productivity due to investment in manufacturing technology (Sim 2001). A reliable cost prediction with consideration of technologies levels, or LoAs, and their corresponding production capabilities can then help to avoid such failures.

6.3. The imposed challenges by the LoA decision issue specifications

Because of the particular early phase of automation decision, the cost estimation represents a complex task. The complexity reasons are detailed in following sub-sections.

6.3.1. A conceptual design of the process to cost estimate

Predicting the cost of a future production is a difficult process (Koonce, et al. 2003). The delicateness is resulting from the phase of this estimation: a phase during which the process does not exist yet and is still under design. It is consequently not exhaustively known. A lot of information about this process and about the production is still lacking or missing. Pehrsson et al (Pehrsson, Ng et Stockton 2013) mentioned that a cost analysis or a simple parametric cost calculation alone is not enough to support decision-making in the development of production systems: expertise from other disciplines, such as industrial engineering and operation management, is required to support such complex decision-making based on both costing and operational information. These multidisciplinary competences should enable predicting, based on a conceptual system design, performances and indicators that can be crucial in cost estimation.

6.3.2. The non-availability of cost input information

The early phase of assembly cost estimation is characterized by a non-abundance of information and data to use for cost computation, basically because of the absence of the real assembly process. Much of information is then lacking. According to (Needy, Billo et Warner 1998), a cost model that attempts to include all cost factors tends to disallow both the collection of such data and the proper usage of such data in decision-making. Parameters to consider in the cost computation have to be available and obtainable during that phase. A model using too few parameters will be obviously inaccurate. The cost parameters have then to be rigorously selected: basically available and most impacting ones. Product design and strategic information about planned production should represent the basis of a cost prediction during the early phase of assembly systems design and automation decision.

6.3.3. No conventional cost model exist

In the field of cost estimation, several works exist. Some of the models are dedicated to a specific case solving, others are little more generic. The existence of a multitude of models itself can represent a sign that no conventional, fully generic, or standard model exists. Different classifications of cost models exist in the literature. Some of the relevant existing classifications methods are used in the literature review in section 6.4 to evaluate models and search for a most appropriate one to the purpose of early cost estimation and automation decision. Results of this review are summarized in next section. They confirm the non-existence of generic models for early phase cost estimation and automation decision.

6.4. The literature in cost estimation for automation decision

A review in cost estimation for the sake of supporting automation decision was performed in (Salmi, et al., 2016). We briefly present in this section the major findings of the study. The review sweeps a large time interval from 1984 to 2015 with a heterogeneous set of 32 reviewed cost estimation works belonging to different journals and books in assembly and manufacturing. The list of reviewed models is shown in column 2 of Annex B – Table 1.

The selected models were first classified with regard to 3 criteria:

- The approach type: qualitative (analogical or intuitive) or quantitative (parametric or analytical).
- The granularity level: top down (an estimation at a high level of abstraction) or bottom up (a low granularity level with estimation of elements and sub-elements costs accumulated to a total product cost).
- The applicability phase: early phase or late phase applicability of the model.

By analyses and argumentation, the review underlines the fact that an appropriate cost model to support automation decision should be quantitative (parametric and/or analytic), bottom-up, and applicable during the early phase with regard to the availability of input information to be involved in the estimation. The literature models classification with regard to these criteria is shown in Annex B – Table 1 where suitable classes are highlighted with green colour and unsuitable are red coloured.

Following these classifications and their suitability analyses, all the models among the initial selection that suit the 3 presented criteria were more thoroughly reviewed, described, and analyzed with regard to automation decision purpose in order to search for an appropriate and applicable model to help automation decision. This led to 9 most appropriate models identification. These models were themselves categorized to 2 categories: product design based models and production information based model. The first category considers the product design characteristics and features complexity impact on cost. Basically more complex products are more time consuming in assembly and are then more costly. Literatures techniques to predict time in assembly with databases are then proposed, such as MTM (Maynard & Stegemerten, 1948) or DFA (Boothroyd, Dewhurst, & Knight, 2011) methods. The second category is more conventional and proposes parametric models with cost equations allowing, based on some cost input parameters, to compute the cost. These most appropriate filtered literature models are shown in Table 20 references. Readers interested in the complete review and detailed analyses of the different models are invited to find the work of (Salmi, et al., 2016).

Table 20: Most appropriate cost models candidates to automation decision as identified in (Salmi, et al., 2016)

Product design based	(Dewhurst & Boothroyd, 1988) (Jung, 2002) (H'mida et al., 2006) (Boothroyd et al., 2011) (Quintana & Ciurana, 2011) (Ou-Yang & Lin, 1997) (Shehab & Abdalla, 2002) (Swift & Booker, 2013)
Cost information based	(Son, 1991) (Pehrsson et al., 2011) (Jha, 1992, 1996)

Based on Table 20, the different classifications and filtering led to 8 product design based cost models versus 3 cost information based models. This can be due to the fact that traditionally early phase cost estimation models do not use parametric equations to predict the cost, but basically analogical or intuitive approaches.

It was found that the model of (Boothroyd et al., 2011) is the most complete and satisfactory in this category. The model uses DFA approach with handling and insertion assembly operation time estimates to predict assembly time and cost for multiple mono-automation level processes. Yet, for us the vocabulary using only these two operations (handling and insertion) is not enough representative of assembly environment. It does not include explicitly assembly techniques (such as Welding, riveting, or snap fitting) and also other non-productive motions (such as walk, move, open containers, etc). It does not handle multiple resources or hybrid automation processes with various automation levels. This cost model needs some enhancement and accuracy improvement. Concerning the other category of cost information based models, the model of (Son, 1991) was identified as the most promising due to its exhaustively and multitude of cost drivers considered in an early phase model. The analytic way to compute the cost per jobs or activities is coherent to automation decision issue because it allows the possibility to compute the cost per work area and its associated automation level. Yet, we think a cost per product is more appropriate rather than a cost per unit time or for a planned horizon as defined in (Son, 1991). This should be adapted or considered in the future model we develop for the issue. Multiple cost elements are also enumerated and need equations to be objectively computable, such as resources cost rates. The source of time estimates of the corresponding jobs is also not tackled. A technique allowing obtaining the time values, as MTM or DFA, should be associated with an appropriate adaptation to cover the different jobs types in assembly. This consists in one of the challenges of the future complete approach.

The review and models of Table 20 include early phase models and late phase models, generally more advanced and more exhaustive. In fact, exhaustive parametric equations, basically models belonging to the second category of cost information based models of Table 20, and especially when they are late phase models, can be partially used to enhance the estimation accuracy. The late phase models are not shown in Table 20 because the table represent most appropriate models to LoA while early phase applicability was a filter to obtain the list. Yet, late phase models references and evaluation can be found in the detailed review of (Salmi, et al., 2016). To conclude, a combination of the two categories of Table 20 with an involvement of late phase models equations, eventually after adaptation to the context, can be worth doing. That is why late phase models are considered in the cost model proposal in the equations definition and justification later in section 6.6.

In our detailed review (Salmi, et al., 2016), interesting features (Fi) of literature cost estimation models that can be useful to automation decision are identified with corresponding literature models references shown. We summarize these interesting features in Table 21.

Table 21: Interesting literature models features to automation decision issue (Salmi, et al., 2016)

F1	Product **design** and its **complexity** involvement in the estimation
F2	**Different types of resources and their automation levels** consideration (at least manual and machines)
F3	**Multiple resources** handling in the process (more than one resource)
F4	Use of process **graphic representation** to provide a global view and easier cost estimation
F5	**Decomposition to operations** and **cost per operations** approach / A**ctivity-based** costing
F6	**Time values** determination source approach
F7	**Standardized** approach for **generic** application to assembly, or to manufacturing including assembly
F8	**Cost rate** computation and its considered components **detailed**
F9	**Non-productive** costs / **overheads** considered (set-up, tool change, etc..)

In addition to the identified interesting features, all cost drivers that can be impacting or useful to automation decision are identified in (Salmi, et al., 2016). A list of 12 cost drivers (Dj) to be taken into account for a complete model is summed up in Table 22.

Table 22: Important cost drivers to be considered in cost estimation for automation decision (Salmi, et al., 2016)

D1	Production **volume** / **batch** or **lot** size
D2	**Production life/payback** period/number of **working days**, **hours**, **sec**
D3	**Resources** initial **purchasing** cost or capital investment
D4	Resource **working life** considering resource **depreciation**, **renovation** or **replacement** consideration
D5	Resources **defects (rate) / downtime / stoppage**
D6	**Test/ Inspection/ Quality** control cost
D7	**Rejections / non-conform** assembled parts cost
D8	**Rework** cost of non-conformities (by resource)
D9	**Maintenance** /**supervision/repairing** /machines **faults correcting**
D10	Workstations **Set-up**/ **preparation/reconfiguration/reprogramming**
D11	**Energ**y cost consideration, resources consumption, power
D12	Occupying workstations' **surfaces** cost consideration

To sum-up, the approach should then be applicable during the early phase, quantitative, and low granularity as a bottom up approach. It should consider product design features and complexity, and at a same time should involve the planned production information to maximize the estimation accuracy combining consequently the 2 columns of Table 20. The model should also consider interesting features to LoA decision found in the cost models review and summarized in Table 21. To be exhaustive, the approach has to consider the 12 cost drivers listed in Table 22. As according to the review of (Salmi, et al., 2016) no approach in the literature combines all these features, we propose in this research an integration of all these aspects into one complete and computerizable approach. The approach is proposed starting from next section. It will be later checked in the discussion (section 6.11) with regard to the previously identified requirements for a suitable cost estimation approach to support automation decision.

6.5. A proposed cost estimation methodology

This approach aims at fulfilling the previously enumerated requirements in the review. A simplified scheme of the cost estimation methodology we are proposing is shown in Figure 20.

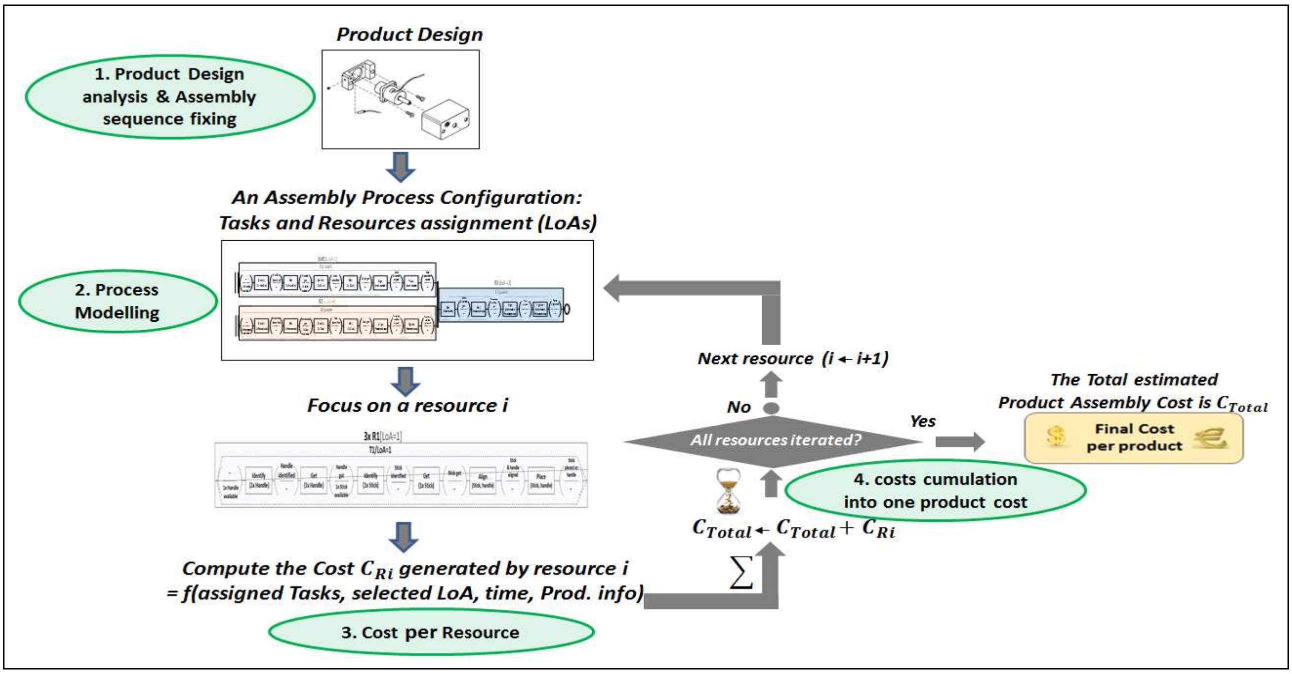


Figure 20: The product assembly cost estimation approach

As it can be seen in Figure 20, the method is based on the product to be assembled design prototype. In step 1, by product design analysis, an assembly sequence can be fixed. Based on the product complexity analysis and the assembly sequence, a generic representation of the assembly process can be then modelled, as shown in step 2 of Figure 20, to prepare to the cost estimation step. This representation describes the processes by assembly operations representation, their sequentially (serial, parallel, choice sequences), and the assigned resources to them. An appropriate modelling language can be useful to ease and standardize this description. The use of ASML language is presented in chapter 4 and 5. Other languages such as Sequence of Operation (SoP) (Bengtsson, et al., 2012) can be tested for this sake of cost estimation. For us, we use ASML to model processes because its high standardization thanks to the rigid rules to graphic modelling and the use of associated standardized vocabularies (chapter 4), and time estimation techniques (chapter 5) which is missing in other less dedicated literature languages to automation decision. The resources assembly cost computation can then be performed in step 3 as shown in Figure 20. This step needs a cost model to allow estimating the assembly cost generated by a given resource with consideration of its LoA. The cost model we propose allowing to solve this issue is presented in details in section 6.6. This represents the core issue of this chapter. The proposed cost model is then illustrated in a one page flowchart in section 6.7. Once the cost per resource is possible, and after independent application to all the process resources with consideration of the associated LoA to each of them, the costs generated by the different resources can be accumulated and summed into a total cost that represents the process assembly cost per product as shown in step 4 of Figure 20. This is treated in details in section 6.8. In section 6.9 we present performance indicators that can be deduced from this approach and that can easier the analysis of an assembly system alternative and comparisons with other alternatives to lead to the identification of most suitable process that should be implemented. The approach is validated in section 6.10 on an industrial case study.

6.6. An integrated cost model proposal to estimate assembly cost per resource

The finality of this section is to define an early phase, bottom-up, quantitative cost model addressing and considering the different features and cost drivers identified in the review section 6.4. To do so, we propose an integrated literature-based cost model allowing to compute of the assembly cost generated by a resource of the process according to its automation level.

To compute this cost, different cost input information are involved and should be entered to the cost model as equations' parameters. These parameters should be already gathered or have to be retrieved from databases as standardized data. Different stakeholders of the company may be involved in the data gathering. The concerned stakeholders or involved actors depend on the company organization, hierarchy, and the internal strategies. Our scope is focused here on providing the model rather than on the data gathering step. We consider data which can be obtainable during such early phase.

In the following model proposal, the different equations are numbered between parentheses (i). The different parameters (P) are also numbered as Pj to easier the reuse of the parameter and avoid redundant definitions of concerned parameters. All used input parameters are listed below the equations. In the case of a first use of the parameter, this parameter is defined by a brief description and its computation equation when necessary. If the parameter is already used before in previous equations, a referencing to the description of the parameters corresponding to its first use is mentioned so that the initial description can be easily found without a need to redescribe it so that redundancies can be avoided.

The proposed cost model provides a cost per product for a given resource of the process. The model should involve the different identified cost drivers (Table 22) and integrate interesting features (Table 21).

The model aims at distinguishing the computation with regard to the LoA of the resource. Concerning the possible LoAs, we use a scale of 4 levels defined in Table 4 – chapter 1. For such a scale, we distinguish 3 categories differentiating the costs:

- **C_L:** Costs characterizing labor resources
- **C_M:** Costs characterizing machined resources: including automatic and robotic
- **C_{LM}:** Common costs to both labor and machined resources.

We detail the computation of the different cost elements **C_L**, **C_M**, and **C_{LM}** in the next 3 sub-sections.

As a result, the assembly cost (**C_{Ri}**) generated by a resource Ri of one of the different LoAs, particularly for the 4 LoA scale (LoA=1-4), can be obtained as follows:

- For LoA = 1 (Manual):

$$\mathbf{C_R = C_L + C_{LM}} \quad (1)$$

- For LoA = 2 (Manual with automated tool):

$$\mathbf{C_R = C_L + C_M + C_{LM}} \quad (2)$$

- For LoA = 3 (Automatic) or LoA = 4 (Robotic):

$$\mathbf{C_R = C_M + C_{LM}} \quad (3)$$

We assume that an automatic or a robotic resource is autonomous. The presence of a human resource for supervision or parts feeding should be considered as an additional independent manual resource. If the assembly operations require continuously a collaborative work between a human and a machine, it should be otherwise represented as a LoA=2, e.g. a manual with automated assistance resource.

The different cost elements we identify in the cost model we propose, by consideration of the cost drivers to be considered, are shown in Figure 21. In This figure, the cost elements are also organized by cost categories (**C_L**, **C_M**, and **C_{LM}**) so that the computation of the cost caused by a use of a resource, according to its LoA, can be performed as described by the equations (1-3).

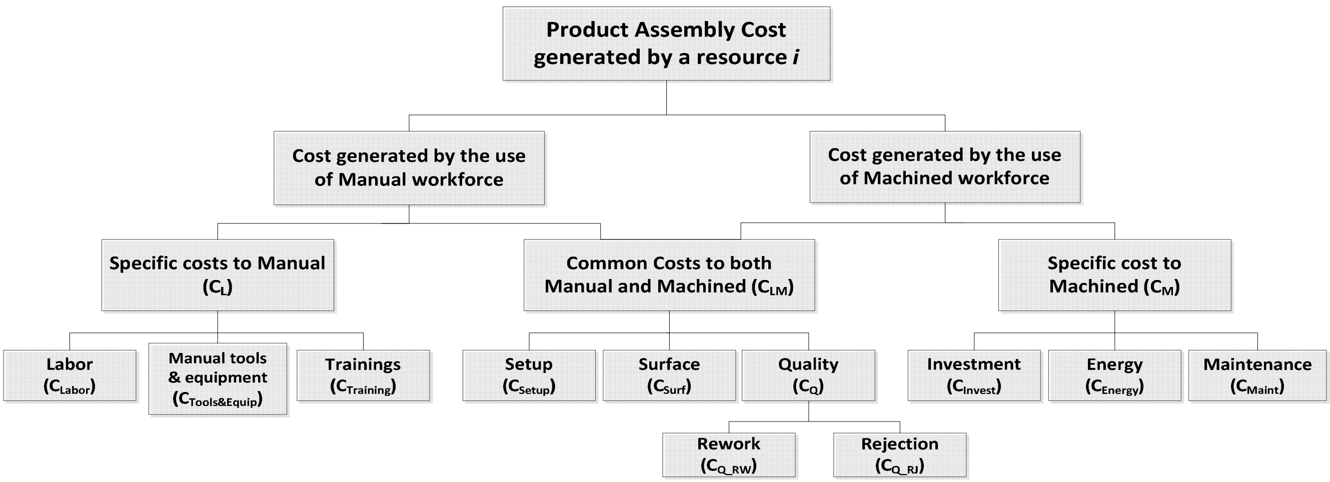


Figure 21: The assembly cost generated by a process resource according to its LoA: the cost model architecture

Figure 21 shows the architecture of the cost model we are proposing. This structure, in which all cost elements are organized according to the resource LoA, shows how to compute the cost generated by the use of manual workforce, by machined (or automated) workforce, or by a combination of both of them. To compute cost generated by a manual workforce, there are cost elements that characterizes this LoA and other more generic and which does not depend on this LoA. The specific elements to manual are noted (C_L) and includes labor cost (C_{Labor}), manual tools and equipment costs ($C_{Tools\&Equip}$), and workers trainings costs ($C_{Training}$). Similarly, for machined workforces, specific cost elements to the machined workforce and others common to all LoA exist. The specific elements to machined LoAs are noted (C_M) and includes Machines investments costs (C_{Invest}), energy costs (C_{Energy}), and maintenance costs (C_{Maint}). The common or identically computed costs for both manual and machined have to be obviously considered for manual and machined costs computation. These costs labelled C_{LM}, as abbreviation of common Cost elements to Labor and Machined, include setup costs (C_{Setup}), surface cost (C_{Surf}), and quality cost (C_Q) with rework (C_{Q_RW}) and rejections costs (C_{Q_RJ}). In the case of integrated or collaborative human-machine resource, such as manual with automated assistance LoA, cost elements considered the 3 categories (C_L, C_M, and C_{LM}) are concerned as shown in Figure 21 and equation 2. The different cost elements are detailed in next section with equations allowing their computation.

The scheme of Figure 21 provides the concerned cost categories for a selected resource i according to its LoA. For a full manual resource, the user has to consider the bloc "Cost generated by the use of manual workforce" linked to C_L and C_{LM}. For a full automatic or robotic resource, the estimator should consider the bloc "Cost generated by the use of machined workforce" linked to C_M and C_{LM}. For a combined collaborative manual-machine resource (as the "manual with automated assistance" LoA in our approach), both of the two blocs have to be considered to include the contribution of both the manual and machine use in such a combined resource. In this case all cost elements have to be considered.

As previously mentioned, we distinguish cost elements specific to manual, specific to machined, and other common costs. For the automation decision issue, the estimator can then focus on specific cost elements rather than on the common ones. In fact, these costs are the most significant for the different LoAs and can make an automation option, particularly based on these cost elements, more or less profitable because represent the major part of the generated cost. However, even common cost may make LoAs more or less profitable in some particular cases. For example, a rejection rate (a common cost) may be more or less important according to an assembly machine (LoA= 3 for example) type than other manual resource (LoA=1). In this case a common cost may make the difference. In fact, we mean by "common" the commonality from computation way point of view.

We detail in next sub-sections the computation equations of the different cost categories $\mathbf{C_L}$, $\mathbf{C_M}$, and $\mathbf{C_{LM}}$. These equations concern a selected unique resource R of the process. Input parameters which depend on the studied resource itself are used with an index "*R*" (for example: resource investment cost $ResInvestCost_R$). Other general information, which does not necessary depend on the resource itself, such as ones related to the planned production information, do not use this index.

6.6.1. Costs characterizing manual resources

The costs defined in this section are specific to manual resources.

This category includes labor cost (C_{Labor}), manual tools cost ($C_{Tools\&Equip}$), and workers trainings costs (C_{TR}). These costs are parts of the assembly costs per product. The total labor cost (C_L) is then given by the sum of the 3 mentioned elements as follows:

$$\mathbf{C_L = C_{Labor} + C_{Tools\&Equip} + C_{Training}} \quad \textbf{(4)}$$

The 3 cost elements are separately detailed in the following sections.

6.6.1.1. Manual labor cost (C_{Labor})

It is widely admitted that the manual labor cost for executing operations can be computed as the labor rate multiplied by the time to execute concerned operations (Swift & Booker, 2013) (Boothroyd, Dewhurst, & Knight, Product Design for Manufacture and Assembly – third edition, 2011) (Ostwald, 1988) (Boothroyd & Dewhurst, 1983). The labor cost is then expressed as:

$$\mathbf{C_{Labor} = C_{Rate_R} * T_R} \quad \textbf{(5)}$$

Where:

- $\boldsymbol{T_R}$: the estimated time the resource R is able to execute the assigned operations [sec] ***(P1)***
- $\boldsymbol{C_{Rate_R}}$: the labor cost rate of resource R [€/sec] ***(P2)***

The total estimated time (T_R) to execute operations assigned to a resource R can be estimated as the sum of assigned tasks (chapter 5) in case of mono-task resources. In case of more complex architectures (parallel or choice sequences of tasks), the maximum among tasks should be taken (chapter 5). We consider here the simple case of sequential tasks. The estimation of resource time (T_R) is given in this case by:

$$\mathbf{T_R = \sum_{k \in S_R} T_k} \quad (\mathbf{6})$$

Where:

- $\boldsymbol{T_k}$: the estimated time of task *k* [sec] ***(P3)***
- $\boldsymbol{S_R}$: the set of assigned tasks to resource *R* ***(P4)***

The labor cost rate of the resource R (C_{Rate_R}) consists in a secondly cost rate of using a worker to perform assembly tasks. Conventionally, the cost rate can be obtained as detailed in equation (7) based on the literature (Shehab & Abdalla, 2002) (Swift & Booker, 2013).

$$\boldsymbol{C_{Rate_R}} = \frac{\mathbf{AnnualLaborCost1W_R}}{\mathbf{YearlyWrkDur1W1S}} \quad \textbf{(7)}$$

Where:

- ***AnnualLaborCost1W$_R$***: the total cost [€] of one worker for resource R, including all charges and taxes. This quantity is significantly dependent on the manufacturing geographical location, country, etc... It can influence considerably process automation or the location decision where to install the manufacture. ***(P5)***
- ***YearlyWrkDur1W1S***: the yearly working duration in [sec] of 1 worker on 1 shift. ***(P6)***

The yearly working duration quantity can be computed as defined in (Swift & Booker, 2013) as :

$$\mathbf{YearlyWrkDur1W1S = (YearlyWrkD - StatHolid)*DailyWrkH1S*3600} \quad \textbf{(8)}$$

Where:

- ***YearlyWrkD***: the global number of working days per year for the concerned manufacturer considering the number of working days per week without subtraction of yearly holidays and leave days ***(P7)***
- ***StatHolid***: the statutory yearly holidays and leave days according to the manufacturer and location ***(P8)***
- ***DailyWrkH1S***: the daily working hours for 1 worker for 1 shift ***(P9)***
- ***"3600"***: constant used to convert [Hours] to [sec]

6.6.1.2. Manual tools and equipment

This category includes all furniture that a worker may need in order to be productive and capable of performing corresponding tasks according to their types, nature, or specifications. This includes the manual station itself, equipment, and fully manual tools and facilities. For this manual LoA, manual tools are tools necessitating only the worker's physical strength without any automated assistance (manual screw drivers, hammers, desks, etc..).

To compute the corresponding cost to this element, we use an analogy of the equation of (Gorlach & Wessel, 2008) dedicated to machines amortization costs in which the total investments cost is divided by the volume to be produced during a period of depreciation. As completely manual tools are generally reliable and rarely renovated during the production life, the period of depreciation can be confused with the production life. The equation we propose to compute manual tools and equipment is then simply as follows:

$$\mathbf{C_{Tools\&Equip}} = \frac{\mathbf{Dupli_R * Tools\&EquipInvestCost_R}}{\mathbf{TotalProdVolume}} \quad \textbf{(9)}$$

Where:

- ***ToolsEquipInvestCost$_R$***: the manual tools and station equipment cost for resource R [€] ***(P10)***
- ***TotalProdVolume***: the total planned production volume ***(P11)***
- ***Dupli$_R$***: the number of identical parallel stations of resource R duplicated to reach the required productivity ***(P12)***

The number of duplication ($Dupli_R$) is defined as the number of identical parallel resources of type (R) according to the tasks assignment (Salmi A. , David, Blanco, & Summers, 2015). The duplication allows to exactly reaching the required productivity according to the required takt_time and the estimated time for the assigned tasks and selected LoA. The way we propose to compute this duplication number is given by:

$$\mathbf{Dupli_R} = \mathbf{trunc}\left(\frac{\mathbf{T_R}}{\mathbf{T_{Req\_TaktTime}}}\right) \quad \textbf{(10)}$$

Where:

- $\boldsymbol{T_{Req\_TaktTime}}$: The required production takt-time for the planned production corresponding to the required production cadence ***(P13)***
- $\boldsymbol{T_R}$: see (P1)

The required takt-time ($T_{Req\_TaktTime}$) can be already known by the manufacturer during the automation decision phase. In other cases, if the value is unknown, we propose its computation, as proposed in equation (11), as the yearly effective working duration divided by the annual volume:

$$T_{Req\_TaktTime} = \frac{YearlyWrkDur1W1S * NbShiftsDay}{YearlyVolume} \quad (11)$$

Where:

- ***NbShiftsDay***: The number of working shifts per day ***(P14)***
- ***YearlyVolume:*** The yearly volume ***(P15)***
- ***YearlyWrkDur1W1S***: see (P6, 8)

The yearly volume for a uniform production through an entire year production can be computed as:

$$YearlyVolume = MonthlyVolume * YearlyWrkMonths \quad (12)$$

Where:

- ***YearlyWrkMonths***: The number of working months per year for the given manufacturer ***(P16)***
- ***MonthlyVolume*****:** The monthly volume ***(P17)***

The monthly volume can be calculated for a uniform production as:

$$MonthlyVolume = \frac{TotalProdVolume}{ProdLife} \quad (13)$$

Where:

- ***ProdLife***: the planned production life in months ***(P18)***
- ***TotalProdVolume***: see (P11)

6.6.1.3. Trainings costs

This category includes the costs of the needed education and trainings that can be necessary for worker to be completely productive and efficient. The cost of such trainings is associated to tasks types or nature and consists in the cost of trainings needed to master given tasks according to their complexity and characteristics. These trainings costs can be significant and we propose to take them into account in the cost per product by the cost model. In the equation we propose we consider a mean value for the worker average during which a worker is generally recruited before leaving or resigning. This period, expressed in months, will be converted to the corresponding volume according to the planned monthly volume. The equation we propose is then given by:

$$C_{Training} = \frac{TotalTrainingsCosts1W_R}{(\min\{ProdLife,\ AverageRecruitPeriod1W_R\}) * MonthlyVolume} \quad (14)$$

Where:

- ***TotalTrainingCosts1W***$_R$: the total trainings costs for a worker during his recruitment period ***(P19)***
- ***AverageRecruitPeriod1W***$_R$: the average recruitment period of a worker in months ***(P20)***
- ***MonthlyVolume***: see (P17, 13); ***ProdLife***: see (P18)

6.6.2. Costs characterizing machined resources

This category includes the resource investment cost amortization (**C_{Invest}**) and the energy cost (**C_{Energy}**) consideration in the assembly cost per product specific to machined resources. This machined cost (C_M) is then given by the sum of these 2 elements:

$$C_M = C_{Invest} + C_{Energy} \quad (15)$$

We detail the computation of these cost elements in the following sub-sections.

6.6.2.1. Resource investment costs

The resource investment costs are generally the most significant for machined resources: automatic assistance, automatic, or robotic resources. For us, the resource investment cost includes the resource purchase, transport, and installation fees. It should concern the initial investment, but also the resource renovation investments if required during the production life as well by the consideration of the resource estimated working life. For this amortization cost computation, the model of (Gorlach & Wessel, 2008) can be useful. Unfortunately, it does not consider renovations costs. To have a complete equation including renovations costs, we focus on the model of (Windmark, Gabrielson, Andersson, & StŒhl, 2012) in which the number of resource renovation is computed as the truncation of the quotient of the total number of shifts divided by the number of shifts between renovations. The number of shifts between renovations should be related to the estimated resource working life or to the manufacturer strategy. Expressing the working life by the number of working shifts as defined in (Windmark, Gabrielson, Andersson, & StŒhl, 2012) is for us a little confusing and inaccurate.

We propose to express resources working life in a time scale as an estimated number of working hours. As we propose a time based approach where each task can be time estimated, this can help to compute the number of required renovations. This can be performed considering the volume that the resource is able to produce according to the number of duplications to reach the required productivity, and to the resource working life expressed in number of working hours. We propose then the following equation to compute the number of required renovations:

$$\mathbf{Ren_R} = \mathbf{trunc}\left(\frac{\mathbf{T_R} * \frac{\mathbf{TotalProdVolume}}{\mathbf{Dupli_R}}}{\mathbf{ResWorkingLife_R} * \mathbf{3600}}\right) \qquad \textbf{(16)}$$

Where:

- $\boldsymbol{Ren_R}$: the number of renovation required for resource Ri during the production life ***(P21)***
- $\boldsymbol{ResWorkingLife_R}$: the estimated working life [Hours] of resource Ri ***(P22)***
- ***"3600"***: constant used to convert [Hours] to [sec]
- $\boldsymbol{T_R}$***: see (P1, 6);*** ***TotalProdVolume***: see (P11), $\boldsymbol{Dupli_R}$: see (P12, 10)

Once the number of renovations available, as a direct input or computed using equation (16), the total investment costs, including initial and renovation investments, can be computed using the following equation:

$$\mathbf{C_{Invest}} = \frac{\mathbf{Ren_R} * \mathbf{Dupli_R} * \mathbf{ResInvestCost_R}}{\mathbf{TotalProdVolume}} \qquad \textbf{(17)}$$

Where:

- ***ResInvestCost***$_R$: the resource investment cost including purchase cost, transport, and install ***(P23)***
- ***TotalProdVolume***: see (P11); $\boldsymbol{Ren_R}$: see (P21, 16); $\boldsymbol{Dupli_R}$**:** see (P12, 10)

6.6.2.2. Resource energy cost

In this category the cost caused by energy consumption of running machined resources is considered.

The generated cost can be significant and may have an important impact on the cost per product, especially for highly machined or automated manufactories. Thus, it can make manual solutions more profitable and more advantageous especially if the cost criterion has relevant consideration on the decision. Consequently, the energy cost integration can have a crucial importance.

The energy cost is generally computed as the consumed energy during a period of time according to the resource power. The energy consumption is defined as the resource power multiplied by the running time of the resource (Bornschlegl, Kreitlein, Bregulla, & Franke, 2015). This way is the simplest one with assumption of neglected eventual penalties that may be caused by machines inefficiencies and generated reactive energy. These machines energy behavior are difficult to obtain during the early phase. We also think their cost when amortized on the volume will be insignificant and can be neglected. Yet, this assumption can be compensated if the estimator thinks relevant to be integrated, as an extra cost consideration of an additional percentage to the final energy cost, of 20% for example.

In our time-based cost approach, as previously mentioned, all operations of the process are time estimated, and consequently resources execution time can be predicted ($\boldsymbol{T_{Ri}}$ for the obtained value). The energy cost for a given automated resource of a power $\boldsymbol{Power_R}$ executing assigned assembly operations can be defined by (Bornschlegl, Kreitlein, Bregulla, & Franke, 2015) formula. The equation we propose in our context to compute resources energy using presented parameters is then as follows:

$$\mathbf{C_{Energy}} = \frac{\mathbf{Power_R * C_{KWH}}}{\mathbf{3600}} * \mathbf{T_R} \qquad \textbf{(18)}$$

Where:

- $\boldsymbol{Power_R}$: the power consumption of resource Ri [KW] ***(P24)***
- $\boldsymbol{C_{KWH}}$: the cost of 1KWH ***(P25)***
- ***"3600"***: constant used to convert [€/Hour] to [€/sec];
- $\boldsymbol{T_R}$: see (P1, 6)

6.6.2.3. Maintenance cost

Our expression of the maintenance cost is based on the equation of (Bornschlegl, Kreitlein, Bregulla, & Franke, 2015). In this equation, the authors consider a total maintenance cost including 2 elements. The first element contains the labor cost during the maintenance and the time required to repair the failure. This time quantity is multiplied by the labor rate of the maintenance staff to compute the resulting cost. A multiplication by a failure probability is also performed to generalize the cost on the assembled products. The second element of (Bornschlegl, Kreitlein, Bregulla, & Franke, 2015) equation contains the spare parts cost as the sum of maintenance spare parts costs during a horizon of time.

We use this equation as a basis to predict the maintenance cost with some vocabulary and equations adaptations to be efficiently used in our context.

First, we use in our proposal for 'time required to repair a failure' parameter a maintenance probabilistic parameter called MTTR as the "Mean Time To Repair".

We use a second probabilistic maintenance parameter: the MTBF as the "Mean Time Between Failures" to be expressed in number of hours working. This quantity, when divided by the time to produce a product by the given resource, provides the volumes produced between each consecutive failures. We amortize then each maintenance repair cost element on the produced quantity between each two consecutive failures. (15).

In addition to these adaptations, we propose an enhancement by adding, as for setup downtime equations of (Jönsson, Andersson, & Ståhl, 2011) and (Windmark, Gabrielson, Andersson, & StŒhl, 2012), the cost of stopping production per second. This quantity is then added to the labor cost rate during the downtime in the previously defined first element, also expressed as a secondly cost during the downtime. In fact, according to our discussions with manufacturers, a stop of production during few minutes can be heavily costly and can cause thousands of euros lost for short production stoppage. Technological and automation choices can cause such failures and generate significant corresponding costs. This should be then taken into account in the cost model.

Concerning the spare parts cost amortization representing the second member of the maintenance equation of (Bornschlegl, Kreitlein, Bregulla, & Franke, 2015), we project this equation on an annual scale rather than on a total value to be estimated for a whole planned production horizon less meaningful in our context. In fact, yearly estimates of the planned production can be predictable since the early phase even approximately based on the manufacturer experience considering the selected resources LoAs and the machines initial purchase cost. Consequently, a division of annual maintenance spare parts cost by the annual volume is needed.

The obtained equation we propose, with also 2 cost members, is then given by:

$$\mathbf{C_{Maint}} = \frac{\mathbf{MTTR_R} * \left(C_{RateMaint_R} + \frac{\mathbf{HourlyStopProdCost}}{\mathbf{3600}}\right) * (\%\mathbf{Failure_R})}{\frac{\mathbf{MTBF_R}}{\mathbf{T_R}}} + \frac{\mathbf{C_{AnnualSP_R}}}{\mathbf{YearlyVolume}} \quad \mathbf{(19)}$$

Where:

- $MTTR_R$*:* the mean time to repair a failure for resource R: an average estimation time [Hours] to repair a failure including the time to detect the fault, repair, and regain production ***(P26)***
- $MTBF_R$*:* the mean time between failures for a resource R: an average estimation time [Hours] separating 2 consecutive stops caused by resource failures ***(P27)***
- $C_{RateMaint_R}$: the maintenance cost rate of a labor assigned to repair defects on resource R ***(P28)***
- $\%Failure_R$: The failure rate associated to resource R ***(P29)***
- $\mathbf{C_{AnnualSP_R}}$**:**The estimated annual spare parts costs for maintenance of resource R ***(P30)***
- ***HourlyStopProdCost***: the hourly cost of downtime ***(P31)***
- ***"3600"***: constant used to convert Hours to Seconds
- T_R: see (P1, 6); ***YearlyVolume:*** see (P15, 12);

6.6.3. Common costs to manual and machined

These costs are common to manual and machined resources. They are generated by the existence of the station itself whatever its type or LoA. But, the LoA can have an impact on them, their magnitude, or frequency. For example, setup cost, as a common cost, can be frequent or long for automated (resource configuration) but rare or short for manual station (station preparation).

We distinguish in this category of common costs 3 elements:

- Set-up cost (C_{Setup})
- Surface cost (C_{Surf})
- Quality cost (C_Q) including rework (C_{Q_RW}) and rejection (C_{Q_RJ}) costs

These cost elements computation are detailed in the following subsections.

6.6.3.1. Set-up cost

The set up cost is defined in (Son, 1991) as the cost of preparing machines for each production run. This cost is generally computed with consideration of the batch size or the number of products to manufacture in downstream the set up (Jönsson, Andersson, & Ståhl, 2011) (Windmark, Gabrielson, Andersson, & StŒhl, 2012) (Jha, 1996) (Jung, 2002) (H'mida, Martin, & Vernadat, 2006).

The setup cost concerns for us all types of resources and stations, and then all selectable resources LoAs. For manual LoAs, it consists in the station preparation, parts feeding, and organizing the tools and the next production batch. For automatic dedicated machines, the setup consists in machines configuration and setting. For robotic LoA, the setup can consist in reconfiguring or reprogramming robots before moving to the next production batch.

The setup cost equation we propose is based on the equation of (Jönsson, Andersson, & Ståhl, 2011) and (Windmark, Gabrielson, Andersson, & StŒhl, 2012). The equation considers the hourly cost of machines during downtimes with regard to the batch size, multiplied by the setup duration. We add to this formula the consideration of the labor cost of the operator or programmer mobilized to perform the setup operation. The obtained complete equation is then as follows:

$$\mathbf{C_{Setup}} = \frac{\frac{\mathbf{HourlyStopProdCost}}{\mathbf{3600}} + \mathbf{C_{RateSetup_R}}}{\mathbf{BatchSize_R}} * \mathbf{T_{Setup_R}} \quad \textbf{(20)}$$

Where:

- $\boldsymbol{C_{RateSetup_R}}$: the setup cost rate of a labor assigned to configure resource R during the setup period ***(P32)***
- $\boldsymbol{BatchSize_R}$: The volume that should be launched on resource R after being set up ***(P33)***
- $\boldsymbol{T_{Setup_R}}$: The setup average time according to resource R LoA and assigned tasks types to it ***(P34)***
- ***"3600"***: constant used to convert hourly cost to secondly cost
- ***HourlyStopProdCost***: see (P31)

6.6.3.2. Surface cost

This cost element represents the cost of the surface that the concerned resource is estimated to occupy in the manufactory. In the literature, major researches consider this cost as one of the overheads (Bernet, Wakeman, Bourban, & Månson, 2002). Few researches detail the computation equation of this cost.

In (Son, 1991), a formula is defined proposing to compute the manufacturing space cost as the cost of the floor space per square foot multiplied by the resource floor space. In some researches, particularly for late phase estimations, this quantity is divided by production duration to have a cost per time unit (Windmark, Gabrielson, Andersson, & StŒhl, 2012). For us, as we seek for a cost per product computation, we adapt the formula of (Son, 1991) by a computation of a monthly surface cost and a division by the monthly volume instead of the time horizon. This leads to obtain a cost per product rather than a cost per second. The equation we propose is then given by:

$$\mathbf{C_{Surf}} = \frac{\mathbf{Surface_R} * \mathbf{C_{MonthlySqrMeter}}}{\mathbf{MonthlyVolume}} \quad \textbf{(21)}$$

Where:

- $\boldsymbol{Surface_R}$: the occupied surface by the resource R ***(P35)***
- $\boldsymbol{C_{MonthlySqrMeter}}$: cost of a monthly square meter surface according to the factory geographical location, country, or city ***(P36)***
- ***MonthlyVolume***: see (P17, 13)

6.6.3.3. Quality cost

For the quality cost, we distinguish 2 separate elements:

- The rework cost ($\mathbf{C_{Q\_RW}}$)
- The rejection cost ($\mathbf{C_{Q\_RJ}}$)

The quality cost is then defined as the sum of the 2 elements the rework and rejection costs as follows:

$$\mathbf{C_Q} = \mathbf{C_{Q\_RW}} + \mathbf{C_{Q\_RJ}} \quad \textbf{(22)}$$

In the following sub-section the computation of the 2 elements is separately detailed.

6.6.3.3.1. Rework cost

The rework of non-conform assembled product depends on the manufacturer organization and strategy of non-conformities management. 2 cases may exist:

- Case A. A second pass through the given resource for rework
- Case B. A dedicated station for rework

We detail independently case A and B.

- ***Case A. A second pass to rework non-conformities***

In this case, non-conform products are reworked on the resource R that caused the non-conformity.

In (Bernet, Wakeman, Bourban, & Månson, 2002), to compute this reasoning, a non-conformities percentage factor F_{rew} is defined reflecting unsatisfactory assembly quality ratio. All non-conform parts are considered as additional new parts to go another time through the process. To do so, the cost of owning and running a piece of equipment during the period of time defined by t_{run} is multiplied by $(1 + F_{rew})$ to take into account the time spent to assemble and rework defective parts considered as virtually new parts. To adapt this reasoning to our approach, as our model is a model by resource, this non-conformity percentage depends on the given resource. We label this factor as "$\text{Non\_Conform}_\text{R}$". This non-conformity can depend on the resource LoA and on the assigned tasks types and complexity. In some cases, if the given resource or its associated tasks natures should not cause non-conformities or are not included in the rework process, this percentage can be assumed to be null.

The percentage of non-conformities includes for us assemblies that can be reworked and others that cannot and that should be rejected, labelled "Reject_R". As our interest here is to compute the rework cost, we do not consider the rejected parts that have to be then subtracted. The cost generated by non-conformities rejection will be treated separately in next section 6.6.3.3.2. Among the different cost elements, the involved ones for which a cost is wasted in producing non-conform parts involve time-based cost in which time should be again spent to rework non-conformities. Regarding all presented cost elements, the concerned cost elements correspond then, according to the resource LoA, to labor cost for manual, and to energy cost for machined resources. The rework cost is then given by:

- For manual resources:

$$\mathbf{C_{Q\_RW}} = \mathbf{C_{Labor_R}} * (\%\mathbf{Non\_Conform_R} - \%\mathbf{Reject_R}) \quad \textbf{(23)}$$

- For machined resources:

$$\mathbf{C_{Q\_RW}} = \mathbf{C_{Energy_R}} * (\%\mathbf{Non\_Conform_R} - \%\mathbf{Reject_R}) \quad \textbf{(24)}$$

Where:

- ***%Non_Conform***$_{\boldsymbol{R}}$: the estimated percentage of non-conformities with regard to the whole ***(P37)***
- ***%Reject***$_{\boldsymbol{R}}$: the percentage of rejects with regard to the whole volume ***(P38)***
- $\boldsymbol{C_{Labor_R}}$: see (5) ; $\boldsymbol{C_{Energy_R}}$: see (18)

- ***Case B. A dedicated rework station***

In this case, a dedicated resource, manual or automated, is tailored to perform rework operations. From cost computation point of view, a rework station should be treated as any resource of the process according to its LoA and should be then already modelled as resource. The previously presented equations of the model are applicable to that resource to compute the cost generated by this rework station. The obtained cost from such resource will contribute to rework cost category.

Consequently, the cost equations to be used depend on the modelled rework resource LoA. The rework cost should be computed using equation (1) for a manual rework stations; equation (2) for a manual with automated assistance rework resource, and equation (3) for machined resources that can consist in an automatic or robotic rework station. The detailed equations can be found through the previous different sections 6.6.1 to 6.6.3.2.

6.6.3.3.2. Rejection cost

We consider in this section the cost caused by produced non-conform parts that should be rejected. These parts cannot be reworked or disassembled to make them conform as the ones treated in section 6.6.3.3.1. This cost can have important impact particularly when parts to be assembled are significantly costly from material, previous manufacturing steps, complexity, time, or rareness points of views. The parts costs should be known during the early phase of automation decision. This should not represent an issue once generally prototypes exist during that phase and parts suppliers sometimes already negotiated and decided as well.

In (Windmark, Gabrielson, Andersson, & StŒhl, 2012) and (Jönsson, Andersson, & Ståhl, 2011), the rejection rate is computed as the difference between the initial number of products and the real obtained conform products number, divided by the initial number of products to be assembled. Time spent in producing rejected parts can be then obtained (Gary Teng & Garimella, 1998). The obtained time can be consequently converted to a cost using the running cost rate according to the selected resource LoA as previously done for the non-conformities to be reworked in section 6.6.3.3.1.

In addition to the time and cost spent to produce rejected parts, we consider also the material cost of these rejections. The cost will be then considered as an amortization on unitary cost per product. For a resource characterized by a rejection percentage $\%Reject_R$, with available concerned parts costs in assembly, a multiplication by the reject percentage may provide the rejection cost caused by parts rejection.

The complete formulas of rejection cost according to the resource LoA are detailed in (25) and (26) as follows:

- For manual resources:

$$\mathbf{C_{Q\_RJ}} = (\mathbf{C_{Labor_R}} + \mathbf{C_{Parts_R}}) * \%\mathbf{Reject_R} \qquad \textbf{(25)}$$

- For machined resources:

$$\mathbf{C_{Q\_RJ}} = (\mathbf{C_{Energy_R}} + \mathbf{C_{Parts_R}}) * \%\mathbf{Reject_R} \qquad \textbf{(26)}$$

Where:

- $\boldsymbol{C_{Parts_R}}$: the cost of parts assembled on resource R that can be damaged by a resource R failure causing their reject ***(P39)***
- $\boldsymbol{C_{Labor_R}}$: see (5) ; $\boldsymbol{C_{Energy_R}}$: see (18) ; ***%Reject$_R$***: see (P38)

6.7. Cost model illustration

In this section, we illustrate the proposed cost model using a graphic bloc scheme. The purpose is to provide visibility and an easy way to implement the modular model we proposed, the different cost elements or modules computation, and to identify and classify the model inputs and outputs. The graph is shown in Figure 22.

In Figure 22, the different cost model input parameters and the different computed cost elements can be easily distinguished. Different arithmetic operators (sums, subtractions, divisions, and multiplications) and mathematical operators (truncation function) are used as shown in the figure in yellow color. These operators allow the implementation of the cost equations and the computation of the cost elements outputs connecting the concerned input parameters. These input parameters are organized in basic or essential parameters and advanced or deducible input parameters. The deducible parameters are computed from basic inputs using previously defined equations. If the advanced inputs can be available, they can replace and instantiate corresponding basic inputs.

The input parameters are classified, using the color code as shown in the figure key used in Figure 22, to planned production information and to data related to the resource itself. The planned production information (clear blue colored) is generally independent from the resource and concerns the planned production strategic information. The specific data to the resource (green colored blocs) is more dependent to the resource itself, its LoA, to the assigned tasks, their nature, and the resulting time estimates retrievable from databases by consideration of all these parameters and to product design complexity.

Concerning the output information, they are classified to 3 classes as shown in the figure using the 3 columns organization and the blocs' colors as well. These outputs are classified to: cost elements computation (grey blocs - 1st column of the outputs category), the classified outputs to specific to manual, to machined, or common costs (purple - 2nd column), and the final cost per product estimation generated by the selected resource according to its LoA (dark blue - 3rd column). The figure shows a summary or illustration of the whole cost model equations, elements, inputs, and outputs in a compact one page scheme. It provides a panoramic view on the integrality of the proposed cost model. It can also be considered as a worksheet making the use of the model easier or faster by user estimators.

6.8. Multi-resources process cost estimation

The previously presented equations concern a unique selected resource. The equations allow computing the assembly cost per product generated by the selected resource according to its assigned tasks and associated characteristics (time estimation, initial investment, energy consumption, surface, etc), and with consideration of the planned production information (volume, yearly working duration, location, etc).

To handle the total product assembly cost generated by the whole process configuration, the procedure has to be iterated on all the process resources separately. The total product cost is then the sum of the obtained costs per product per resources as previously shown in Figure 20 and described in following equation (27).

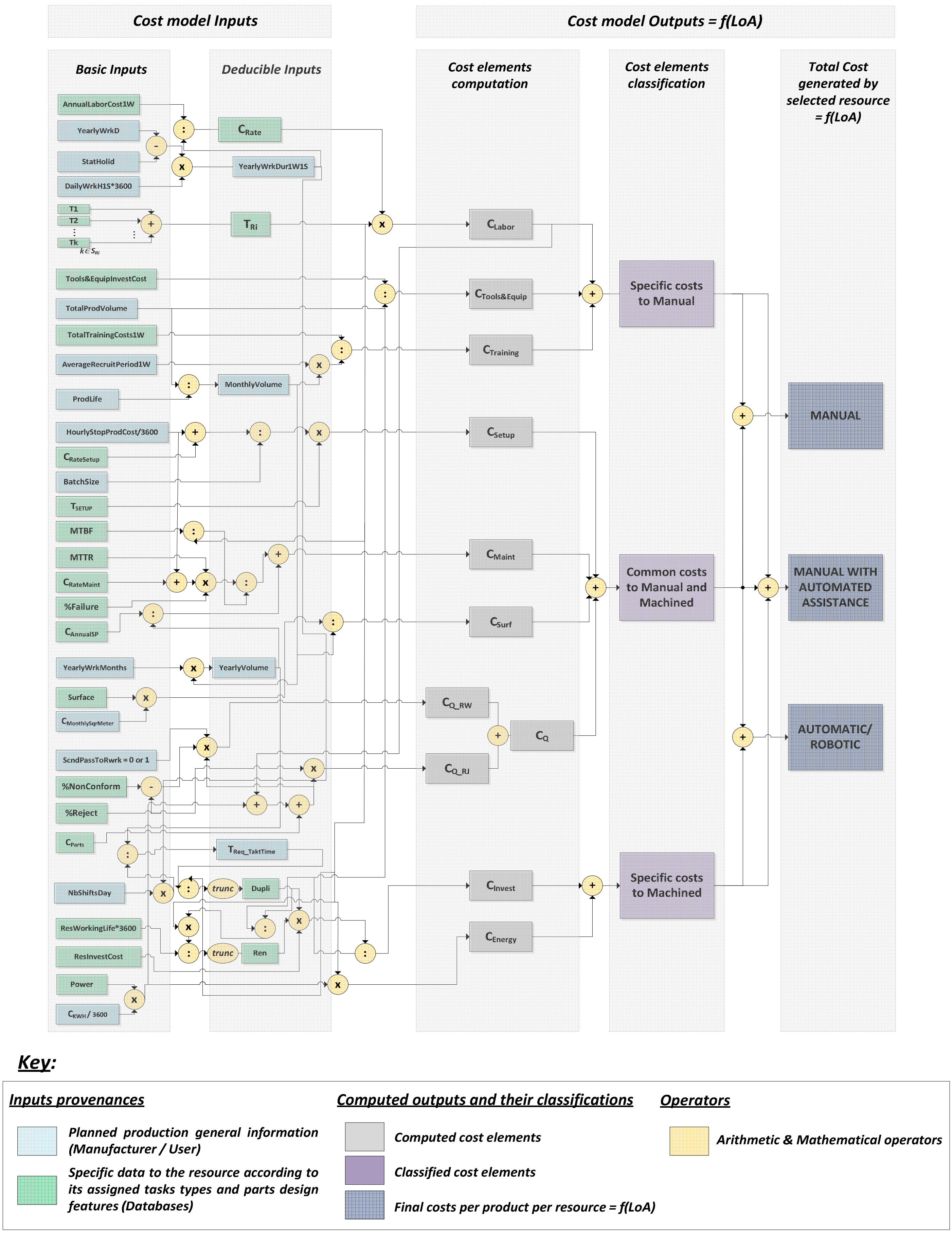


Figure 22: Illustration of the integrated cost model to support automation decision

$$\mathbf{C_{Total}} = \sum_{i=1}^{N} \mathbf{C_{Ri}} \quad (27)$$

Where:

- $\boldsymbol{N}$: the number of different resources in the process configuration ***(P40)***
- $\boldsymbol{C_{Ri}}$: The assembly cost generated by a resource Ri according to its LoA: manual (see equation (1)), manual with automated tool (see equation (2)), or automatic / robotic (see equation (3)).

This modular way to compute the cost allows handling multi-resources and hybrid processes with possibly various process resources LoAs and partial automation. This can be possible due to the use of the superposition principle using the independent focuses on the different resources with a modular reasoning to compute the cost (Figure 20).

This procedure allows to compute the assembly cost for a given assembly system configuration with tasks, assigned resources, and resources selected LoAs. The obtained cost corresponds then to one scenario of assembly system and corresponding automation choices.

An advantage of this modular cost computation consists in the possibility to identify resources with high cost compared to the total product cost. This may provide an indicator of a need to optimize such resource from tasks assignment or LoA point of view. The LoA can be high compared to what is needed and imposes extra costs. This can be detected using the proposed time and cost approach.

As another use, the proposed approach can provide a useful tool to compare different systems alternatives from cost point of view. The use of the model can support comparing different processes architectures with various tasks assignments possibilities, organizations, and automation options to be selected to the resources.

To reach the optimal configuration, the cost model with a sum over all the resources for a product cost can be associated as an objective function or as a constraint to be compared to a threshold. A generation of all possible combinations of resources LoAs and tasks assignment can be a track to search the optimal configuration. This will be tackled in chapter 7.

In next section, cost performance indicators are presented to enhance a configuration evaluating and facilitate alternatives comparisons exploiting the different parameters used by the cost model, allow assessing the relevance of the solution, its suitability, economical feasibility, and long-term profitability. The different indicators are detailed in next section 6.9.

6.9. Cost-based performance indicators computation

Exploiting different cost model input parameters, the proposed cost estimation approach provides a possibility to compute other significant cost performance indicators. These cost indicators can be relevant and helpful for automation decision making. We propose the following performance indicators:

- Process Initial investment
- Process total investments through the production life
- Process total surface and cost
- Process total energy power and monthly consumption cost
- Payback period
- Return on Investment estimation

These different parameters are detailed through the different next sub-sections. We use an index *Ri* for parameters which depend on a resource *Ri*. These performances indicators concern the whole process configuration involving all the resources. Consequently, all the process resources and their selected LoAs are concerned.

6.9.1. Process initial investment

The process initial investment for the selected resources consists in the sum over the process resources initial investments. This indicator can be significant in decision making because involves the manufacturer investment potential and his capacity to be engaged in heavy investments. In fact, it can be possible to have a low cost per product on the whole planned production horizon, while, the solution can require an initial important investments, sometimes making it unfeasible for manufacturers or involves a risk that major manufacturers prefer to avoid.

The resources initial investment depends on the resources types according to assembly tasks natures to be performed (soldering, riveting, etc.) and also selected LoAs. For this indicator, our interest is only on the initial investment, the number of resources duplications should be then included. The equation we propose to compute this indicator is as follows:

$$P_{Inv} = \sum_{i=1}^{N} [\, \mathbf{Dupli_{Ri}} * (\, \mathbf{ResInvestCost}_{\boldsymbol{Ri}} + \boldsymbol{ToolsEquipInvestCost}_{\mathbf{Ri}}) \,] \tag{28}$$

Where:

$\boldsymbol{N}$: see(P40); $\boldsymbol{Dupli_{Ri}}$:see(P12, 10); $\boldsymbol{ResInvestCost_{Ri}}$:see(P23); $\boldsymbol{ToolsEquipInvestCost}_{\mathbf{Ri}}$:see(P10)

6.9.2. Process total investments through production life

Compared to the previously presented process initial investment (P_{Inv}), the difference is that the total investment through the production life (P_{T_Inv}) includes the renovation number for machined resource of the process. The indicator (P_{T_Inv}) will help on predicting how much the given process configuration will require in total investment. This can also provide an idea to the manufacturer about the necessity to renovate resources according to the selected LoAs and the need to invest again during the production life, which can be sometimes discouraging for deciders with regard to some LoAs. The formula we propose to compute P_{T_Inv} is given by:

$$P_{T\_Inv} = \sum_{i=1}^{N} [\, \mathbf{Dupli_{Ri}} * (\, \mathbf{Ren_{Ri}} * \mathbf{ResInvestCost_{Ri}} + \boldsymbol{ToolsEquipInvestCost}_{\mathbf{Ri}}] \tag{29}$$

Where:

- $\boldsymbol{N}$: see (P40); $\boldsymbol{Ren_{Ri}}$: see (P21, 16) ; $\boldsymbol{Dupli_{Ri}}$: see (P12, 10) ; $\boldsymbol{ResInvestCost_{Ri}}$: see (P23) ; $\boldsymbol{ToolsEquipInvestCost}_{\mathbf{Ri}}$: see (P10)

6.9.3. Process total surface and monthly cost

The data used to compute the surface cost, for which a resource surface depends on the assigned tasks nature and to the resource LoA, can allow obtaining first the total process surface. This estimation can provide an order of magnitude of the process occupying surface. Even if the factory may exist, such estimation allows checking the feasibility to install the given process configuration, by the total process surface estimation and comparison to the available space, especially when the available space in the manufactory is limited. The equation involves then all resources surfaces with consideration of their computed duplication number. The obtained equation is as follows:

$$P_{Surf} = \sum_{i=1}^{N} [\, \mathbf{Dupli_{Ri}} * \mathbf{Surface_{Ri}}] \qquad (\mathbf{30})$$

Where:

- $\boldsymbol{N}$: see (P40)**;** $\boldsymbol{Dupli_{Ri}}$: see (P12, 10), $\boldsymbol{Surface_{Ri}}$*:* see (P35)

Once the total process surface obtained, a projection to a surface monthly cost can be estimated by a multiplication by the square meter monthly cost according to the manufacturing geographical location. The equation is defined in equation (31)

$$P_{Surf\_MonthlyCost} = P_{Surf} * C_{MonthlySqrMeter} = \left[\sum_{i=1}^{N} [\, Dupli_{Ri} * Surface_{Ri}]\right] * C_{MonthlySqrMeter} \qquad (\mathbf{31})$$

Where:

- $\boldsymbol{N}$: see(P40)*;*$\boldsymbol{P_{Surf}}$: see (30)*;*$\boldsymbol{C_{MonthlySqrMeter}}$: see (P36); $\boldsymbol{Dupli_{Ri}}$: see(P12,10); $\boldsymbol{Surface_{Ri}}$: see (P35)

6.9.4. Process total power and monthly cost

The energy analyses can be important to LoA decision. The total process power can be relevant if the factory and electric installation exist. It can provide the possibility to evaluate if existing power installation can handle obtained total process power and energy, particularly for highly automated configurations. The equation allowing computing the total power is detailed in equation (32).

$$P_{Pow} = \sum_{i=1}^{N} [\, \mathbf{Dupli_{Ri}} * \mathbf{Power_{Ri}}] \qquad (\mathbf{32})$$

Where:

- $\boldsymbol{N}$: see (P40)**;** $\boldsymbol{Dupli_{Ri}}$: see (P12, 10) *;* $\boldsymbol{Power_{Ri}}$: see (P24)

A second proposed energy indicator consists in the process monthly energy consumption cost. The prediction of this running cost may have importance for manufacturers. The proposed equation to estimate the monthly energy cost uses the previously defined equations computing the energy cost per resource per product (C_{Energy}) with consideration of the energy spent on rework. This quantity is simply multiplied by the monthly volume to obtain an estimation of the monthly energy cost of the process. The obtained equation is then given by:

$$P_{EnergyMonthlyCost} = \left[\sum_{i=1}^{N} \left[\frac{T_{Ri} * Power_{Ri}}{3600} * (1 + (\%Non\_Conform_R - \%Reject_R))\right] * C_{KWH}\right] * MonthlyVolume \qquad (\mathbf{33})$$

Where:

- $\boldsymbol{N}$: see (P40); ***MonthlyVolume***: see (P17, 13); $\boldsymbol{T_{Ri}}$: see (P1, 6)**;** $\boldsymbol{C_{Energy_{Ri}}}$**: see (18)***;* $\boldsymbol{Power_{Ri}}$: see (P24); $\mathbf{C_{KWH}}$: see (P25); %$\boldsymbol{Non\_Conform_R}$: see (P37); %$\boldsymbol{Reject_R}$: see (P38)
- ***"3600"***: constant used to convert seconds to hours

6.9.5. Payback period

The payback period can represent important or decisive significance for manufacturers. This indicator allows evaluating the necessary period to recover the invested amount. It also indicates if the investment can be profitable, if the investment can be amortized on the planned production life period, and if yes, when exactly in years or in months can the invested amount be regained.

This indicator is widely used to be calculated, for the simple case where the value depreciation of money through the time is neglected, as the investment amount divided by the yearly or monthly income profit. The income profit unity is generally expressed in [€/month] or [€/Year]; and will provide the $\mathrm{T_{Payback}}$ unity. For example, if the income profit is in [€/Year], $\mathrm{T_{Payback}}$ will be in [Years].

In our context, as the income profit in a large sense is not obtainable because of the early phase decision and due to the non-existence yet of a real production and of products sale, we use this indicator to compare a given current solution (SolCur) to another reference one (SolRef); generally a solution with a minimum of investment or a previous well known solution based on historical data, previous projects, or existing equivalent projects to be improved by a redesign with automation possibilities analyses. We use then a relative profit of a solution with regard to another one. Both solutions, SolCur and SolRef, have associated total investments including renovations, respectively $\mathrm{P_{T\_Inv\_SolCur}}$ and $\mathrm{P_{T\_Inv\_SolRef}}$ (see equation (29)), and costs per product, respectively $\mathrm{C_{Total\_SolCur}}$ and $\mathrm{C_{Total\_SolRef}}$; (see equation (27)). The different configurations have the same planned production volume because the volume has to be imposed to the system candidate which is designed so as to reach that planned volume. The payback period of SolCur compared to SolRef consists consequently to extra investment divided by the monthly or yearly profit of SolCur with regard to SolRef. This profit can be then expressed as a profit per product multiplied by the monthly or yearly volume. To obtain a $\mathrm{T_{Payback}}$ in months, the profit has to be then expressed as a monthly profit, and consequently, the volume has to be expressed as a monthly volume.

The resulting equation allowing computing the payback period $\mathrm{T_{Payback}}$ of current solution candidate SolCur with regard to a reference solution SolRef is given by:

$$\mathbf{T_{Payback}} = \frac{\mathbf{P_{T\_Inv\_SolCur}} - \mathbf{P_{T\_Inv\_SolRef}}}{(\mathbf{C_{Total\_SolRef}} - \mathbf{C_{Total\_SolCur}}) * \mathbf{MonthlyVolume}} \quad (34)$$

Where:

- $\boldsymbol{P_{T\_Inv\_SolCur}}$: the total investment ($P_{T\_Inv}$, see equation (29)) of the current solution. ***(P41)***
- $\boldsymbol{P_{T\_Inv\_SolRef}}$: the total investment ($P_{T\_Inv}$, see equation (29)) of the reference solution. ***(P42)***
- $\boldsymbol{C_{Total\_SolCur}}$: the total product assembly cost (C_{Total}, see equation (27)) of the current solution. ***(P43)***
- $\boldsymbol{C_{Total\_SolRef}}$: the total product assembly cost (C_{Total}, see equation (27)) of the reference solution. ***(P44)***
- ***MonthlyVolume****:* See (P17, 13)

If the total investment of the current solution can be amortized during the production life, the payback period indicator $\boldsymbol{C_{Total\_SolRef}}$ should be inferior to the planned production life, also expressed in Months (see P15). Else, the indicator mean that the solution Sol_Cur is not profitable compared to Sol_Ref. In this case, investment on automation is economically not viable.

6.9.6. Return On Investment (ROI)

The return on investment (ROI) allows informing on the efficiency of an investment and evaluating the percentage of profit generated by the investment considering the total invested amount. Consequently, it can allow comparing the advantages or profits generated by different investment alternatives. Thus, in our context it allows comparing different assembly systems automation configurations profitability. For manufacturers, it allows comparing a process configuration solution to completely other financial options, such as bank money saving and account benefits recovery or investment in trading if they can be more lucrative than the one generated by investing in automating the process. The ROI indicator is commonly computed as the generated profit amount divided by the investment amount. The profit amount is calculable as the difference between the total generated incomes and the investment amount.

As we previously proceeded with the payback period computation of section 6.9.5, we apply this principle in the context of predicting the profitability of new systems under design not yet existing. We then similarly use this indicator to compare a given current solution (SolCur) to another reference one (SolRef). The aim is to evaluate the profit percentage with regard to the extra investment that the current solution is able to generate compared to the reference one. By contrast, as here we are evaluating the total profit percentage of the solution, the profit should be then the total profit generated by the solution through the whole production life. The profit per product should be then multiplied by the total planned production volume rather than a monthly profit as with ($T_{Payback}$). Concerning the extra investment, we use the difference between total investments required by the 2 solutions Sol_Cur and Sol_Ref. The resulting equation allowing to compute the ROI is then as follows:

$$\mathbf{ROI} = \frac{[(C_{Total\_SolRef} - C_{Total\_SolCur}) * \mathbf{TotalProdVolume}] - [P_{T\_Inv\_SolCur} - P_{T\_Inv\_SolRef}]}{P_{T\_Inv\_SolCur} - P_{T\_Inv\_SolRef}} \quad \textbf{(35)}$$

Where:

- $P_{T\_Inv\_SolCur}$: see (P41); $P_{T\_Inv\_SolRef}$: see (P42); C_{Total_SolCur}: see (P43); C_{Total_SolRef}: see (P44); ***TotalProdVolume***: see (P11)

After presentation of the proposed cost approach, model and performance indicators, we apply the whole proposal on an industrial case study. The case study is presented and analyzed in next section 6.10.

6.10. Case study

In this section, the proposed cost estimation approach is tested on a real industrial assembly case study. The concerned manufacturer is a leader painting guns supplier for diverse applications, principally for industrial automotive painting applications: manual painting guns and painting robots as well. The application concerns an assembly process of microvans. Such microvans are found within multiple variants of the produced painting systems. This component represents a crucial element for diverse references of painting guns and allows passing or blocking the painting fluid for each of the used fluid color shade (a microvan for each color shade) from the fluid tank to the gun output extremity with the right debit and dosing.

The microvan in its final assembly stage is represented in Figure 23. For confidentiality reasons, the real dimensions of the different parts are not provided through this study. Also, for the same reasons the strategic information concerning the production that will be presented is a little modified.

Figure 23: The product example: a microvan for painting guns

When performing the study, an assembly process allowing assembling the different components of the microvan already exists. The process is almost manual with some intermediate automation levels, classified as manual with automated assistance, with a use of multiple presses operated by assembly workers. The manufacturer has some questionings about the possibility to increase the process automation level in order to decrease the assembly cost with keeping or increasing the assembly quality level. In our study, we proceed by analyzing the product to be assembled, modelling the process, evaluating the cost generated by the existing process using the proposed cost model, proposing processes automation alternatives, again using the cost model to predict the cost of these alternatives, and finally providing analyses, comparisons, and feedbacks about the different processes based on the obtained results using the proposed approach.

6.10.1. Product design analysis

Because of the tightness functional requirements of the microvan, its design includes multiple seals to be assembled inside and outside the body. Multiple quality inspections and tightness tests have to be performed in the assembly process in order to verify that all the right components are appropriately included, in the correct orientation and that the final product is valid. The detailed components to be assembled are shown in Figure 24. As shown in the figure, the product includes following components: a body, a piston, a spring, a plug, a needle, a witness, and 9 seals numbered from seal1 to seal 9.

As for a given product, corresponding parts assembly can be performed in various orders or sequences; we opt for one assembly sequence, the one recommended and used by the manufacturer during our observations and study. This assembly sequence is presented in next section.

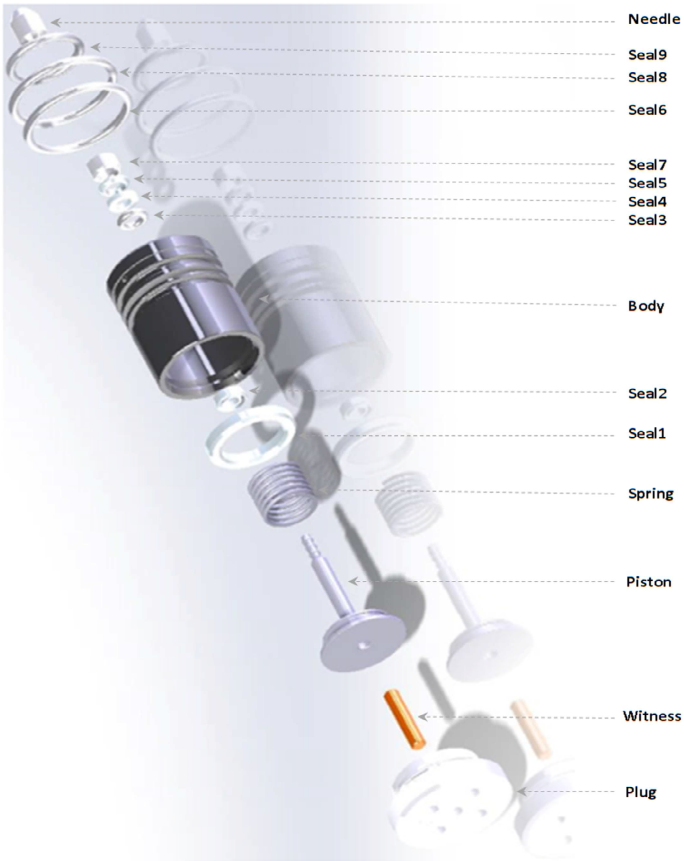


Figure 24: The product components to be assembled

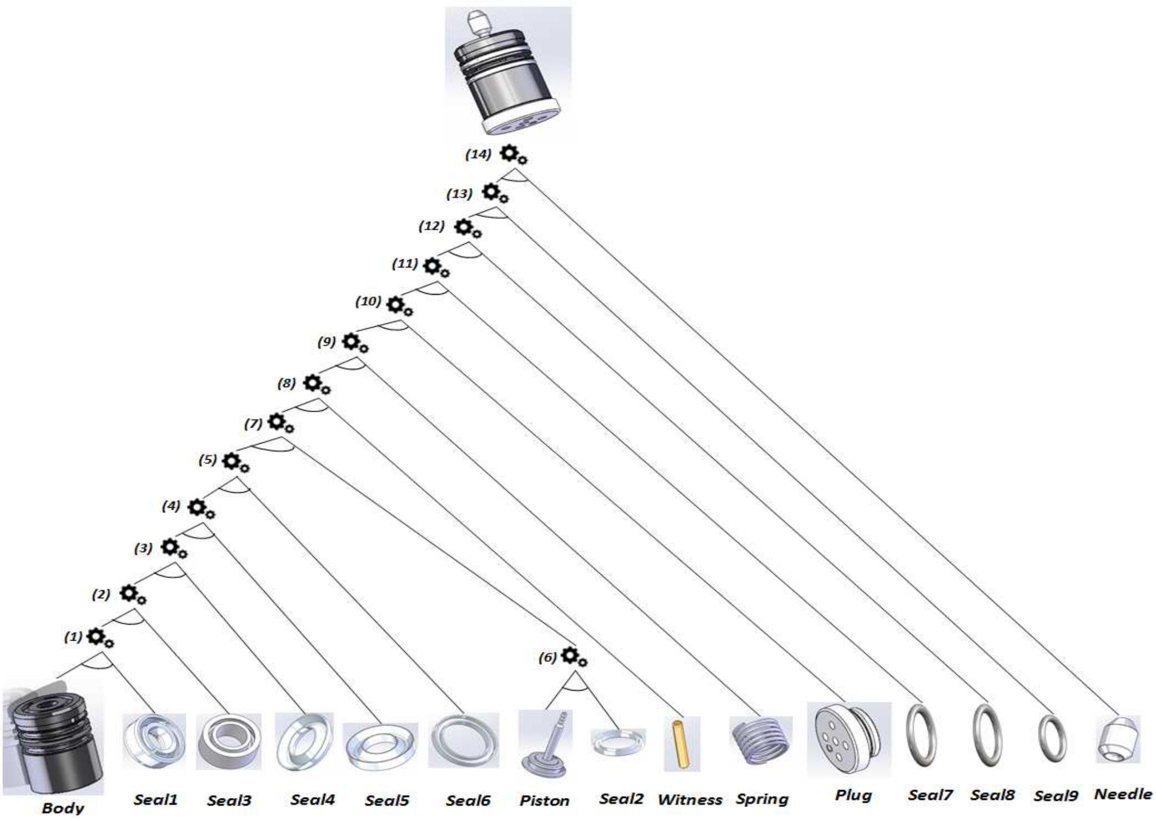


Figure 25: The product assembly sequence representation using an AND/OR graph

6.10.2. Assembly sequence

We present in this section the assembly sequence that will be used in the whole case study. This sequence is the one implemented in the existing process to be improved. Our interest in the sequence is the corresponding operations sequencing, first without figuring the resources in order to obtain a generic representation. The resources representation with their assignment to tasks and selected LoAs depends on a corresponding assembly system alternative. They are studied through next steps. To limit the representation to the assembly sequence, we represent first in Figure 25 the list of components assembly using an AND/OR graph.

Concerning the operations representation and schedule, and to facilitate automation alternatives description and analyses, we use the ASML language representation (chapter 4). The generic representation is performed in section 6.10.3. Then, based on the obtained generic model, alternatives of processes with automation levels will be defined and analyzed through the study: the current process and 2 proposed automation alternatives.

6.10.3. The generic process representation

For a selected assembly sequence presented in Figure 25 shown by a numbered AND/OR graph, we start by representing the assembly process generic model. We mean by genericity the non-dependence to a certain resources allocation or LoAs. The idea is to be able to generate multiple assembly systems model based on this model and to perform analyses and cost predictions. The obtained ASML model representing the studied process is shown in Annex C Figure 1. The model shows the operations required to perform the product assembly based on product design analysis and the assembly techniques natures allowing fixing parts to each other. The model provides all possibilities of assembly systems definition, for a given assembly sequence, later by resources allocations to operations and automation levels association to resources and concerned assigned tasks. In our case we will use this model to study other automation alternative from cost point of view. Some constraints, as the required takt-time satisfying, should be later fulfilled when defined automation alternatives for an appropriate assignment and dimensioning of the resources (chapter 5) as it will be presented in the next section.

6.10.4. The assembly process input information for cost computation

After the presentation of the generic model, we present in this section the data on cost that are common to all possible automation alternatives. This standard data concerns the planned production strategic information and does not depend on given selected automation options. It consists in the parameters which does not have the "*R*" index in the previously defined cost model. The standard data and their values are shown in Annex C Table 1 for the planned production information and Annex C Table 2 for the product parts costs.

Concerning the rework strategy, we consider the case B of the proposed model (section 6.6.3.3.1) for which a non-conform product should have a second pass through the process. In fact, as modelled in the generic process of Annex C Figure 1, no dedicated operations are represented concerning the rework of non-conformities. This means that no resource will be used as a dedicated station for non-conformities rework.

In the next section 6.10.5, automation alternatives are studied, cost estimated, and analyzed. These alternatives will be compared and more thoroughly analyzed, discussed, and compared in section 6.10.6.

6.10.5. Automation alternatives description, cost estimation, and analyses

Once all standard data and generic assembly process model are available, alternatives of assembly systems with automation options can be defined and analyzed. In this section, we define 3 alternatives: the actual process alternative: the one installed during our study, a first proposed alternative with automation assistance, and a second proposed alternative with an increased automation. These alternatives have been defined through discussions with our industrial partner. The 2 new proposed alternatives aim at reducing the assembly cost. Each of the 3 alternatives is modelled; cost estimated and analyzed using the proposed approach. In next section the 2 proposed alternatives will be compared and more deeply analyzed from automation decision perspective with regard to the currently installed assembly system.

6.10.5.1. The existing assembly process alternative study

Based on the process generic model performed in section 6.10.3, we represent the current process model and we estimate the assembly cost using our proposed cost model. We did the same for the 2 proposed automation alternatives. Each time we proceed in 3 steps: first by the resources allocations and time estimation, second, by cost input information presentation of the corresponding process, and finally by the cost estimation results and analysis.

- **Resources allocations and time estimation for the current process**

Based on the modelled standardized representation of the process in Annex C Figure 1, we represent the tasks assignment of the existing assembly system with associated resources. We obtain the actual assembly system configuration shown In Annex C Figure 2. Based on this figure, we represent in Annex C Table 3 an enumeration of the different process resources. The different resources are listed in column 1 of the table with their corresponding automation levels in column 2, and a brief description of the nature of the resource in column 3. The assigned set of tasks $\boldsymbol{S_{Ri}}$ to each resource is shown in column 4. According to the time estimate of every task $\boldsymbol{T_k}$, with consideration of the corresponding resource LoA, we store in column 5 the total resource time estimation $\boldsymbol{T_{Ri}}$ which may require, as considered in the next section, duplications or renovations according to the selected takt-time, resource working life, and planned production life. This has to be involved in the cost estimation as previously described in the model proposal. In column 6, we store the concerned product assembly parts in the level of every resource according to assigned tasks and involved parts that can be impacted when non-conformity can happen because of the resource and can cause a reject. In column 7 we compute the total parts costs based on involved parts enumerated in column 6 and the cost of the different parts to be assembled shown in Annex C Table 2.

- **Resources cost input information for the current process**

The cost input parameters related to the existing process resources are here detailed. The input parameters are classified, as previously done with the proposed general case cost model, to parameters specific to manual, to machined resources, and to commonly parameters used for manual and machined. The classified input parameters are described in Annex C Table 4.

At this stage, all required cost input information for the cost model are gathered. The cost estimate can be then performed to each of the modelled process resources. The total assembly cost defined as the sum of the cost generated by each resource can be also computed.

- **Cost estimation and interpretation for the current process**

The assembly cost generated by the current assembly system is estimated in this section. The cost estimates are performed using the proposed cost model and the concerned input information detailed in Annex C Tables 1-4. As defined in the proposed model, we also present the results by resource and by cost category. The results are shown in Annex D Table 1.

Based on the obtained results of Annex D Table 1 obtained by application of the cost model, it can be observed that the total assembly cost per product is of 4.78€. By discussion with the manufacturer stakeholders, this value to the considered input data is realistic and credible compared to their current process assembly cost.

It can be observed based on the same table that the resource R1 which is a full manual labor resource costs 2.42€, so more than the half of the total assembly cost. This value concerns only the full manual part. If we also consider the manual part involved in manual with automation assistance resources (R2-R8; R10-R14), the manual cost contribution would be higher. This can be a sign to increase automation level for some or all tasks concerned by most costly manual resources, especially resource R1.

We also realize by analysis of the output data of Annex C Table 1 and Annex D Figure 1 that quality cost (C_Q) generated by the different resources of the actual process is of 2.22€, which is about the half of the total assembly cost with a major cost of rejects of 2.19€. This is coherent with the identified problem found in the company (historical data files of the company) and for which the manufacturer is asking for automation possibilities analyses: the risk of operator error in assembly, non-conformities generated because of confusing parts, the high number of operations and sequence of components to be memorized, especially concerning the full manual resources (R1). This consists in the key point leading to define the next studied automation scenarios to tackle these problems and excessive costs by two different proposed alternatives: a first one with a minimum automation investment proposing an assistance and guidance to operators during assembly (section 6.10.5.2), and a next one with an automation of the major assembly operation using a dedicated automated machine (section 6.10.5.3).

6.10.5.2. A proposed 1st alternative process with automation assistance

This alternative with an automation assistance system is proposed to assist the operators during the assembly. The aim is to facilitate selecting the appropriate components and their orientation using: lights indicators, automatically controllable components cover plates to authorize or forbid the access to right and wrong components according to the required parts to be assembled, and a screen with work instructions synchronized with the assembly steps. This will assist and accompany particularly resource type R1 during the assembly.

- **Resources allocations and time estimation for the proposed 1st automation alternative**

As previously explained, the only change of this proposed alternative concerns the assistance to the manual resource R1 causing high costs, especially rejects costs. The modelled process of this proposed alternative is shown in Annex C Figure 3 and Annex C Table 5. It can be observed that all resources are unchanged except resource R1 which becomes green colored instead of grey colored. This is due to its LoA modified from full manual to manual with automated assistance. In Annex C Table 5 in which we detail the resources and assigned tasks, the only change, highlighted in green, concerns the resource R1 LoA. No other modifications exist, including the tasks allocation as well. We also assume no change in this resource time estimation, kept to 183 sec same as for the previous actual alternative, even if it should slightly decrease with the use of the automated assistance because the worker will be helped by the automation guidance and will spend less time in components selection and in reasoning about assembly and appropriate components and operations.

- **Resources cost input information for the proposed 1st automation alternative**

Concerning the resource cost input information for this proposed alternative, some changes concerning resource R1 exist. These changes are principally related to the automation assistance system. As resource R1 becomes manual with automated assistance, the costs related to machined resources should be filled for R1. These related data are highlighted with green in Annex C Table 5-6. We assume a cost of 6000€ of such systems with a very low failure rate because of its simplicity. Such systems are also having low energy consumption. Concerning the MTBF and MTTR, respectively 5000 hours and 10 hours are for as broadly reasonable. Other changes for R1 concerns the percentage of non-conformities (%Non-Conform) and of rejects (%Reject). As the automation system is tailored to this issue, these percentages are assumed to decrease, as shown in Annex C Table 6, respectively from 5% to 2.5% and from 2% to 0.5%. The impact of these changes on cost is studied in next part.

- **Cost estimation and interpretation for the proposed 1st automation alternative**

The use of the cost model we proposed gives the results shown in Annex D Table 2 and Annex D Figure 1. It can be seen that the cost per product falls to 3.83€. This give 0.95€ saved per assembled product compared to the previous scenario. With consideration of the planned yearly volume, this gives a saved amount of 28606€ per year and a total amount of about 171638€ throughout the planned production life. Moreover, half non-conformities are saved compared to the full manual actual system. Also, non-detectable non conformities that can reach the customer and cause bad painting efficiency or customers' dissatisfaction, such as the ones caused by inappropriate parts orientation or mismatch of appropriate seals, should be reduced with the new automated assistance. This proposed solution seems to be meaningful and promising also according to discussions with the company stakeholders and operators. However, we study in next section a more increased automation corresponding to the first company wishes to observe if it can be more profitable.

6.10.5.3. A proposed 2nd alternative process with advanced automation

This section aims at studying the possibility of a more advanced automation level and its impact on cost. The purpose is to try to decrease more the assembly cost by completely automating the assembly work areas causing non-conformities and human errors. We propose then a global automation of almost assembly operations of all the product parts except the needle assembly representing the last part to be assembled and following multiple inspection operations that will be kept unchanged compared to the initial process. The changes are more deeply presented in the following next part.

- **Resources allocations and time estimation for the proposed 2nd automation alternative**

In Annex C Figure 4 the model of this proposed alternative is represented. As it can be observed, all assembly parts except the needle are automatically assembled using an automatic dedicated machine R15 (LoA=Automatic). This resource executes all tasks from task T1 to T27, consequently, until the lapping bench, which represents a second automatic resource R9 performing task T28. The tasks T28 to T40 and their associated resources are unchanged compared to the initial existing process of Annex C Figure 2. Only R1, the full manual resource, will have some information modified compared to the initial process, as shown in Annex C Table 7, because it performs in this new scenario less tasks than initially. These missing tasks for R1 are here executed by the automatic assembly line R15. Resource R1 executes then, as shown in Annex C Figure 4 and Annex C Table 7 the tasks T29, T31-T32, T34-T36, and T39. These tasks are related to the quality inspection operations and to the needle part assembly. The time estimate of this manual resource R1 decreases then to 34 sec instead of 183 sec thanks to the use of the new resource R15. Resources R2 to R8, consisting in the different presses with manual and automated assistance LoA, are removed in this alternative. These press fitting tasks are performed by the automatic assembly line R15 using completely automatic presses with automated conveyor handling and automatic press fitting ensured by R15.

- **Resources cost input information for the proposed 2nd automation alternative**

The cost input information of this alternative is expressed in Annex C Table 8. Conformity to our previous presentation, changes as highlighted with colors in the table basically concern an apparition of the automatic resource R15, modifications of resource R1 cost information, and removal of resources R2 to R8. The changes concerning R1 grey color highlighted assigned to less tasks compared to the initial process, concerns a diminution of tools costs and of the training costs required to master the much fewer tasks. Its associated time T_R also decreases as previously mentioned. The total cost of assembly parts C_{Parts} that can be impacted by this resource error and reject also decreases to 4.5€ as analyzed in Annex C Table 7. A last modification for this resource concerns a diminution of the required surface to perform the new assigned set of tasks.

Concerning the automatic resource R15, the cost information is listed in Annex C Table 8 and are blue highlighted. Most important information is the initial investment cost of 60000€ with 1000€ of annual spare parts costs for maintenance, a failure percentage of 20%, so a reliability of 80%, with a MTBF of 1000 working hours and a MTTR of 20 working hours. We assume a power of 15KW and a required line surface of 20m^2. The assembly estimated time for this resource is of 143sec with 50 sec of setup to run a batch of 60 parts. This gives a better margin to handle demand fluctuations and high pics of productivity than the too adjusted current process. According to the assigned tasks, the cost of concerned assembly parts in case of reject cost is of 55.88€ as shown in Annex C Table 7. The non-conformities rate is assumed to be 1.5% and the rejects percentage of 1% (Annex C Table 8). The percentage are slightly lower because such machine is supposed to rarely make errors with parts already correctly oriented and fed to the machine. The impact of these different changes is analyzed in the following section.

- **Cost estimation and interpretation for the proposed 2nd automation alternative**

The cost impact of this advanced automation alternative are shown in Annex D Table 3 and Annex D Figure 1. According to the new architecture and the different input data, the obtained cost per product decreases to 3.4482€. Compared to the initial process, 1.34€ is saved per assembled product. An amount of 0.39€ per product is saved compared to the previous proposed alternative. Moreover, we would like to remind that this supposes an enough reliable machine with little rates of non-conformity (1.5%) and rejection (1%) caused.

With consideration of these input information, and by modification of these quality rates, we realize using the cost model that rates of non-conformities and rejections rates providing the same assembly cost of initial process (4.78€) are respectively of about 6% and 3.4%. Consequently, if the real rate values of the given automatic line (R15) are higher than these values, the process will be less profitable than the current manual process. Otherwise, it would cost more than 4.78€ per assembled product. The process would be not profitable.

As another simulation, if we regain the admitted non-conformities and rejections costs of Annex C Table 8, respectively 1.5% and 1%, and we try to find the limit threshold of automatic assembly line resource R15 initial investment cost, we realize that the global process is still profitable compared to the current manual process until an initial acquisition cost of 240000€ for R15 with annual spare parts costs for maintenance of 10000€. This assumes the non-conformities and rejections rates previously mentioned are guaranteed. Such costly process can consist in technologically advanced industrial robots cells with performant vision systems and parts features recognition. It should be highlighted that if such system exceeds these amounts, especially for such performance systems with high assembly accuracy and complex assembly of relatively small parts can easily be higher than 500000€. Automation in this case will be not profitable anymore. This underlines that automation is not always profitable and depends on multiple aspects according to the given industrial case.

The discussed values can then be useful for manufacturers and gives good orders of magnitudes about costs and required performances to negotiate with machines suppliers purchasing costs, but also machines performances, quality, reliability and warranties, before engagement in a new investment or industrial installation, because the long-term consequences can be crucial and significant.

In next section, the proposed cost performances indicators presented in the model proposal are computed for the 2 proposed alternatives. This will help on comparing the profitability of these solutions with regard to the current installation and evaluate if such investments can be worth doing, and which one can be better as a support to the automation decision.

6.10.6. Proposed alternatives performance indicators computation and analysis

In this section, we compute the different performance indicators presented in section 6.9 for the two proposed alternatives. We use these indicators to compare these alternatives by identification of advantages and drawbacks of each of them. The indicators are computed with regard to the reference case: the one of the initial current process and its associated obtained cost of 4.78329€ per assembled product and the total process investment cost of 22840€ including an initial investment of 21340€ and one time renovation of resource R3 of 1500€, as it can be observed in Annex C Table 6. When projecting this renovation on the time scale according to the resource working life (400 working hours), the planned production life (6 years - 72 months), the yearly volume (30000 units), and the spent time per product for resource R3 (10 seconds per assembled product), we realize that this renovation should occur around the month 57 among the 72 months production life. This renovation of R3 concerns the current process solution, and also similarly the proposed alternative 1 with the automation assistance (Annex C Table 6). For the alternative 2, resource R3, as previously mentioned, is removed and replaced partially by resource R15 (Annex C Table 7-8). Alternative 2 does not include any renovation as it can be observed in Annex C Table 8. This alternative presents then an additional profitability of 1500€ compared to the reference case current solution starting from month M57. This can be observed on an increase small acceleration in the alternative 2 profitability curve (blue colored) of Figure 26 around month M57, while alternative 1 is following the same increase monotony because these curves compare the proposed alternatives with regard to the current process solution. The detailed performances comparison results are shown in Annex D Table 4 and Annex D Figure 1.

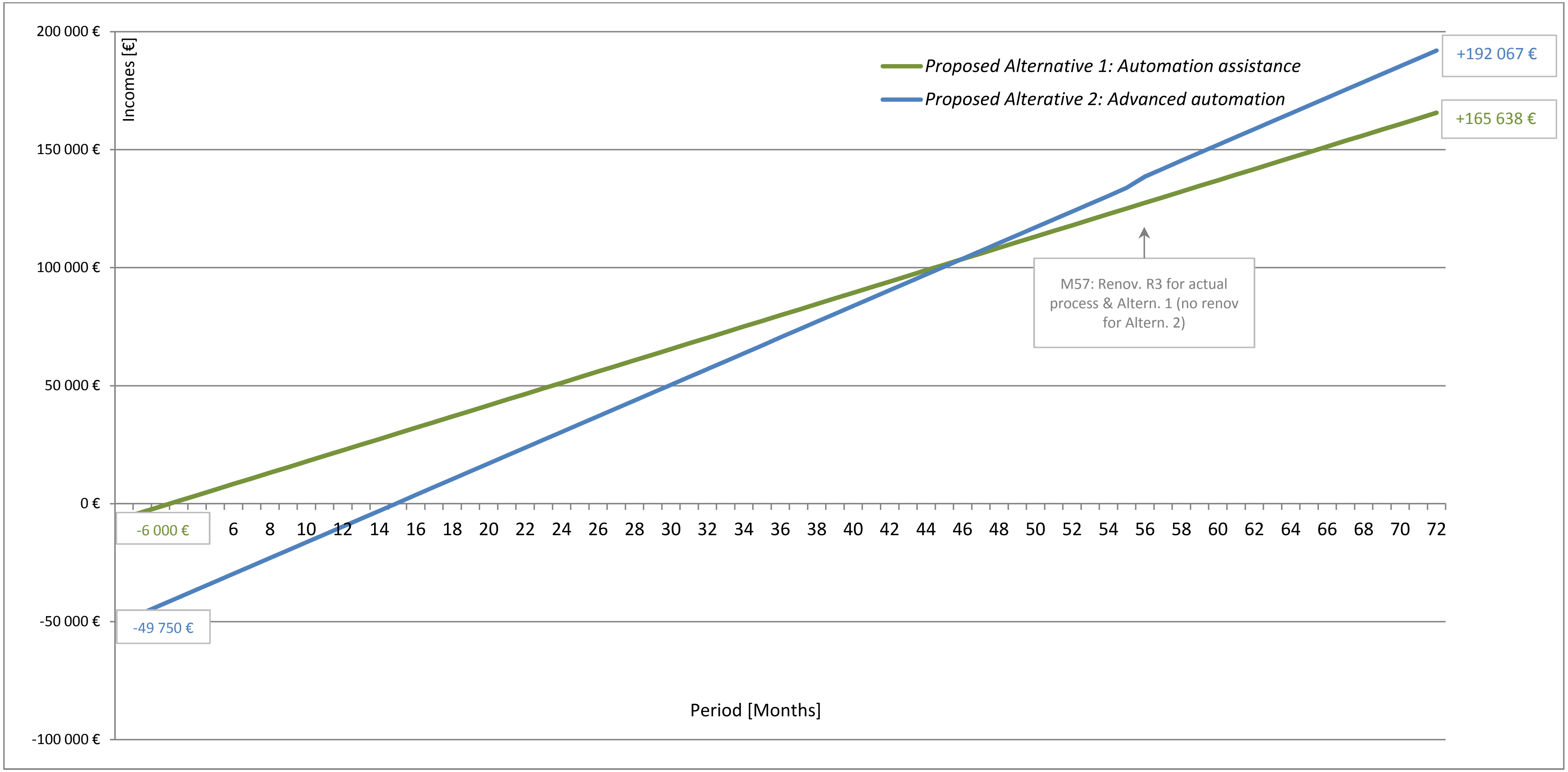


Figure 26: Investments, payback periods, and total incomes through the planned production life for the proposed 2 alternatives compared to the actual process solution

Based on Figure 26 showing profitability of the 2 proposed alternatives compared to the reference case, it can be observed that initially alternative 1 is more profitable than alternative 2 because it requires much less investment: only 6000€ corresponding to the automation assistance system investment cost compared to the actual manual process solution, while alternative 2 requires 71090€ as initial investment, consequently 49750€ more than the actual process solution.

At about month 2.5 and 14.5 in Figure 26, respectively curves of alternative 1 and of alternative 2 cut off the abscissa axe. This corresponds to the start of profitability and processes payback of these solutions compared to the reference case actual solution. This is confirmed in Annex D Table 4 (computed payback periods).

At the month M46 in as shown in Figure 26, an intersection between the 2 curves occurs: the one of alternative 1 and the other of alternative 2. Starting from this time slot, alternative 2, initially much more costly, becomes more profitable then alternative 1.

At the end of the production life, it can be seen on the curves of Figure 26 that the total profitability income of alternative 2 compared to the current process solution is about 192 067€, while the one of alternative 1 is about 165 638€. Alternative 2 presents then an extra profitability of about 26K€ versus alternative 1 through the whole production life. The return on investment (ROI) is about 4 times higher (Annex D Table 4) if we consider the 49750€ additional investment compared to the current process investment cost.

Even if the profitability of alternative 2 can be evaluated as not significant compared to alternative 1, more thorough analysis should be handled.

By analyses of Annex D Table 4, alternative 2 requires less surface space than alternative 1 (63m2 Vs 89m2), and consequently less surface cost (Annex D Table 4). Yet, even if the assembly process power of alternative 2 is lower than the one of alternative 1 (22.745KW Vs 30.85KW), the effective energy consumption to perform the concerned assembly tasks is more costly (257.55€/Month Vs 44.26€/Month) because almost assembly operations are concentrated on the automatic assembly line R15 representing the most important power and which should be running during almost all the time, while alternative 1 is composed of small presses separately not too energy intensive and running independently during short laps of time.

From another side, it could be worthwhile to remind that alternative 2 represents little less quality issues than alternative 1 (total quality cost per product of 1.17€ (Annex D Table 3) Vs 1.23€ (Annex D Table 2)). This can also imply less undetectable non-conformities that can reach the final costumer thanks to the automatic assembly line limiting human errors. From this point of view, again alternative 2 can be more interesting.

It can be realized that each of the processes presents advantages and limitation: a minimum investment with a limited investment risk, few maintenance, and an interesting return on investment for alternative 1, while more profitability, fewer error risk with less manual involvement, less surface required for alternative 2. For a final decision, all these elements should be taken into account and the best interesting compromise, involving also the manufacturer culture and experience. For example, some manufacturers do not accept a payback period longer than one year. Other manufacturers prefer solutions with initial minimum investment and low or no risk. For such cases, alternative 2 could be directly eliminated. In other cases, some manufacturers have more tendencies to advanced technology processes, to computerized control, easier and automated traceability of automatic process, or remote supervision. In those cases alternative 2 can be the most appropriate to such manufacturers.

6.11. Discussion: cost approach suitability to LoA decision

The cost approach we propose in this chapter allows predicting the assembly cost of systems alternatives with various automation levels options. This approach is tailored to support automation decision making. It is defined to fulfill different specified requirements and exigencies in this field of automation decision, particularly the ones summarized and presented in section 6.4. As a result, the proposed approach is defined as a quantitative approach with both analytic and parametric issues. It uses bottom-up reasoning with a low granularity level by analysis performed on the level of the detailed process and assembly operations. It is built so that it can be applicable during the particular early phase of assembly systems design. Moreover, all identified cost interesting features of the reviewed literature models are here combined and integrated into one complete and as exhaustive as possible model. The approach is then based on the product design and analysis of its complexity (row F1 of Table 21) basically when defining the initial process model and estimation of the assembly operations by analysis of the product parts complexity and corresponding operations. An automation level is associated to every resource (row F2 of Table 21) when describing an automation alternative process. The whole approach, allows handling multiple resources in the process (row F3 of Table 21) by the use of graphic modelling language to represent the assembly system alternative (row F4 of Table 21). This representation uses assembly operations and resources assignments representation (row F5 of Table 21) where the cost model estimates the costs per operations per resources (section 6.6 and 6.7) and then per process (section 6.8). The time estimate values are obtainable and literature sources are defined in beginning of section 3 (row F6 of Table 21). The proposed approach is standardized from modelling and cost estimation points of views (row F7 of Table 21). In fact it uses dedicated standardized language to graphically model the process, standardized vocabularies to represent the operations, generic rules to time estimate, and a standard cost model allowing to compute the cost whatever the resources are, their number, their connections architectures, and their LoAs. Resources cost rates computation (row F8 of Table 21) depending on the resource selected automation levels are handled using the proposed cost model as presented in section 6.6. All non-productive costs and overheads (row F9 of Table 21) can be taken into account in the proposed approach. Setup-time of resources is taken into account and amortized on the batch size. Other overheads can be taken into account by an extra-time consideration of concerned operations, by modelling these non-productive operations in the initial model. An associated cost can be then computed for these operations or can be easily added, as overhead per product, to the final assembly cost.

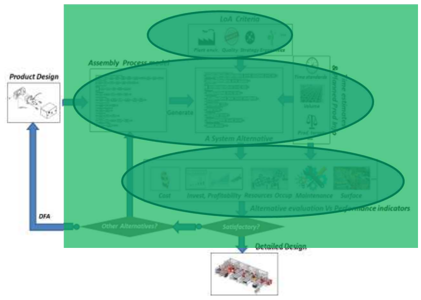
Concerning the identified set of literature models cost drivers that should be taken into account in cost estimation for automation decision as defined in section 6.4, all the 12 cost drivers of Table 22 are considered in the proposed approach. All These drivers, except cost driver D6, are considered by the model equations. Concerning cost driver D6 consisting in test, inspection, and quality control, it is handled by the process model as an assembly task, and should be modelled, if exists, in the process model. To do so, if standardized vocabularies of chapter 4 are used to model the operations, the task "Inspecting" can be used to model quality inspection operation.

The validation case study (section 6.10) highlights the ability of the approach to offer the possibility to accurately compare alternatives from cost performances point of view with a projection on the whole planned production life. Consequently, the proposed approach seems to match all the different fixed requirements and can be revealed promising in supporting automation decision from the cost criterion and profitability perspectives.

6.12. Conclusion

This chapter proposes a cost estimation approach to support automation decision during assembly systems design. The proposed approach includes modelling and cost estimation using an integration of cost models equations. The obtained model addresses identified tailored requirements to automation decision purpose defined based on literature review and analyses. The cost model is organized into categories of costs according to resources automation levels (LoAs). Depending on a selected resource LoA, categories should be included or not. For full manual resources, specific categories to this LoA concern labor, manual tools and equipment amortization, and operator's trainings costs. For full machined resources including dedicated automatic machines and robotic ones as well, similarly, specific categories include the heavy investments amortization, the energy cost, and maintenance. Other categories concern commonly manual and machined resources and include costs of setup, surface, and quality with rework and rejection costs. For manual resources collaborating with automated assistances, all categories are involved in the cost computation. For each of the mentioned categories, cost equations are proposed and supported by literature references. The whole cost model allows the assembly cost computation for a given resource dependently from its LoA. The proposed model is illustrated on a one page bloc scheme. The integration of this model in the computation of the whole process assembly cost with multiple resources is then tackled. Cost performance indicators are also proposed to support automation alternatives further evaluations, analyses, and comparisons. The cost estimation approach is then validated on an industrial case study treating different alternatives estimation, analyses, and comparisons. The case study underlines the contribution of this model into a quantitative and objective approach supporting assembly automation decision. The proposal seems then to be promising in the area of automation decision. The proposed model may be also useful for late phase cost estimation of existing assembly systems. It looks possible to use it on both early and late estimations. Yet, for late estimation, this requires validations and comparisons with real costs to accurately evaluate the error with regard to the exact cost value.

The next step is to combine modelling, time estimation, and cost estimation models to integrate them into an optimization program (chapter 7). In fact, there is a need to generate scenarios to be automatically evaluated and compared using a computerized tool. Combinatorial generator or mathematical programs seems to be interesting candidates. Other performance indicators, such as initial investment, may be associated as constraints with thresholds to be entered. From another side some enhancements may be addressed to the proposed cost model as openings. In fact, the model treats uniform productions and does not tackle fluctuating productions, variable volumes through the production life, or customized products assembly. The extension of the proposed cost model to handle heterogeneous productions can represent a future research axis. Also, the proposal only handles cost estimation within a wide multi-criteria decision about automation where several criteria, such as flexibility or ergonomics, have to be taken into account. The integration of a whole automation decision approach into one software tool, where the cost represents one criterion among a multitude of others, may represent one of future targets and finalities in the issue of automation decision support during assembly systems design.



Chapter 7. An optimization model for alternatives generation & optimization[6]

Abstract

This chapter is focused on optimization models and techniques to support automation alternatives generation and optimization. The target is to automate the computations through the proposed LoA approach and enable searching to the optimal automation configuration. According to the review in the area of LoA decision previously performed in chapter 3, no existing approach proposes a method to systematically generate alternatives or practically search to reach the optimal solution. Consequently, to seek for a model to solve LoA optimization problems, we proceed by reviewing similar fields to the topic. We present in this chapter the review results and findings. The core proposal is then a mathematical model driving alternatives generation, evaluation, and optimization based on a linear program formulation using integer programming technique to solve LoA decision problems. An illustration of the proposed program on a validation example is presented.

7.1. Introduction

The LoA decision issue is currently not sufficiently supported to reach the optimal automation configuration in assembly systems design. Tools guiding the decision are needed to facilitate the exploration of multiple scenarios with rapid evaluation of automation options to provide a quick support to experts and systems designers. Such evaluations are often limited and decisions are almost driven by intuitive approaches or built on analogies with previous projects without any concrete warranty to reach the optimal solution.

The review in automation decision literature (chapter 3) confirms the lack of methods guiding efficiently the search to the optimal decision, particularly for new systems design issue. Multiple encountered principles can be of high interest such as: product design and assembly complexity consideration in automation decision (Boothroyd, Dewhurst, & Knight, 2011) and high involvement of planned production information generally associated to cost computation and solution profitability evaluation (Windmark, Gabrielson, Andersson, & Stoehl, 2012). The exhaustive evaluation of all possible alternatives (Gorlach & Wessel, 2008) to guarantee the solution optimality represents another interesting attempt.

At this stage of this manuscript, multiple developments were made and presented through previous chapters. The developments proposed first a standardized modelling language with vocabularies allowing representing assembly processes based on product design by a generic model. Then automation alternatives can be defined based on this generic model. An associated time and cost estimation approach are then developed to help the right configuration of the

[6] This proposal is submitted to a journal and is under review: (Salmi A. , David, Blanco, Summers, & Briant, An Optimization Model to Support Early phase Assembly Systems Design & Automation Decision, Under review)

process and to evaluate the resulting assembly cost of a configuration. The whole approach handles, thanks to the modelling and reasoning, taking into account multiple decision criteria, by analyses of the assembly operation, or tasks, with regard to the criteria so that LoAs can be authorized or prohibited for the different model tasks. The issue then becomes the automatic generation of alternatives and the consideration of the different developments into a computerized tool. Such a tool can allow, if an efficient resolution technique is considered, to reach the optimal automation solution with a guarantee of optimality. This chapter is then dedicated to the optimization issue and to the possibility to generate automation alternatives and search to the optimal configuration.

The chapter is organized as follows. In section 7.2, we present a literature review in optimization of fields we think closed to the issue of assembly automation decision. The review includes balancing problems, resource planning problems, and systems design problems. Multiple helpful features were found and presented through the optimization literature review. A discussion tackled in section 7.3 shows a global status and highlights a need to build a dedicated optimization model to the issue of LoA decision. Inspired from different approaches found, a mathematical model for assembly automation optimization is proposed in section 7.4. The proposal is validated on a demonstrative assembly case example in section 7.5 by application of the integral optimization approach. Finally, the chapter is concluded in section 7.6.

7.2. A review in similar optimization issues

The purpose of this section is to review the literature in problems that can be similar to systems design and automation decision. The review will be used to find analogies and ideas to solve LoA optimization and automation decision problems in assembly systems design.

7.2.1. Functional requirements

In the problem to be solved, we assume we dispose of a description of the process with an enumeration of tasks or operations with their immediate precedencies. This description shows the assembly sequence for a given single product to be assembled. Such representation can be performed based on the product design analysis, such as ASML (chapter 4). The representation, especially when described with macro-tasks, such as soldering(part_A, part_B), screwing(part_A, part_B, 4Screws), is independent from resources allocation and LoAs point of view. It is then supposed to be generic. The genericity can be helped by standard rules of modelling and the use of controlled vocabularies (chapter 4). The, an automation alternative consists in resources allocations to tasks with associated LoAs to obtain the complete process picture. Every considered alternative consisting in a combination of resources assignment with a selection of automation levels should be designed so that it can satisfy, with a just fit to purpose and no overproduction, the planned production targets. Time-based rules defined in chapter 5 can help on guiding appropriate resources assignment to ensure a fit to purpose cadence of resources and the whole process using time estimation and early balancing approach. Every alternative is then subject to analyses and evaluations with regard to performance indicators. The generation of all possible alternatives should be exhaustive and objective. It should guarantee the optimality of the solution to be proposed. The finality is to keep the best scenario according to performed evaluations. These evaluations are basically checking the problem requirements and input data with eventually a use of standardized databases of selectable information (time, cost, general rules of LoA evaluation with regard to criteria, etc) according to the problem characteristics.

7.2.2. A review in optimization

Based on the problem characteristics presented in previous sections, and considering particularly the brief description of section 7.2.1 with the presented functional requirements, we perform a review in optimization to search for analogous problems solutions that can be useful to solve automation decision problems.

In (Rekiek, Dolgui, Delchambre, & Bratcu, 2002), an exhaustive state of the art in literature optimization techniques used in assembly system design during the preliminary design stage is presented. The automation axis is unfortunately not evoked in the paper. The review is performed since 2002 and is naturally not up-to-date with the research evolution. Yet, it presents an interesting exhaustive review in the topic of assembly systems design. The paper highlighted the importance of research in assembly as a strategic core activity of crucial industrial importance. The need to develop assembly lines in the most efficient way is underlined. The authors explain that typically assigning operations to workstations for given cycle time and with a target of minimizing idle time and satisfying constraints between operations represents classical line-balancing problems. In such problems, no types of equipment are considered. In the case of existence of equipment types, the problem is called a resource-planning problem. A more recent larger scope review is presented in (Battaïa & Dolgui, 2013) in 2013. The review covers different fields: assembly, disassembly, and machine line balancing problems. The paper provides a taxonomy of different optimization problems and the different solution techniques proposed by the literature to solve them. The reviewed techniques include exact, simple heuristics, and metaheuristics up to 2013.

In this research, our goal is focused on assembly problems and exact resolution. We present in this chapter a review with consideration of mentioned reviews with a focus on assembly field and an update and integration of most recent research works. We use the literature classification used to categorize these kinds of problems to balancing and resource planning and we add a third category to the classification consisting in a combination of these two ones called in the literature, according to (Michalos, Makris, & Mourtzis, 2012), as systems design problems.

7.2.2.1. Balancing problems

According to the literature, in this category of balancing, resources are identical and supposed to be able to execute any task (Rekiek & Delchambre, 1998).

Problems in balancing are classified in the literature to 'Simple Assembly Line Balancing Problems' (SALBP) and to 'General Assembly Line Balancing Problems' (GALBP) (Rekiek, Dolgui, Delchambre, & Bratcu, 2002). GALBP are also known in the literature as mixed-assembly lines balancing. SALBP tackles single products lines while GALBP treats lines handling multiple products and particular configurations. Both of the problems are admitted as NP-hard problems. GALBP problems are generally treated as a combination of two problems: line balancing and product variants ordering. The problem generally concerns existing processes. The lines flexibility represents one of the major issues in GALBP problems where some resources have to handle different variants, even if products in mixed-assembly lines have usually high similarities for technical and feasibility targets. For GALBP problems, the review of (Rekiek, Dolgui, Delchambre, & Bratcu, 2002) presents only three works: (Lee & Johnson, 1991), (McMullen & Frazier, 1998), (Rekiek & Delchambre, 1998). For most recent references, we suggest also the works of (Ramezanian & Ezzatpanah, 2015) and (Gökċen & Erel, 1998) handling different variants of products. GALBP are less related to our focus in automation decision for which we assume, as mentioned in section 7.2, a unique product to be assembled. Interested readers in

GALBP are then invited to find the review of (Rekiek, Dolgui, Delchambre, & Bratcu, 2002) or the mentioned references. This work does not focus on the problem where the flexibility represents the main issue. Flexibility represents for us a secondary criterion among the several criteria to be considered in the automation decision. Namely, the optimization module is for us indirectly concerned by flexibility management since it may be treated earlier in the LoA decision methodology as presented in chapter 3. The optimization is, in our case, receiving the sequence of operations as input, of a unique or eventually multiple products. It has to handle these operations, if they belong to a unique product or not. Also, flexibility can be managed by the presented approach by forbidding inflexible LoAs for operations concerning different variants of products, for example: forbidding dedicated automatic LoA for such tasks.

Multiple works around heuristics and metaheuristics resolutions are also proposed in this field to decrease high computational time of exact resolutions and to tackle high size problems. As in our case we seek to satisfy requirements specified in section 7.2.1 to tackle the gap in the literature of automation decision methods by defining an efficient and objective approach, our priority is to focus on exact resolution methods. Readers interested in heuristics are invited to find the review of (Rekiek, Dolgui, Delchambre, & Bratcu, 2002).

According to the previously presented definitions and to the particularities of our problem, we concentrate our review objectives then on exact resolutions particularly of SALBP problems. Multiple exhaustive reviews on SALBP exist. Readers can find the review of (Rekiek, Dolgui, Delchambre, & Bratcu, 2002), (Baybars, 1986), and (Ghosh & Gagnon, 1989). In this chapter we present a synthesis of most related papers to the topic with old and recent works. The synthesis includes a global state of the art review with most recent researches that are not included in the existing state of the art literature papers because of their publication dates.

SALBP are classified in the literature to SALBP-1 and SALBP-2 (Rekiek, Dolgui, Delchambre, & Bratcu, 2002). In SALBP-1 the purpose is to assign tasks to stations so that the number of stations is minimized for a given production rate; while SALBP-2 seeks to maximize the production rate for a fixed number of workstations. In (Michalos, Fysikopoulos, Makris, Mourtzis, & Chryssolouris, 2015), assembly systems design problems are classified as SALBP-1. We think the LoA problem has more similarities with SALBP-1 than SALBP-2, even if it has more particularities and is not perfectly a simple SALBP-1 for multiple reasons. In fact, in the LoA problem, multiple resources options may exist depending on tasks types and to selectable automation levels, which is not considered in SALBP problems. Also, the purpose is not effectively minimizing the number of stations, but can be different depending on the manufacturer's targets. For example, the objective can be minimizing the cost per product, minimizing the total process surface, maximizing the quality, satisfying ergonomics requirements, etc...

The exact resolutions found in the literature basically consist in mathematical formulations as will be detailed through this review.

A first model is defined in 1955 in (Salveson, 1955). A linear program is presented including all possible combinations of station assignments. The formulation is not perfectly efficient because of the possibility to generate infeasible solutions.

In a second formulation defined in (Jackson, 1956), a notion of a tree is used to tackle the problem where each arc represents a station and each path represents a feasible solution. All feasible assignments are first generated to the first station. Then all feasible stations are generated to the second. Then, for the combination of the first and second stations, all feasible solutions are constructed for the third station, etc. An exhaustive search is used each time for all remaining possibilities in the level of each considered station.

In 1960, (Bowman, 1960) proposes changing the linear programming issue of (Salveson, 1955) presenting unfeasibility issues to a 0-1 integer formulation. This formulation represents one of the simplest and most conventional formulations for typical SALBP-1 problems. We find useful, as proceeded in the review of (Rekiek, Dolgui, Delchambre, & Bratcu, 2002), to detail this formulation that represents a basis for more recently developed literature formulations basically using the same logic, but also for the most recent works. We explain to readers the detailed simple formulation and how such formulations can solve such problems.

- Decision variables:

$$x_{ij} = \begin{Bmatrix} 1: & \text{if operation } i \text{ is assigned to station } j \\ 0: & \text{otherwise} \end{Bmatrix}$$

- Objective function:

$$minimize\ g = \sum_{i=1}^{N}\sum_{j=1}^{M} w_j x_{ij}$$

- Constraints:

$$\sum_{j=1}^{M} x_{ij} = 1 \qquad \forall\, i = 1..N \qquad (1)$$

$$\sum_{i=1}^{N} t_i x_{ij} \leq C \qquad \forall\, j = 1..M \qquad (2)$$

$$x_{i_2 j_2} \leq \sum_{j_1=1}^{j_2} x_{i_1 j_1} \qquad \forall i_2, \forall k_1 = 1..N \ and\ i_1 \in P_{i_2} \qquad (3)$$

$$x_{ij} \in \{0, 1\} \qquad \forall i\ \forall j \qquad (4)$$

The model parameters are as follows:

N: the number of operations

M: the number of resources

t_i: the duration of the operation *i*

w_j: the weight or cost of a use of resource *j*

C: the line required cycle time

P_i: the set of immediate predecessors of operation *i*.

In this basic model, the objective function aims at minimizing the total weighed cost of assigning operations to resources. Constraint (1) ensures that all operations are assigned. Constraint (2) specifies that for the set of operations assigned to every workstation, the total duration satisfies the cadence target and the total duration is consequently not exceeding the corresponding required cycle time. The constraint (3) ensures the respect of precedencies between tasks: if a task *j2* is a successor of *j1*, it forbids the assignment of *j2* to a resource with an inferior index than the one of *j1*. Constraint (4) shows the domain of the decision variables as a Boolean in this 0-1 integer formulation. This simple formulation can be useful to explain the formulation strategy and provide basic skills in tasks assignment problems using mathematical models. It can be easily adapted to more specific problems as well as it can be observed according to the literature chronological evolution in this field of balancing problems.

Later in 1997, for a little different problem of tasks assigning and sequencing, Agnetis and Arbib (1997) proposed dynamic programming techniques to solve the particular problem they defined in which a sequence of ordered operations have to be assigned to stations in a flow line production. The problem target is to minimize the makespan or the inventory cost.

In (Andrés, Miralles, & Pastor, 2008), the authors enhanced the traditional SALBP problem by consideration of resources setup times required before running a group of assigned tasks on the resources. This is solved in the modelled 0-1 integer program by adding positions of tasks within every resource. The precedencies are then satisfied not only with regard to resources, but also inside the schedule of every resource in the presented formulation.

In (Tuncel & Topaloglu, 2013), a promising integer programming model is defined for balancing assembly lines with identical parallel stations possibilities to reach productivity targets. The formulation manages precedencies but also tasks-resources restrictions allowing or not the assignment of some tasks to some given specific stations.

Recently in 2015, (Ramezanian & Ezzatpanah, 2015) propose also a 0-1 integer programming formulation for tasks assignments in manual assembly with consideration of workers skills. Their model is limited to manual assembly.

7.2.2.2. Resource planning problems

Compared to the previous category of balancing problems, variants of selectable resources are considered in resources planning problems (Rekiek & Delchambre, 1998). The issue is to find the appropriate tasks assignment to resources with alternative equipment. Literature in exact resolutions also in resources planning is almost mathematical (Rekiek, Dolgui, Delchambre, & Bratcu, 2002).

In (Graves & Whitney, 1979), the authors present a linear programming model to solve resource planning, only in programmable assembly context. The issue is to optimize the assignment of tasks to the different existing robotic stations. No precedencies constraints are considered between tasks or in the assignments. The linear problem is solved using Branch and Bound techniques.

An integer programming model is defined in (Graves & Lamar, 1983) as an extension of the work (Graves & Whitney, 1979). This enhanced work represents a promising formulation of the problem with stations types' consideration. Yet, they still not consider initial precedencies constraints between tasks. The tasks assignments schedule represents one of the model's outputs in their model. The line can then run tasks whatever the initial schedule. In real assembly problems this can make solutions unfeasible because of mandatory order that can exist between parts to be assembled according to the product design and possible assembly sequences.

In 1988, the authors present an enumeration procedure with generation of all possible sequences (Graves & Holmes Redfield, 1988). The model considers the possibility of multi-product assembly. Flexibility issue represents one of the major occupations of their work. The approach uses a graph in which each candidate station represents an arc. An enumeration of all possible stations with costs computations is performed with selection of the least costly resource for each task. The best set of stations minimizing the total cost is kept at the end as the optimal solution.

In (Ağpak & Gökçen, 2005), assembly line balancing problem with constrained resources is presented. The problem is called RCALB (Resource Constrained Assembly Line Balancing). The resource constraint is related to the restriction for resources to a set of possible tasks. The proposed formulation is also modelled as a 0-1 mathematical integer program. By contrast to traditional works aiming at minimizing the total number of stations or the total cost, the innovative idea in this work is to deal with the maximization of resource usage. Two cases of problems are presented. In a first case, named RCALB-Type1, there is no task that can be assigned to different resources. Each resource has then a set of tasks that it can exclusively execute. No common executable tasks exist for the resources for this case. In the second case, named RCALB-Type2, some tasks can be assigned to different resources. In this case common tasks in the resources sets may exist. The issue proposes the priority to assign tasks that can be performed by an included resource in the configuration so that the total number of resources can be minimized as described in the objective function. The formulations of the two different problems are performed separately and are quite similar except few differences in some constraints definitions.

7.2.2.3. Integrated: Systems design problems

Assembly Systems Design problems are defined as a combination of assembly balancing and resource planning problems (Michalos, Makris, & Mourtzis, 2012). As a result, problems that belong to this category are also NP-hard (Michalos, Makris, & Mourtzis, 2012). The aim of such works is to build tools to help or guide the designers of assembly systems to appropriate alternatives. The approaches generally aim at finding satisfactory solutions rather than the absolutely optimal one as it is the goal for us. Encountered interesting research works are detailed through this section.

In (Bukchin & Tzur, 2000), the lack of literature in systems design problems is highlighted. The authors propose in the paper a 0-1 integer program for flexible assembly system design where several alternative resources and tasks precedencies exist. The purpose of their model is to minimize the total equipment cost. Yet, the assumption saying that each task can be performed by any resource limits considerably a realistic application of the method. In fact, in assembly, different techniques exist (screwing, soldering, riveting, etc.). Then, the existence of different types of corresponding resources and machines should be handled. Other important aspects for real industrial assembly, such as process surface or resources workload, are also missing. The total process investment cost minimization is the only target, which is not obviously the only issue for manufacturers, and not necessarily the most significant target. The formulation makes limits the method applicability to similar tasks and identical types of resources, or for infinitely flexible resources. Unfortunately, this does not match with requirements or with real industrial assembly systems design and automation decision issues.

The work of (Fasth, Provost, Fabian, Stahre, & Lennartson, 2012) also tackles the assembly systems design question. The paper discusses resources allocation and task optimization in systems design. They developed a tool named “Sequence Planner (SP)” which associates a graphic modelling language called “SOP” as Sequence of OPeration” (Lennartson, et al., 2010). The tool considers the flexibility as a prior criterion in the search to the appropriate solution. Each task is evaluated with regard to the convenience to the allocation to the different possible resources. To do so, 3 approaches are proposed. A first one proposes a global optimization and assigns each task to a unique resource as the optimal alternative. Yet, the way to find this optimal solution is not tackled. Only the allocation modelling is presented. A second approach models the different possible assignments alternatives of each task to the different resources. In a third approach, these possible resources are ranked for every task as assignment priorities or preferences according to criteria as the time, flexibility, etc. A final choice has then to be given according to the different ranked possibilities. This leads to a good feasible solution, but not evidently to the optimal solution. Our interest is more oriented to a global optimization technique than to heuristics of satisfactory solutions as we previously mentioned in the problem specifications.

In (Colledani, Franchini, Micchetti, Ratti, & Taurisano, 2015) and (Colledani, Bolognese, Ceglarek, Franchini, Marine, & Mistry, 2015), a methodology and software tool for assembly systems design are presented. The aim is to be able to generate (using a "Process Concept Generator" module) and evaluate (using a "System Configuration" module) alternatives of technologies and to keep the best one using an optimization procedure. The approach starts by stations modelling using a "process concept generator" module. Resources are selected from a database. This phase is guided and obligatorily performed by an expert user as mentioned in the paper. Then, using a components database and a reliability database, the system is configured in a second module called the "Station Configuration" module on which analytic optimization is driven. Our interest is focused on this last module. The authors mention that in this layer different alternative system configurations are automatically generated by an optimization algorithm and analyzed by the analytical performance evaluation model. It is also mentioned that the problem is modelled as a multi-objective optimization problem and solved using a commercial multi-objective optimizer. Yet, unfortunately the authors do not provide or describe the complete optimization model details in the paper. This model represents the goal of this research. Such a model could be helpful to solve LoA problems, even if there can be a need to adapt or enhance it since the authors do not mention handling different possible automation levels for resources.

In researches of Michalos et al (Michalos, Fysikopoulos, Makris, Mourtzis, & Chryssolouris, 2015) (Michalos, Makris, & Mourtzis, 2012), the authors describe a software tool to support the design of robotic assembly systems. As in the previous work, the authors define the problem as a multi-criteria problem. The authors propose a resolution using a search problem by application of intelligent search algorithms to find high quality solutions. It is mentioned that a multitude of criteria are considered in the tool such as the cost, productivity, and energy consumption. Concerning the methodology, the approach is quite similar to the one of the previously presented work of Colledani et al. The resolution is driven by two main phases: a design stage (stage 1) and a configuration stage (stage 2). During the design stage, an analytical way of calculating the required number of resources (robots) is provided according to input data that concerns the product (e.g. number of parts), process (e.g. cycle time), resources with generic information (e.g. their speed), and more specific information concerning available resources (e.g. MTBF, MTTR, Investment cost, energy efficiency etc..). In this stage an initial design is provided. Then, in stage 2, this initial design is further detailed via an intelligent algorithm capable of selecting specific resources for each station. For the different alternatives, a set of assignments between resources and operations are evaluated with regard to criteria in this stage. The evaluations are also combined to simulations. This generation is performed using a tree iterating all possible alternatives. As approved by the authors, such a way is time consuming and impossible to run for high or medium size problems due to the combinatory explosion. A need to intelligent algorithms is identified. The author proposes then a "search algorithm" limiting the search space and guiding the generation. The presented algorithm is based on random generation of nodes and leads to a smaller tree. The process is repeated until finding a satisfactory solution. The algorithm represents then a limited heuristic using a strategy of random generation. We think the procedure may converge to poor quality solutions. To obtain satisfactory ones, thresholds of considered criteria are used. The thresholds can be interpreted as performance targets corresponding to solution. They are entered by the user and allow then comparing the alternatives with regard to the expressed solution corresponding to the entered performances so that a solution, as good as the user's performances indicators, can be reached thanks to the algorithms' stop condition. One problem is that the good thresholds should be known in advance. This is not evident for high size projects. Even if these thresholds values can be available, the real optimal solution can be far better than the approximate one that designers can claim by the thresholds. Also, the authors mention that criteria are analyzed during the generations, (e.g. resources payload and compatibilities). Unfortunately the evaluation approach is not tackled in the paper. The approach is also dedicated to robotic assembly workstation design while we are interested in analyzing possibly hybrid processes with workareas automation in the assembly system.

7.3. Discussion: the need to develop a dedicated model

The previous review on optimization techniques for similar problems to level of automation decision shows useful existing models in different analogous fields: balancing problems, resource planning problems, and systems design problems. These fields, even if they do not completely match with the automation decision specifications and characteristics, have multiple similarities. The logic and reasoning strategies are of interest and may be helpful to this topic. Yet, a dedicated model to the automation decision issue has to be defined. Analogies of most closed found models can be driven to help such a model formulation.

Assembly systems design problems combining balancing problems and resource planning are the most closed to automation decision issue. Some additional particularities and constraints should be considered. Yet, the works in that field do not handle resources automation levels or automation possibilities. The research in systems design of (Colledani, Franchini, Micchetti, Ratti, & Taurisano, 2015), (Michalos, Fysikopoulos, Makris, Mourtzis, & Chryssolouris, 2015), and (Fasth, Provost, Fabian, Stahre, & Lennartson, 2012) present optimization strategies, tools, and software that can be adapted to automation decision and could be considered as similar to the automation decision reasoning presented of our LoA approach presented in chapter 3. Yet, the optimization models are not presented or detailed in these works. Only outlines and software architecture are presented. Only the work of (Bukchin & Tzur, 2000) in this category shows a detailed model as a 0-1 integer programming model.

Based on the reviews in the two other classes of balancing and resource planning problems, we realize that most of models are 0-1 integer programing models. Two types of decision variables are generally used: a first one for the affectation of a task i to a resource j, and a second concerning a resource j included or not in the solution consisting in the optimal assembly system design configuration. An integer mathematical model may represent, consequently, an alternative technique to model and solve the automation decision problem representing the target of our research.

In the category of assembly balancing, the 0-1 integer model of (Bowman, 1960) we previously detailed gives a global idea about how such models look like. This model is easily understandable and adaptable by making analogies according to a given problem specifications. From this same category, most found interesting and advanced one that can help automation decision issue is first the model of (Tuncel & Topaloglu, 2013). This model shows the possibility to duplicate identical resources to reach the productivity targets. The integer model of (Ramezanian & Ezzatpanah, 2015) is also interesting even if it only tackles manual assembly. Another interesting model is the one of (Andrés, Miralles, & Pastor, 2008) with the feature of tasks positioning within resources, considering resources setup times. It shows a mean to schedule assigned tasks to resources. Yet, as this does not represent a priority in automation decision issue and also may increase the problem complexity and computation time, it is preferable to perform this scheduling in a post treatment.

In the resource planning problems category, the work of (Graves & Lamar, 1983) considers various station types in a 0-1 integer formulation. The major issue is that no tasks types are considered. Only resources types are considered. These resources types have just an impact on the duration of running a task. Every task can still be executable in all types of resources of the process. All resources can be then suitable whatever what is the task to be assigned. This may represent a gap to treat real assembly systems design where different types of tasks, techniques, and corresponding resources can exist. A second interesting work in this category of resource planning is the most recent one defined in (Ağpak & Gökçen, 2005). The proposed 0-1 integer model deals with the possibility to assign tasks to resources. It supposes for every resource, basically known, sets of possible restricted possible tasks. Compared to other literature works, the approach is more concrete and more applicable to real industrial cases. Unfortunately, it supposes that resources are already fixed and known. The issue consists in the optimal assignment finding rather than optimal resources selections as it is the case for automation decision problems.

Based on the different reviews, it can be realized that developing a tailored model to handle the automation decision problem is confirmed. The development of such a model should consider the problem specifications and requirements. The model definition can be helped by the different interesting features found in the performed optimization literature review. The consideration of the research and literature evolution may be helpful. The tailored model to automation decision issue we are proposing is defined in next section 7.4.

7.4. An optimization model to drive scenarios generation and evaluation

As previously mentioned, the LoA problem has more similarities with systems design problems. It has also similarities with the SALBP-1 problem with additional features and constraints. Some similarities with the resource planning problem exist as well. This can be explained by the fact that systems design problem combine balancing and resources planning approaches. As all these categories are NP-hard problems, and as the LoA problem can be observed as an extension of the assembly system design problem, the LoA problem should be consequently, by the way, NP-hard.

The different reviewed literature models, particularly the ones cited in the discussion of section 7.3, help us to define this dedicated model. We consider the interesting features and a similar approach of an integer program model adjusted to the LoA problem by consideration of its additional specific aspects: different types of resources according to the LoA, their corresponding costs and surfaces, the different predefined assembly tasks and the possible resources they can be assigned to, compatibilities between tasks, automation possibilities to tasks, preferences to automate or not some tasks, etc,.. In fact, in the automation decision issue, the manufacturer is involved and can impose or forbid choices according to his preferences, practices, context, company culture, or strategies. Compatibilities between tasks to be executed by same selected resources also need to be managed according to their natures and LoAs. Concretely, we need to identify tasks that can be technically given to a same resource. For example, it can be possible to assign to a same operator (LoA=Manual) a sequence of riveting, painting, action while this combination is quasi-impossible for a same dedicated machine

(LoA=Automatic) because actions are too different or incompatible actions for such LoA and technically infeasible or extremely difficult be ensured.

According to the additional specificities compared to the reviewed models, we define a dedicated model to solve the LoA decision problem. The proposal is coherent with the performed reviews as a mathematical model. We proceed first by defining assumptions and parameters of the model in section 7.4.1. Then, in section 7.4.2 the model is detailed.

7.4.1. Assumptions, input parameters, and notations

In order to completely define the problem and better characterize its requirements, we state the different assumptions we propose to solve for the problem and precise the industrial context with consideration of our previous developments. We obtain the following assumptions:

(i) The set of tasks is known. Precedence relationships between assembly tasks may exist.

(ii) The precedencies include possibilities of sequential execution, parallel, or choice sequences.

(iii) Each task belongs to a task type. This type is a macro assembly operation or assembly technique (e.g. screwing, riveting, or welding).

(iv) According to the task type, a corresponding resource cost depending on the LoA should be associated.

(v) Tasks processing times are indivisible, deterministic and known. They depend on the task type, involved parts to assemble features, and the selectable LoAs.

(vi) In a given configuration, each task must be processed, only once, and in a unique LoA.

(vii) Resources can process only one task at a time in a unique LoA.

(viii) Resources can process sequentially multiple compatible tasks in a given LoA.

(ix) Compatibilities between couples of tasks defining the possibility to execute the tasks by a same resource in a given selected LoA are known.

(x) Possibilities to execute a task in the different LoAs are defined task by task.

(xi) In the level of every resource, obtained time estimation according to selected LoA and assigned tasks should not exceed the required takt-time expressing the productivity target. Else, the resource should be duplicated n times to reach the cadence.

(xii) The number of resources candidates is equal to the number of tasks as a maximum. Initially the set of included resources in the system configuration is empty. Then, as needed, resources are involved and are defined by assigned tasks and a selected LoA.

After definition of the different assumptions, we give the list of different parameters that are supposed to be known as inputs to the optimization model. In a first class, inputs are related to the automation decision approach, such as the LoA scale or the tasks vocabulary. Then, in a second class we find data related to the product design and assembly sequence that can be based on initial modelled generic process with operations as mentioned in our LoA decision method presented in chapter 3. We define them as the process information. Other data, stored in a third class, can be retrieved from databases of standardized data of tasks time estimates and resources information according to LoAs. This information is supposed to be generic, at least as default values. The user may be able to impose his own values if the standardized data is found to be not suitable for him. In the fourth and last category, data that belong to the planned production strategic information can be found. These parameters should be already gathered upstream from the manufacturer stakeholders. This information is more related to the given application. The different model parameters are listed below considering the defined classes and are ranked in alphabetic order with priority to non-capital letters first as follows:

- **Automation decision approach information**

m	Number of automation levels in the decision method; e.g. size of the LoA scale
n	Number of different possible variants of tasks in the decision method vocabulary
LoA_Scale	The decision method scale to describe the different $\boldsymbol{m}$ possible resources and tasks LoAs
T_Vocab	The vocabulary of all possible $\boldsymbol{n}$ variants of tasks

- **Process information**

i	Used index for tasks: *i* in [1..*N*]
j	Used index for resources: *j* in [1.. *Max_R*]
k	Used index for a selected LoA: *k* in [1..*m*]
l	Number of duplication of a resource: *l* in [1..*Max_dup*]
r	Number for renovation of a resource: *r* in [1.. *Max_ren*]
v	Variant of task type: *v* in [1..*n*]
C_Parts_i	C_Parts is a $Vector_{\boldsymbol{N}}$ of costs of involved parts in each task $\boldsymbol{i}$ of the process: parts involved and that can be impacted or harmed during assembly task $\boldsymbol{i}$ should be identified before consideration of cost of the given parts. The costs of all parts (purchasing cost, manufacturing, etc..) should be already gathered
N	Number of assembly tasks in the model
$T\_Comp^k_{i_1 i_2}$	T_Comp is a $Matrix_{\boldsymbol{NxNxm}}$ of tasks compatibilities: each task (row) is assessed with regard to all other tasks of the model (column) in the possible LoAs (cell in the intersection: a vector of *m* values): "1" if task $\boldsymbol{i_1}$ is compatible with $\boldsymbol{i_2}$ in LoA $\boldsymbol{k}$, else "0"
$T\_Succ_{i_1 i_2}$	T_Succ is a $Matrix_{\boldsymbol{NxN}}$ of the model tasks successors: each task (row) is assessed with regard to the other tasks of the model (columns): "1" if $\boldsymbol{i_2}$ is a successor of $\boldsymbol{i_1}$, else "0"
T_{iv}	T is a $Matrix_{\boldsymbol{Nxn}}$ of the model tasks variants: variant of task (vocabulary) to which every task $\boldsymbol{i}$ of the model belongs to: "1" if belongs, else "0"

- **Standardized data information**

t^k_i	*t* is a $Matrix_{\boldsymbol{Nxm}}$ of time estimation values of the model tasks: a value for each task $\boldsymbol{i}$ of the model in each LoA $\boldsymbol{k}$.
t_S^k	*t_S* is a $Vector_{\boldsymbol{m}}$ of required time to setup, configuring, or/and re-programming a resources according to their LoA to launch a production batch: a value for each LoA $\boldsymbol{k}$.
C_KWH	Cost of 1 KWH energy consumption [€/H].
C_L^k	C_L is a $Vector_{\boldsymbol{m}}$ of hourly labor cost rate for a standard assembly worker [€/H] according to LoA $\boldsymbol{k}$.
$C\_R^k_v$	C_R is a $Matrix_{\boldsymbol{nxm}}$ of investment cost of resources for the different possible $\boldsymbol{n}$ variants of tasks in the different possible $\boldsymbol{m}$ LoAs: a value for each variant of task $\boldsymbol{v}$ in each LoA $\boldsymbol{k}$. The initial cost includes resource purchasing, logistics, and installation costs.
C_S	Cost of 1 m^2 resource surface occupation [€/m^2].
C_SCP^k	C_SCP is a $Vector_{\boldsymbol{m}}$ of hourly labor cost rates [€/H] for a skilled worker to prepare, setup, configure, or program resources according to its LoA $\boldsymbol{k}$.
L_R^k	L_R is a $Vector_{\boldsymbol{m}}$ of estimated resources working life [$Hours$] for the different possible $\boldsymbol{m}$ LoAs: a value for each variant of task $\boldsymbol{v}$ in each LoA $\boldsymbol{k}$. The aim is to consider the disadvantage to renovate automated resources compared to manual ones.
NC_R^k	NC_R is a $Vector_{\boldsymbol{m}}$ of estimated resources non-conformities rates [%] in each LoA $\boldsymbol{k}$.
$P\_R^k_v$	P_R is a $Matrix_{\boldsymbol{nxm}}$ of resources energy power consumption [KW] for the different possible $\boldsymbol{n}$ variants of tasks in the different possible $\boldsymbol{m}$ LoAs: a value for each variant of task $\boldsymbol{v}$ in each LoA $\boldsymbol{k}$.
RJ_R^k	RJ_R is a $Vector_{\boldsymbol{m}}$ of estimated resources rejection rates [%] in each LoA $\boldsymbol{k}$.
$S\_R^k_v$	S_R is a $Matrix_{\boldsymbol{nxm}}$ of required surfaces [m^2] of resources for the different possible $\boldsymbol{n}$ variants of tasks in the different possible $\boldsymbol{m}$ LoAs: a value for each variant of task $\boldsymbol{v}$ in each LoA $\boldsymbol{k}$.

- **Planned production strategic information**

t_Takt	Required takt-time [sec] according to the required productivity cadence for the planned production. It can be directly imposed by a value if known, else to be defined as production duration interval (e.g. obtainable by the number of working days per year (*Ndays_Y*) multiplied by the number of shifts per day (*Nshift_D*) and the production duration of a shift per day (*Ndura_S*) divided by the corresponding required volume (e.g. Yearly volume).
$Altern$	The possibility to alternate resources; e.g. to reuse a resource to execute other tasks, so to be alternated by another resource for intermediate tasks in the sequence of tasks, for example: R1, than R2, than R1 (reused): "1" if possible to alternate, "0" if a resource cannot be reused.
Max_dup	Maximal number of duplicating a resource to reach the required productivity targets.
Max_ren	Maximal number of renovating a resource during the production life.
Max_R	Maximal number of resources allowed in a system configuration. By default, this maximal number is equal to the number ***N*** of the model's tasks.
Max_TR	Maximal number of grouping tasks per resource
$NWdays\_Y$	Number of working days per year
$Nshift\_D$	Number of working shifts per day
$Ndura\_S$	Duration of each shift [sec] per day
P_L	Planned production life duration [Months]
RwS	Non-conformities rework strategy: "1" if a second pass through the process is required for the rework, "0" if a dedicated station is used. In the case of a dedicated station, the rework station should be already modelled in the initial model. If no rework exists, use "0" and do not model rework operations.
$T\_Autom_i^k$	T_Autom is a $Matrix_{\boldsymbol{Nxm}}$ of tasks automation possibilities of the model tasks: a value for each task ***i*** of the model in each LoA ***k***. "1" if task *i* can be executed in LoA *k*, else "0". This matrix provides the possibility to allow ("1" for the task-LoA to allow), forbid ("0" in corresponding task-LoA to be forbidden), or impose ("1" for the task-LoA to be imposed, "0" for all the others) some automation levels choices according to some LoA criteria (such as, for example, welding that can be forbidden for manual LoA for quality issues). It provides also the possibility to take into account some choices of the manufacturer. This matrix should be filled by analysis of each task with regard to selected LoA criteria to be considered in the decision (chapter 2- Table 7 for LoA criteria table; section 4.8.3 – chapter 4 for more details about their consideration).
V	Planned total production volume
V_An	Required annual volume
V_Bat	The planned production batch size

7.4.2. The mathematical model proposal

After presentation of the different model assumptions, parameters and inputs parameters, we present the mathematical integer program which allows to model and solve assembly automation decision problems.

The proposal is a mathematical model defined by the decision variables, constraints, and objective function as follows:

- **Decision variables**

$$X_{ijlr}^{k} = \begin{Bmatrix} \textbf{1}\text{: if task } i \text{ is performed by resource } j\text{, in LoA } k\text{,} \\ \text{with } l \text{ duplications to reach desired cadence,} \\ \text{and } r \text{ renovations during the production life} \\ \\ \textbf{0}\text{: otherwise} \end{Bmatrix}$$

$$Y_{jlr}^{k} = \begin{Bmatrix} \textbf{1}\text{: if resource } j \text{ is included in the system configuration,} \\ \text{in LoA k, with l duplications to reach the required} \\ \text{cadence, and } r \text{ renovations during the production life} \\ \\ \textbf{0}\text{: otherwise} \end{Bmatrix}$$

F_j^k: The investment cost of resource *j* in LoA *k* according to its assigned tasks in the system configuration

S_j^k: The required surface for resource *j* in LoA *k* according to its assigned tasks in the system configuration

- **Constraints**

$$\sum_{j}\sum_{k}\sum_{l}\sum_{r} x_{ijlr}^{k} = 1 \qquad \forall i \qquad (1)$$

$$\sum_{k}\sum_{l}\sum_{r} Y_{jlr}^{k} \leq 1 \qquad \forall j \qquad (2)$$

$$X_{ijlr}^{k} \leq Y_{jlr}^{k} \qquad \forall j,i,k,l,r \qquad (3)$$

$$X_{ijlr}^{k} \leq T\_Autom_{i}^{k} \qquad \forall j,i,k,l,r \qquad (4)$$

$$\sum_{i} \left[t_{i}^{k} * X_{ijlr}^{k} \right] \leq t\_Takt * l * Y_{jlr}^{k} \qquad \forall j, \forall k, \forall l, \forall r \qquad (5)$$

$$\sum_{j}\sum_{k}\sum_{l}\sum_{r}\left[j * X_{i_1 jlr}^{k} \right] \leq \sum_{j}\sum_{k}\sum_{l}\sum_{r}\left[j * X_{i_2 jlr}^{k} \right]$$
$$\forall i_1, i_2 / \; (i_1 \neq i_2) \wedge \left(T\_Succ_{i_1 i_2} = 1 \right) \wedge (Altern = 0) \qquad (6)$$

$$X_{i_1 jlr}^{k} + X_{i_2 jlr}^{k} \leq 1$$
$$\forall j, i_1, i_2, k, l, r \quad / \quad (i_1 \neq i_2) \wedge (T\_Comp_{i_1 i_2}^{k} = 0) \qquad (7)$$

$$\sum_{i} X_{ijlr}^{k} \leq Max\_TR * Y_{jlr}^{k} \qquad \forall j,k,l,r \qquad (8)$$

$$\sum_{j}\sum_{k}\sum_{l}\sum_{r} Y_{jlr}^{k} \leq Max\_R \qquad (9)$$

$$\sum_{k}\sum_{l}\sum_{r} Y_{j+1,l,r}^{k} \leq \sum_{k}\sum_{l}\sum_{r} Y_{j,l,r}^{k} \qquad \forall j \in [1, Max\_R - 1] \qquad (10)$$

$$\frac{\sum_{i} \left[t_{i}^{k} * X_{ijlr}^{k} \right]}{3600} \leq \frac{L\_R^{k} * l * r * Y_{jlr}^{k}}{V} \qquad \forall j, \forall k, \forall l, \forall r \qquad (11)$$

$$F_{j}^{k} \geq T_{iv} * C\_R_{v}^{k} * l * r * X_{ijlr}^{k} \qquad \forall j,i,k,l,r,v \qquad (12)$$

$$S_{j}^{k} \geq T_{iv} * S\_R_{v}^{k} * l * X_{ijlr}^{k} \qquad \forall j,i,k,l,r,v \qquad (13)$$

$$X_{ijlr}^{k} \in \{0,1\} \qquad \forall i,j,k,l,r \qquad (14)$$

$$Y_{jlr}^{k} \in \{0,1\} \qquad \forall j,k,l,r \qquad (15)$$

$$F_{j}^{k} \geq 0 \qquad \forall j,k \qquad (16)$$

$$S_{j}^{k} \geq 0 \qquad \forall j,k \qquad (17)$$

- **Objective Function**

$$\boldsymbol{Min} \left\{ \begin{array}{c} \sum_{j}\sum_{k}\left[\frac{\boldsymbol{F_{j}^{k}}}{V} + \frac{C\_S * \boldsymbol{S_{j}^{k}}}{(V_{An}/12)} \right] \\ + \sum_{j}\sum_{i}\sum_{k}\sum_{l}\sum_{r}\left[\left[\begin{array}{c} [C\_L^{k} * t_{i}^{k}] \\ + RJ\_R^{k} * \left[\left(C\_Parts_{i} + C\_L^{k} * t_{i}^{k} \right) \right] \\ + [(NC\_R^{k} - RJ\_R^{k}) * RwS * C\_L^{k} * t_{i}^{k}] \end{array} \right] * \boldsymbol{X_{ijlr}^{k}} \right] \\ + \sum_{j}\sum_{i}\sum_{k}\sum_{l}\sum_{r}\sum_{v}\left[\left[\begin{array}{c} \left[\begin{array}{c} 1 \\ + RJ\_R^{k} \\ + (NC\_R^{k} - RJ\_R^{k}) * RwS \end{array} \right] \\ * \\ \left[\frac{t_{i}^{k}}{3600} * P\_R_{v}^{k} * T_{iv} * C\_KWH \right] \end{array} \right] * \boldsymbol{X_{ijlr}^{k}} \right] \\ + \sum_{j}\sum_{k}\sum_{l}\sum_{r}\left[\left[\frac{t\_S^{k} * C\_SCP^{k}}{V\_Bat} \right] * \boldsymbol{Y_{jlr}^{k}} \right] \end{array} \right\}$$

(18)

In this model we place the k index in the right up side of the variables to differentiate it from the other indexes and to simplify the understandability of the model.

We use 4 categories of variables: 2 Booleans (X_{ijlr}^{k} for tasks assignments to resources; and Y_{jlr}^{k} for resources candidates inclusion in the configuration), and 2 standard integers (F_j^k for resources initial cost, and S_j^k for resources surfaces; both according to the obtained tasks assignments and included resources).

To explain the proposed model, we present the different constraints and the purpose of each of them. Constraint (1) ensures that all operations should be assigned and to only a unique a resource. Tasks are undividable and not interruptible. Constraint (2) specifies that for a resource candidate, if included in a system configuration, the associated automation level k, the duplication number l, and the renovation number r are unique. Constraint (3) mentions that if a task i is assigned to a resource j, this resource j has to be included in the configuration. Implicitly, the constraint mentions also that a task and its executing resource have same indices of duplication (l) and renovation (r). Constraint (4) disallows selecting a forbidden automation level k to a task according to the input matrix T_Autom. Constraint (5) specifies the productivity target. It ensures that at each resource considering its selected LoA k, the resulting cadence according to the total time of assigned tasks and the appropriate computed duplication number l of identical parallel resources, satisfies the required productivity cadence defined by the takt-time t_Takt. Constraint (6) ensures that if task i_2 is an immediate successor of task i_1, then it should be assigned to a resource with a higher or equal index than the one of the resource to which i_1 is assigned. Constraint (7) disallows grouping 2 given tasks in a same resource if they are not compatible in a selectable LoA k according to the defined input matrix T_Comp. Constraint (8) ensures that the maximal number of tasks that can be assigned to a same resource should not exceed the input threshold number Max_TR. Similarly, constraint (9) ensures that the total number of resources included in the configuration should not exceed the entered threshold parameter Max_R. Constraint (10) imposes to include new resources among the resources with an increasing index order and concatenated indexes with no alternate non-involved resources indexes in a given system configuration. Constraint (11) allows the computation, for each resource, the renovation number r according to the resource workload. This workload results from corresponding assigned tasks time estimates and to the resource entered working life estimation L_R^k. Constraint (12) ensures the determination of resources required investment cost according to its assigned tasks variants in the given configuration, with consideration of obtained duplication l and renovation r. Similarly, (13) allows the determination of resources required surface. For the surface, the renovation r is not considered because a renovated resource will take place of the older one. Finally, constraints (14) and (15) define the decision variables X and Y to be binary while constraints (16) and (17) define F and S to be positive variables.

The objective function is defined in equation (18) of the proposed mathematical formulation. We opt for a cost model for this function allowing the computation and minimization of the assembly cost per product considering the different input parameters previously presented in section 7.4.1. We use then as an objective function the cost model we defined in chapter 6 with a little simplification with a non-consideration of some cost drivers to simplify the function as the focus of this chapter is the optimization model rather than the exhaustive cost computation.

In the objective function (18), four cost layers corresponding to the 4 sum sub-functions can be distinguished in this function ranked from the most significant to the least impacting:

- The first layer corresponds to costs caused by resources acquisition. These costs include resources investment amortization and the surface cost per product.
- In the second layer, labor rate is computed in the first member. The second member considers the rejections costs including costs of rejected parts and the labor cost lost in producing these rejections. Finally, the third member includes non-conformities to be reworked, so excluding parts to be rejected, are taken into account by time and labor rate consideration, and corresponding non-conformities with subtraction of rejections rates.
- The third layer considers energy consumption cost based on the power of running the resource according to the task type. As it can be observed, the energy cost per product computation takes into account the time taken to produce a uniform product, the amortization of produced rejected parts requiring energy that should be amortized, and finally the energy cost caused by reworking non-conform products. If a dedicated rework station is used rather than a second pass for rework (RwS=0), energy cost caused by the rework is not taken into account by this formula. It would be computed otherwise by modelled rework operations and their executing assigned resource.
- The fourth layer considers set-up costs related to resources preparation (even for manual stations), machines configuration (e.g. automatic machines), or programming (e.g. robotic or other programmable automatic stations). These operations can be time consuming and then costly, especially for high volume productions where idle time can generate expensive production loss costs, delivery delays, or penalties.

This cost model can be enhanced and be more exhaustive by consideration of other elements defined in chapter 6, such as maintenance costs. We consider in this model major cost parameters that can be considered as the most significant or preponderant cost drivers. Our priority is to differentiate selectable automation levels possibilities and their generated costs by involving the most impacting cost drivers.

Other different objective functions can be used and associated to the same proposed model, such as the maximization of resources workload occupation, load balancing, the minimization of the total number of resources, or of the total process surface.

As it can be observed according to the used indexes in X and Y, the resources (and involved assigned tasks) duplication number (l) according to time estimates and resources production life, and required renovation number (r) represent ones of the model outputs. Basically these parameters (l and r) are related to variable Y and could be used only for Y. Yet, if yes only for Y and not used for X too, this would make the formulation non-linear, particularly in constraint (12) when determining total investment cost of resources based on assigned tasks and numbers of duplications and renovation respectively l and r which are multiplied as it can be observed in (12). Another possibility to avoid the non-linearity could consist in writing the constraints otherwise: Instead of considering the resource investment cost only once according to assigned tasks types as mentioned in (12), computing the resulting initial cost by a same on the different assigned tasks for example. Yet, in this case, using a same resource for multiple tasks or using a resource for each task would be equivalent, and this becomes meaningless. That is why we opt for the maximum cost among the individual resources investment costs ($C\_R_v^k$) corresponding to each assigned task to the resource, as defined in constraint (12).

The determination of the number of duplications and renovations could be obtained by a use of additional integer variables rather than computing those using counters *l* and *r* as it is used in equations (5) and (11-13). But, as it can be observed, this would introduce also non-linearity issues particularly with equations (11-12). This explains the reason of integrating them as indexes, of computing duplications *l* and renovations *r* using counters. This represents one of the advantages of defining *X* and *Y* as 0-1 integer variables.

The proposed model is implemented in IBM ILOG CPLEX OPL 12.6 tool. It was tested on multiple numerical examples and allows reaching optimal systems configuration according to various tested objective functions previously mentioned. The next section 7.5 gives an application example in which we illustrate the whole approach; including modelling and optimization for a simple product assembly solved using the proposed model. We present and discuss the obtained results.

7.5. Application example

To illustrate the whole LoA decision approach and particularly the optimization model we proposed, we present an application study including the multiple aspects of the decision methodology. The application includes consequently process modelling based on the product to assemble design and the search to the optimal assembly system configuration using the optimization model.

7.5.1. The example presentation and input information

For a better understandability, we focus on a simple example inspired from the literature used in multitude of papers (Homem de Mello & Sanderson, 1991) (Zha, Lim, & Fok, 1998) (Salmi A. , David, Summers, & Blanco, 2014), and chapter 4 of this thesis. A small modification compared to the original product assembly of 4 parts used in these references (cap, stick, receptacle, and handle) is performed here by adding a fifth part (the seal) to make the study more complex to be intuitively insolvable and highlight a need to a computerized approach to drive its resolution. The product to be assembled is shown in Figure 27.

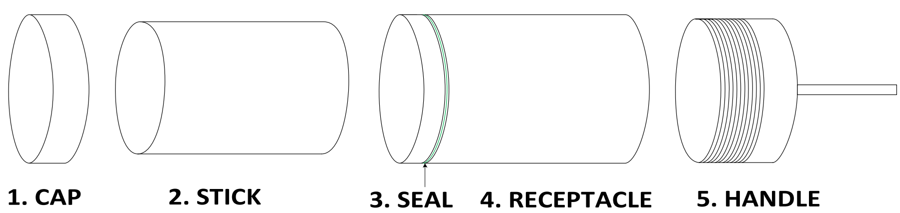


Figure 27: The application example product design

- **Analysis and process modelling**

By analysis of the product design, prototype, and design specifications, the assembly techniques of the different parts can be obtained. Then, an assembly sequence allowing assembling the different parts in an appropriate order has to be determined. Multiple assembly sequences may exist. In our study we select one among these feasible sequences to base the study. We determine impose a sequence and represent it in the following AND/OR Graph of Figure 28. The figure includes an assembly order using numeration (T_i) of the different assembly steps required to perform the corresponding parts assembly.

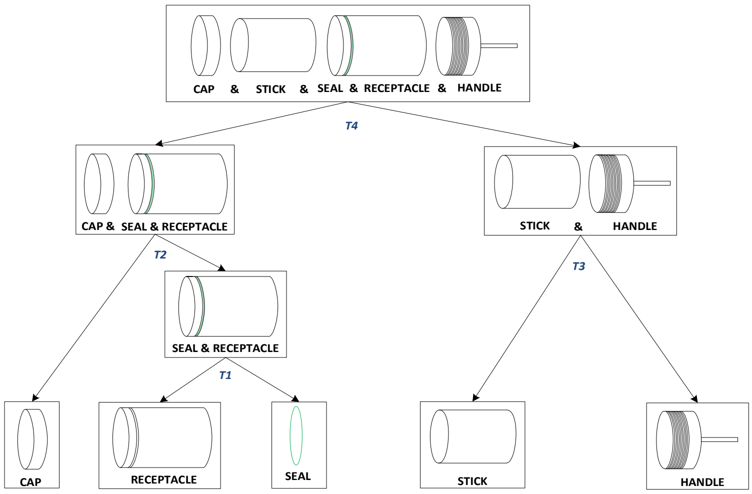


Figure 28: The defined assembly sequence

We assume the consideration, according to the design specifications, of the following tasks types (or techniques) for the different tasks T_i as shown in Table 23.

Table 23: The process tasks variants

	Task variant or technique	*Concerned parts*
T1	Placing	Seal & Receptacle
T2	Clamping	Cap & Seal & Receptacle
T3	Snap fitting	Stick & Handle
T4	Screwing	Handle & Receptacle

According to the selected assembly sequence and different tasks, the generic assembly process model using ASML language showing the different assembly tasks and process architecture can be built. This model is shown in Figure 29.

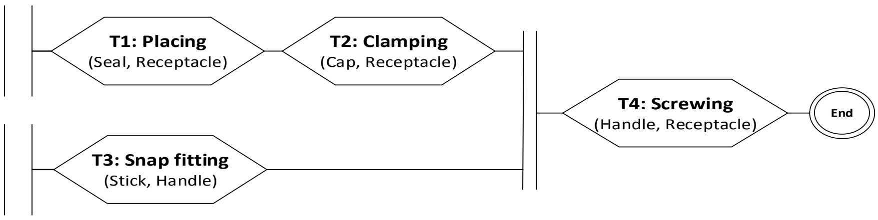


Figure 29: The generic process model

- **Automation decision approach information**

In our approach, we use a 4 levels LoA scale. Consequently, the set *LoA_Scale* = { Manual (M, LoA=1), Manual with automated assistance or tool (MT, LoA=2), Automatic dedicated machine (A, LoA=3), and industrial robot (R, LoA=4)}. Thus, *m*=4 and index *k* will vary in [1..4].

In this example, we have 4 variants of tasks according to Table 23 (4 different assembly techniques). Basically, in related research studies we defined a general vocabulary of 20 tasks enabling the description of a multitude of possible tasks in assembly (Salmi A. , David, Blanco, & Summers, 2016). To simplify the application in this study, we limit the vocabulary to a 5 tasks vocabulary that can allow defining the previously presented process. We consider then one extra task in order to show how the approach can be applied for a larger vocabulary use. The vocabulary we consider is defined as follows ranked in an alphabetic order:

T_Vocab={*Clamping*,*Placing*,*Riveting*,*Screwing*,*Snapfitting*}

According to this assumption, *n* is then here equal to 5 and index *v* will vary in [1..5].

- **Process information**

Concerning the tasks, it can be observed that we have 4 tasks according to the product design and model, so N=4 and index i will vary in [1..4].

With consideration of the process tasks previously presented in Table 23 and the considered vocabulary *T_Vocab* of 5 tasks, the matrix of the model's variants T is defined in Table 24 as follows:

Table 24: The matrix of the model's tasks variants

T_{iv}	*Vocabulary (index v)*				
	v=1 **Clamping**	*v=2* **Placing**	*v=3* **Riveting**	*v=4* **Screwing**	*v=5* **SnapFit.**
i=1	0	**1**	0	0	0
i=2	**1**	0	0	0	0
i=3	0	0	0	0	**1**
i=4	0	0	0	**1**	0

Using a vocabulary, even with a larger number of tasks, the definition can be easily established by filling such a matrix.

We consider the following material costs: 1.1€ for the Cap, 1.4€ for the Stick, 0.7€ for the Seal, 3€ for the Receptacle, and 2.5€ for the Handle. The tasks parts costs C_Parts_i (i=1..4) are calculated as the sums of the costs of concerned effective parts in each assembly task and shown in Table 25.

Table 25: The tasks involved parts costs [€]

C_Parts_i	*Involved effective parts*	C_Parts_i [€]
i=1	Seal & Receptacle	0.7+3= **3.7**
i=2	Cap & Seal & Receptacle	1.1+0.7+3= **4.8**
i=3	Stick & Handle	1.4+2.5= **3.9**
i=4	Handle & Receptacle	2.5+3= **5.5**

Concerning the duplication and renovation number l and r, we consider intervals of [1..10]. This interval can be limited or extended by strategic values *Max_dup* and *Max_ren*.

The matrix *T_Succ* defines the successors for each task of the model. According to the defined initial generic assembly process (Figure 29), the matrix is defined in following Table 26:

Table 26: The successors matrix

$T\_Succ_{i_1,i_2}$	i_2=1 (T1)	i_2=2 (T2)	i_2=3 (T3)	i_2=4 (T4)
i_1=1 (T1)	0	**1**	0	0
i_1=2 (T2)	0	0	0	**1**
i_1=3 (T3)	0	0	0	**1**
i_1=4 (T4)	0	0	0	0

Concerning the compatibility matrix *T_Comp*, we first assume the default case in which each task is compatible with all the others in all LoAs k. The resulting matrix (4x4x4) is then fully filled by "1" except the diagonal elements as shown in Table 27.

Table 27: The compatibilities matrix

$T\_Comp^k_{i_1,i_2}$	i_2=1 (T1)	i_2=2 (T2)	i_2=3 (T3)	i_2=4 (T4)
i_1=1 (T1)	[0,0,0,0]	[1,1,1,1]	[1,1,1,1]	[1,1,1,1]
i_1=2 (T2)	[1,1,1,1]	[0,0,0,0]	[1,1,1,1]	[1,1,1,1]
i_1=3 (T3)	[1,1,1,1]	[1,1,1,1]	[0,0,0,0]	[1,1,1,1]
i_1=4 (T4)	[1,1,1,1]	[1,1,1,1]	[1,1,1,1]	[0,0,0,0]

- **The retrieved data from standardized databases**

Considering the different tasks types and the characteristics of the parts to be assembled, corresponding tasks time estimates can then be determined in the different LoAs. The time estimates values are shown in Table 28.

Table 28: The tasks time estimates in the different LoAs [sec]

	LoA (index k)			
t_i^k	*k=1 (M)*	*k=2 (MT)*	*k=3 (A)*	*k=4 (R)*
***i*=1**	20	10	8	13
***i*=2**	50	15	20	33
***i*=3**	40	30	15	20
***i*=4**	60	50	30	40

As in the model 4 tasks exist (N=4), a maximum of 4 different resources, non-considering duplications to reach the productivity target, can be included in the configuration. This extreme case with maximal number of resources consists in assigning each of the 4 tasks independently to a dedicated resource that executes only this task. The number of resources candidates Max_R is then of 4 resources and equals to the number of tasks N. Else if the manufacturer imposes a different threshold, Max_R would be inferior to 4 and equal to the given value. Here, we consider the default case. Accordingly, the resource index *j* varies in the interval [1..4].

For the setup time depending on resource's LoA, we consider the values of Table 29.

Table 29: Resources setup time according to selectable LoAs [sec]

	LoA (index k)			
	k=1 (M)	*k=2 (MT)*	*k=3 (A)*	*k=4 (R)*
t_S^k	20	10	8	13

The resources labour rates C_L^k, basically related to manual resources, consequently LoA=1 and 2, is a vector defined in Table 30. As the presence of manual in automatic or robotic stations should be modelled as an independent resource assigned to operation that it achieves (even if feeding or supervision), we consider a null labour cost for automatic and robotic LoAs as it can be observed in Table 30. We assume a cost rate of 40€/Hour corresponding to 0.011€/sec.

Table 30: Resources labour cost rate [€/sec]

	LoA (index k)			
	k=1 (M)	*k=2 (MT)*	*k=3 (A)*	*k=4 (R)*
C_L^k	0.011	0.011	0	0

The setup cost rate C_SCP^k concerns all automation levels. This is related to the setup of batch productions. This setup consists in station preparation or organization for manual (LoA=1) and manual with automatic assistance (LoA=2), configurations for automatic (LoA=3), or re-programming for robotic (LoA=4) according to the next batch to be launched specificities. According to the LoA, the cost rate of a corresponding worker according to his skill may vary. In this study, we consider the cost rates values presented in Table 31.

Table 31: Resources setup cost rates [€/sec]

	LoA (index k)			
	k=1 (M)	*k=2 (MT)*	*k=3 (A)*	*k=4 (R)*
C_SCP^k	0.011	0.013	0.015	0.019

Another important parameter is the resource working life L_R^k that can impose renovation of the resource if the production life is too long compared to the resource working life. For manual we consider an infinite number. For manual with automated tool LoA, the working life is related to the automated tool life which is generally long. For automatic and robotic resources, equipment has a limited working life that can be predicted. We consider values shown in Table 32.

Table 32: Resources working life [Hours]

	LoA (index k)			
	k=1 (M)	k=2 (MT)	k=3 (A)	k=4 (R)
L_R^k	infinite	25 000	4 000	5 000

The other parameters that we consider as dependent to LoAs are quality parameters consisting in non-conformities rates NC_R^k and rejections rates RJ_R^k. The parameters values we consider are shown in Table 33.

Table 33: Resources non-conformities and rejection rates [%]

	LoA (index k)			
	k=1 (M)	k=2 (MT)	k=3 (A)	k=4 (R)
NC_R^k	7	2	3	2
RJ_R^k	3	1	2	1

Other resource parameters previously presented in the formulation depend on the tasks variant in addition to LoA: resources investment costs $C\_R_v^k$, resources required surface $S\_R_v^k$, and resources power consumption $P\_R_v^k$. According to the 5 tasks types vocabulary, we present in Table 34 the values we consider for the application example to be solved.

Table 34: The resources investment cost [K€], surface [m2], and power [KW]

	Vocabulary (index v)				
	v=1 Clamping	v=2 Placing	v=3 Riveting	v=4 Screwing	v=5 SnapFit.
$C\_R_v^k$	[3,5,10,50]	[1,6,10,30]	[2,5,20,40]	[1,4,9,25]	[2,3,30,45]
$S\_R_v^k$	[9,12,30,25]	[9,12,35,25]	[9,12,30,25]	[9,12,35,25]	[9,12,40,25]
$P\_R_v^k$	[0,0.5,6,8]	[0,0.6,8,5]	[0,0.8,7,10]	[0,0.5,7,6]	[0,0.5,6,10]

Concerning the unitary cost of a monthly occupied square meter surface (C_S) and the unitary consumed KWH energy cost (C_KWH), we consider in this study the following values:

C_S = 1 €/m^2

C_KWH = 0.152 €/KWH

- **The planned production strategic information**

We detail in this part the planned production strategic information input values. The automation possibilities matrix *T_Autom* provides the possibility to forbid automation options for given model's tasks. We assume first by default that all automation possibilities are allowed. The matrix is shown in Table 35. To forbid an automation option, the user has to put "0" in the cell that corresponds to the concerned task and the corresponding automation option to disallow.

Table 35: The automation possibilities matrix

	LoA (index k)			
$T\_Autom_i^k$	k=1 (M)	k=2 (MT)	k=3 (A)	k=4 (R)
i=1	1	1	1	1
i=2	1	1	1	1
i=3	1	1	1	1
i=4	1	1	1	1

The remaining planned production strategic information elements we consider are listed in Table 36.

Table 36: The strategic information parameters values

	Parameters values
t_Takt	$(NWdays\_Y * Nshift\_D * Ndura\_S)$ /V_An = **126.72** [sec]
$Altern$	0 [Boolean]
Max_dup	10 [Resources]
Max_ren	10 [Times]
Max_R	4 [Resources]
Max_TR	4 [Tasks]
$NWdays\_Y$	220 [Days]
$Nshift\_D$	1 [Shift/Day]
$Ndura\_S$	**8**[H/Shift]*3600=**28800**[sec/Shift]
P_L	48 [Months]
RwS	1 [Boolean]
V	200 000 [Unity] (= V_An*P_L/12)
V_An	50 000 [Unity/Year]
V_Bat	60 [Unity/Batch]

7.5.2. The case study resolution

According to the different information of this study, we analyse in this section the performances of the optimization model run on IBM CPLEX OPL 12.6 and the optimal solutions it allows to reach according to given input values. To validate and show the different possibilities and adaptation of the optimal solution according to the input data, we run the model on different cases: first, the reference case corresponding to the detailed input information of section 7.5.1. We show later the obtained optimal configuration taking into account all the presented problem information (case 1). Then, as we obtain a hybrid automation solution, we try in case 2 the possible full LoAs configurations that can be obtained to can compare the obtained costs to the ones of case 1 and consequently have proofs and justifications about the optimality of the case 1's solution. To perform the study of full LoAs, we perform then 4 sub-cases (case 2.1 to 2.4) corresponding to the 4 levels LoA scale we use in the method. Then, in case 3, we run the model with initial data of section 7.5.1 with a modification of resources working life in order to understand why 2 resources are used in the obtained optimal solution of case 1 while only one could ensure the required productivity. A last scenario is studied in case 4 allowing testing added incompatibilities between tasks and observing the impact on the optimal solution. These different scenarios may correspond to industrial realities that can be imposed by the manufacturer according to his preferences (case 2.1 to 2.4), related to the market and machines suppliers competitiveness that can allow finding more reliable machines with a same price (case 3), or caused by technical feasibilities that can cause incompatibilities between tasks and the non-possibility to run couples of tasks by a same resource for a given LoA (case 4). The execution results of the different case are presented in Table 37.

Table 37: Optimal assembly systems configurations for different application cases

Tasks	Res.	LoA	Res. time estim. [sec]	Dupli. Nb.	Res. takt-time [sec]	Resource workload [%]	Renov.	Res. surface $[m^2]$	Res. Investment [€]	Cost per prod per Res. [€]	Cost per product [€]
Case 1: Reference case											
T1	R1	LoA=4	66	1	66	52.08	0	25	50 000	0.394	**0.538**
T2											
T3											
T4	R2	LoA=3	30	1	30	23.67	0	35	9 000	0.142	
Case 2: Full LoA configurations imposed											
Case 2.1: Full manual											
T1	R1	LoA=1	150	2	75	59.18	0	18	6 000	2.24	**2.24**
T2											
T3											
T4											
Case 2.2: Full manual with automated assistance											
T1	R1	LoA=2	105	1	105	82.85	0	12	9 000	1.378	**1.378**
T2											
T3											
T4											
Case 2.3: Full automatic											
T1	R1	LoA=3	43	1	43	33.93	0	40	30 000	0.395	**0.54**
T2											
T3											
T4	R2	LoA=3	30	1	23	23.67	0	35	9 000	0.142	
Case 2.4: Full robotic											
T1	R1	LoA=4	66	1	66	52.08	0	25	50000	0.394	**0.58**
T2											
T3											
T4	R2	LoA=4	40	1	40	31.56	0	25	25000	0.184	
Case 3: Resources working life modified											
T1	R1	LoA=4	106	1	106	83.649	0	25	50 000	0.4412	**0.441**
T2											
T3											
T4											
Case 4: Incompatibilities between tasks added											
T1	R1	LoA=3	23	1	23	18.15	0	40	30 000	0.316	**0.55**
T3											
T2	R2	LoA=3	50	1	50	39.45	0	35	10 000	0.23	
T4											

Table 37 showing the solution results is organized as follows: column 1 shows the tasks references corresponding to Table 23 and Figure 29. Column 2 shows the assigned included resources in the obtained optimal configuration. Column 3 shows the LoA that corresponds to each resource. According to the assigned tasks and the LoAs, time estimates presented in Table 28 are retrieved and summed inside each resource and shown in column 4 first without consideration of the obtained resource duplication number of the optimal configuration. According to this information and to the computed takt-time, this resource duplication number is calculated by the optimizer so that the required productivity, expressed by the takt-time (t_Takt in Table 36), can be reached. The resources duplication numbers are stored in column 5.

The time estimate of column 4 divided by the duplication number gives the resource takt time that corresponds to its production cadence. This value is stored in column 6. The obtained value divided by the required takt time allows computing the resource occupation percentage or workload stored in column 7. The resource renovation number is computed respecting constraint (11) of the model and value is shown in column 8. The total surface for each resource considering the assigned tasks types surfaces (Table 34) and the duplication number is shown in column 9. The resources investment during the production life considering the duplications and renovation are stored in column 10. The cost per resource using the cost model implemented in the objective function is shown in column 11. Finally the sum through the different costs per resources gives the total assembly cost per product. This value is shown in column 12.

- **Case 1: The reference case**

This case corresponds to the input values described in section 7.5.1. In this case, all LoAs are possible for the different tasks (Table 35) and each task is compatible with all the others in the different LoAs (Table 27). We would like to highlight that the corresponding takt-time is of 126.72 sec (Table 36). This value is important and quite involved in the appropriate resources assignment and duplication number determination. The resource cadence (column 6) with obtained total tasks time divided by the obtained duplication number should be inferior to this value of takt-time ensured by the constraint (5) of the model. The results concerning the obtained optimal configuration using the proposal optimization model are shown in Table 37.

The obtained solution shows high automated resources that can be explained by the high volume and short production life (4 years). 2 resources are used: the first one is a robot (LoA=4) assigned to the 3 first tasks (T1, T2, and T3). The second resource is an automatic dedicated machine (LoA=3) assigned to task T4. The obtained cost per product is of 0.538€ with resources initial investments of 50 000€ and 9 000€. For the entered data of section 7.5.1 and with the considered objective of minimizing the cost per product, this solution is supposed to be the most advantageous among all possible other configurations. We try to check other cases and compare obtained costs in following analyses. In case 2, we focus on full LoAs.

- **Case 2: Full LoA configurations imposed**

As previously discussed, the aim here is to impose some choices, observe how the optimizer can adapt the solution and handle the new problem specifications, and compare and understand how the obtained solution of case 1 is better and justify its optimality. 4 cases are here studied according to the used 4 levels LoA scale: full manual (case 2.1), full manual with automated tool (case 2.2), full dedicated automatic machines (case 2.3), and full robotic (case 2.4).

- ***<u>Case 2.1: Full Manual</u>***

To obtain the optimal configuration corresponding to imposed full manual assembly, we simply forbid the other LoAs (LoA=2 to 4) for all the tasks. We modify then the automation possibilities matrix as shown in Table 38.

Table 38: The automation possibilities for case 2

	LoA (index k)			
$T\_Autom_i^k$	*k=1 (M)*	*k=2 (MT)*	*k=3 (A)*	*k=4 (R)*
i=1	**1**	0	0	0
i=2	**1**	0	0	0
i=3	**1**	0	0	0
i=4	**1**	0	0	0

The execution results are shown in Table 37 – Case 2.1. According to these results, it can be observed that duplication (x2) is required to reach the required productivity for full manual assembly. Then, even with no considerable investment (only 6000€ for the manual station), the cost per product of 2.24€ is far greater than the one obtained in the optimal configuration where all LoAs are allowed (case 1).

- ***Case 2.2: Full manual with automated tool***

We modify here the automation possibilities matrix by imposing the manual with automatic assistance (LoA=2) and we run the tool. The obtained configuration shown in Table 37 – Case 2.2 gives a cost of 1.37845€. Compared to the full manual, only one resource without duplication is sufficient to reach the required productivity because of the advantageous time estimates significantly lower with (LoA=2) than with manual LoA (LoA=1) (Table 28). The obtained cost is considerably lower than the one of full manual (2.24€), but higher than the optimal configuration of reference case 1 (0.538€). Yet, it can represent some advantages, such as the low surface required (18m2, versus 60m2 for case 1). Another advantage is that no considerable investments are needed (only 6K€ required, versus 59K€ for case 1). No crucial maintenance or stoppages because of failures can be observed in this configuration compared to more automated. The manual with automatic assistance can represent then an interesting alternative and a good deal for implementation.

- ***Case 2.3: Full automatic***

In this test, we forced the use of automatic LoA (LoA = 3) using the automation possibilities matrix. Then, we run the program and obtain the results of Table 37 – Case 2.3.

It can be observed that the cost is quite similar to the one of case 1 (0.54€ versus 0.538€). Identically, 2 resources are used, even if according to the workload, the resource could continue executing task T4. We realize that this can be explained by the required resource renovation because of the resources working life. In fact, if the resource executes all the tasks (resource execution time will increase), it will reach its planned working life and should be renovated before the end of the production. This causes additional costs. In order to use a unique resource, we test the program by modification of the resources working life of Table 32 after multiplication by 10. We obtain then unique resource executing all tasks.

- ***Case 2.4: Full robotic***

We impose here the robotic automation level (LoA = 4).This case allows understanding why in case 1 two resources are used: a robotic resource (R1) was assigned to T1-T3 and an automatic one (R2) assigned to T4 while R1 could continue to run the remaining T4 from productivity perspective. In fact, the total sum of time estimates for the 4 tasks in robotic is of 106 sec (Table 28, sum following column 5), consequently inferior to the takt-time target of 123.76 sec. This case aims at testing another extreme case: a full robotic system. Identically, we modify the automation possibilities matrix by imposing robotic for all tasks and forbidding the other options. Then, we run the program and obtain the results shown in Table 37 – Case 2.4.

According to the obtained performances, we can see that the results are quite similar to automatic, with a same architecture and a similar cost per product of 0.58€ instead of 0.54€. It can be assumed that the use of 2 resources is caused by a renovation required when using a same resource. This is obtainable with an objective function of minimizing the number of resources in the configuration. The number of renovations becomes equal to 1.

- **Case 3: Modified resources working life**

In this case, we consider different types of resources with longer working life. We keep the same initial costs for the resources. This can be interpreted as higher quality or reliability machines for a same cost, as with a more competitive machines supplier that can be found on the market. The initial resources working life values of Table 32 are here multiplied by a factor of 10. The new values are shown in Table 39.

Table 39: Resources modified working life [Hours]

	LoA (index k)			
	k=1 (M)	***k=2 (MT)***	***k=3 (A)***	***k=4 (R)***
L_R^k	*infinite*	250 000	40 000	50 000

The results are shown in Table 37 – Case 3. These results confirm the previous interpretations of the renovation requiring the need to a second resource. Here a same robotic resource is executing all the tasks instead of the use of a second resource. The obtained cost is also lower (0.441€). Yet, the initial input information is here modified and this new optimal configuration corresponds to new kinds of resources 10 times more reliable than initial ones.

- **Case 4: Incompatibilities between tasks added**

In this case, we reconsider here the initial resources working life values of Table 32. We add some incompatibilities between tasks in order to observe how the solution will move and how this modification will be handled in the optimal solution. We assume here that task T2 (clamping) is not compatible with task T3 (snap fitting) in the different LoAs for technical feasibility or because of required sequencing and architecture of the process. This means that these two tasks cannot be executed by a same resource. To consider this constraint, the compatibility matrix *T_Comp* of Table 27 should be updated. The update is shown in Table 40.

Table 40: The compatibilities matrix for case 5

$T\_Comp^k_{i_1,i_2}$	i_2*=1 (T1)*	i_2*=2 (T2)*	i_2*=3 (T3)*	i_2*=4 (T4)*
i_1***=1 (T1)***	[0,0,0,0]	[1,1,1,1]	[1,1,1,1]	[1,1,1,1]
i_1***=2 (T2)***	[1,1,1,1]	[0,0,0,0]	**[0,0,0,0]**	[1,1,1,1]
i_1***=3 (T3)***	[1,1,1,1]	**[0,0,0,0]**	[0,0,0,0]	[1,1,1,1]
i_1***=4 (T4)***	[1,1,1,1]	[1,1,1,1]	[1,1,1,1]	[0,0,0,0]

The results obtained in Table 37 – Case 4 shows that in this case, 2 automatic resources are used. This gives a cost per product of 0.55€. It can be observed that this new assignment satisfies the new incompatibilities constraints.

Following these validations, the suitability of the developed model can be considered as satisfactory. In next section 7.6, the paper is concluded with perspectives and outlooks proposals.

7.6. Conclusion

The proposed approach based on exhaustive search by alternatives generation and evaluation seems to be promising to support automation decision issue for multiple reasons. It provides an objective way to efficiently generate possible alternatives, using a mathematical program, by exploitation of optimized solvers algorithms, such as IBM OPL CPLEX, to iterate among relevant possibilities and guarantee the convergence to the optimal solution with consideration of manufacturers' preferences and exigencies. The optimization model allows the implementation of the different defined modules of the proposed LoA approach in chapter 3. It allows reaching the optimal automation configuration with the possibility to enter an ASML generic model of tasks, a consideration of LoA criteria with LoA possibilities allowing or forbidding using a matrix, appropriate resources assignment using time estimation databases, rules, balancing, and compatibilities, and finally, a cost model as an possibly the objective function of the model as defined in the model proposal and in the case study. The model seems consequently to address all the requirements for an efficient implementation and computerization of the proposed LoA decision method.

Yet, some improvements proposals can be addressed to the LoA decision approach methodology:

First, the dependency to a given fixed assembly sequence, here presented by the initial generic ASML model, can present some inconveniences (Rekiek, Dolgui, Delchambre, & Bratcu, 2002). In fact, the solution corresponds and depends on the considered initial sequence. Such initial sequence can lead to bad or low quality solutions. It can also eventually override interesting solutions. One perspective to tackle this issue can consist in generating all possible initial sequences then use the current tool to treat each of the sequences with exhaustive generation of all possible systems alternatives. The global optimal solution is then the optimal with regard to all initial sequences and all corresponding possible assembly systems to each of them. Ideally, this should be driven by CAD tools, or a module generating all possible sequences. The works of (Baldwin, Abell, Max Lui, De Fazio, & Whitney, 1991) and (Homem de Mello & Sanderson, 1991) can help on the topic of assembly sequences generation.

A second perspective can consist, once a module of sequences generation developed and associated, in an integrated tool of product-process development, eventually with user graphic interaction interface. The tool can be also connected to product design CAD tool to generate the optimal assembly process, and provide feedbacks at a same time to product designer to improve its design for the ease of assembly with integration of criteria related to assembly system design and LoA decision (e.g. assembly cost, ergonomics, or quality) during the product design phase as well.

Then, developing guidelines and rules to improve the product design to optimize the assembly process, especially when no satisfactory solution is found for the obtained optimal process corresponding to a given product design, without causing losses in performances by the eventual product design modification, represents another interesting research axis.

Finally, for the only 4 tasks of the example, the computational time is around 50 seconds with an i5-4200M CPU 2*2.5GHz processor and 8Go RAM processor. The problem is NP-hard and the computational time is exponential. To enable medium and high size problems solving, heuristics developing should represent a necessity to keep reasonable time. This can be proposed also as an interesting research opening in the topic of automation decision.

PART 3. THE CONTRIBUTIONS SUM-UP AND FUTURE WORKS PROPOSALS

Chapter 8. The contributions sum-up

Abstract

In this chapter, we summarize the research contributions presented through the thesis. The purpose is to provide a global overview of the new findings resulting from this work compared to existing research knowledge in the area of automation decision making and assembly systems design. The different contributions are summarized and briefly described through separate sections.

8.1. Introduction

Through previous parts of the manuscript, a review in automation decision making literature is presented in Part 1 where the topic was introduced and the research question was asked. The research question was: "How to decide about the most appropriate Level of Automation (LoA) when designing new assembly systems, and where to automate or not and to what extent throughout the process?". To answer to this question, a new approach was proposed by its outline at the end of Part 1. Different modules to be developed are identified to enable the proposed approach implementation. The modules concern: assembly modelling, time estimation with early phase balancing, cost estimation, and an optimization module to run automation alternatives generation and optimization. The different identified modules were tackled in Part 2 of the manuscript in separate chapters with a review on each corresponding topic. The purposes of the current part 3 are first to identify and summarize the different research contributions in the issue of automation decision in the current chapter 8. Then limitations and future works are proposed in the following last chapter 9 of this thesis.

We present then, through the current chapter, the different contributions in separate sections according to the main topic to which they belong. The identified research contributions are briefly described and summarized in the corresponding sections.

8.2. Contributions in LoA decision making methodologies

This thesis presents a state of the art enumerating all LoA decision methods we found in the literature. We show that current literature proposes detailed methodology for existing systems but the ones treating new systems are either vague or limited. By studying the reasoning employed for LoA decision, we stressed that an appropriate methodology should be analytic and objective, that it should consider manufacturer environment, and give a wide place to cost estimation. We provide a description of a new methodology allowing new systems analysis and handling a wide range of criteria. The method permits to handle assembly sequence, resources allocations, and associated automation levels.

8.3. Contributions in assembly early phase modelling

A review of assembly modelling methods used in assembly completed by languages used in other fields has been performed. The methods were analyzed from their suitability point of view to help automation decision making and support the description of assembly systems automation alternatives. The need to a new language was underlined. This led to the definition of a new modelling language labelled ASML as Assembly Systems Modelling Language. ASML associates graphic representation elements, rules, different structures of representing sequences, a time scale, and controlled vocabularies of assembly motions and tasks. The vocabulary of assembly motions was used from the literature to which we proposed an extension of additional missing motions. The list of motions represents a low layer vocabulary of assembly elementary motions. The vocabulary of tasks is a new proposed vocabulary resulting from a need to use a higher layer to define automation scenarios, and at a same time keeping connection with the lower layer vocabulary. The link between the 2 vocabularies was then defined. It ensures the possibility to move from a tasks model to a motions model layer. The language provides the possibility to represent a generic assembly process model of tasks based on a product design and an assembly sequence. Then, based on this representation, the language allows representing automation alternatives descriptions with resources assignment and automation levels association.

Contributions in assembly modelling can be then summed up in the following points:

- A review in assembly modelling for automation decision making issue.
- A modelling language proposal for early phase assembly processes representation and automation alternatives description.
- A high layer standardized vocabulary of assembly tasks.
- An extension of the motions vocabulary by a proposal of complementary motions.
- The connection between the 2 vocabularies to convert a tasks model to a motions model.

8.4. Contributions in early phase assembly time estimation

In early phase assembly time estimation topic, we presented available methods for time estimation, such as MTM or DFA. In these approaches, we identified interesting features consisting in time estimation databases based on product design features consideration involving assembly complexity and impacting time estimation criteria. We proposed a similar time estimation database with estimates of assembly motions adapted to ASML vocabulary and LoA scale. Time estimation rules according to the process structure were defined. They procedure allows to time estimate conceptual future processes. The representation can be directly in motions, or in tasks, as conversion of tasks to motions is possible as described in contributions of section 8.3. This time estimation was used to appropriately assign resources and schedule tasks. Based on the represented ASML process, obtained tasks estimates, and assigned resources, performance indicators are proposed. The indicators allow evaluating the suitability of the obtained configuration to the required productivity and adjusting the assignments using early phase balancing rules so that it can reach the required cadence. The approach is presented as a time estimation approach powered by ASML modelling and a use of the 2 controlled vocabularies.

Time estimation contributions to support LoA decision can be summarized as follows:

- A review in time estimation and time databases
- A time estimation procedure based on process ASML modelling.
- A time estimation database structure to support LoA decision issue.
- ASML time estimation rules to estimate assembly automation alternatives.
- Production performance indicators to evaluate an early phase assembly system concept with selected automation options with regard to the suitability to required productivity.
- Early phase balancing for assembly systems automation alternatives.

8.5. Contributions in cost estimation

In the field of cost estimation, an exhaustive review was performed of cost models and approaches that could support automation decision. Because of the specificities of the LoA decision issue, literature models are generally not applicable. This is basically caused by the early phase of the estimation imposing constraints of data non-availability, to the need of an analytic low granularity model with consideration of automation choices, and to a need of an objective approach with quantitative cost computation. This led to the integration of multiple cost drivers into a unique model justified by literature models. The model computes the assembly cost per product for a resource according to its LoA. It was then extended to handle multiple resources up to the whole ASML model. This allows to provide the total product assembly cost corresponding to the given automation configuration. In addition to the cost per product, cost performance indicators were proposed to allow evaluating and comparing different automation alternatives.

The cost estimation contributions for LoA decision can be summarized in:

- A literature review and analysis in cost estimation.
- An integrated literature-based early phase cost model proposal.
- Cost performance indicators definition to evaluate and compare automation alternatives.

8.6. Contributions in assembly systems design optimization

To computerize the whole approach and implement an optimization module for automation alternatives generation, evaluation, and optimal solution search, a review was first performed. As no method of the existing literature automation decision methods uses a computerized technique to search the optimal automation configuration solution, the review in optimization was performed in similar issues related to balancing, resource planning, and systems design optimization issues. The review provided multiple useful ideas and a resolution structures. A formulation as a mathematical integer programming model was then defined to solve the issue of automation decision in assembly systems design. The formulation allows the consideration of previous developments in assembly modelling as the generic model of tasks with precedencies and architectures consideration. Time estimation and balancing issues are involved in assigning tasks, grouping, and duplicating them according to the required cadence and to a compatibility input matrix (*T_Comp*) showing the compatibilities of each task with regard to the others in the different possible LoAs. The developed cost estimation model represents the objective function to be minimized. The decision criteria are considered by the mean of matrices allowing or forbidding each task in selectable LoA (*T_Autom*). The matrix has to be filled by analyses with regard to the selected decision criteria of all the model tasks with regard to decision criteria and in the different possible loAs according to the used scale.

The new findings in optimization to support automation decision can be summarized as:

- A review in optimization for assembly systems design and automation decision.
- An exact optimization resolution model proposal based on mathematical formulation with consideration of automation levels.
- Previous proposals in modelling, LoA criteria consideration, time estimation, early balancing, and cost estimation are integrated in the developed optimization model.

8.7. Conclusion

According to this summary of the performed contributions through this thesis, the complexity and multi-criteria aspect of the decision can be realized. In addition to the multi-criteria aspect characterizing the decision, the problem solving required multiple developments and different involved research fields. The problem can then be defined as a multi-disciplinary problem involving various competencies, and then stakeholders in companies, particularly when gathering data, so that the method can efficiently be run.

The different modules were identified in the proposed automation decision method at the end of chapter 3. The approach itself was initially built to address a list of requirements defined to describe the need in automation decision (chapter 3 – section 3.3.1).

To check the usefulness and how satisfactory the performed developments are, a quick backup to the initial requirements list can be interesting. This can allow us to verify how the obtained approach fulfills the requirement.

- The first requirement R1 about the applicability at the early phase is fulfilled as time and cost estimation are tailored to be early phase (R1 fulfilled).
- The approach is computerizable and objective (R2 fulfilled).
- The used reasoning is analytic as it used a low layer of granularity and analysis of product design parts and operations (R3 fulfilled).
- The approach allows partial automation representation in modelling, time, cost, and also optimization resolution (R4 fulfilled).
- Cost computation represents the target of the proposed cost model and represents the objective function to be minimized (R5 fulfilled).
- The manufacturer context, capabilities, and choices, and criteria are considered in the decision process basically by the analysis of tasks with regard to the criteria. This was also computerized in the optimization model by the use of an automation possibilities matrix input to be filled by analyses with regard to criteria. It can also include direct choices and preferences of the decider according to several reasons. (R6 fulfilled).
- Concerning the justification of the optimal solution and associated performance indicators reporting is ensured by the optimization mathematical model which consists in an exact resolution technique with a guarantee of optimality (R7 fulfilled).

The developed approach is then fulfilling with a satisfactory manner all the initial defined requirements. The research target seems then to be reached. The approach seems consequently to be promising. It proposes multiple contributions in different axes of the issue of LoA decision. Yet, limitations can be identified leading to several possible future works and openings. These limitations and future works are proposed in chapter 9.

Chapter 9. Limitations & future works proposals

Abstract

Through this chapter, we identify limitations of the proposed approach aiming at supporting assembly systems design and automation decision making. The proposal, thanks to multiple advantages and support it may offer, and also due to the identified limitations and gaps, represents a basis to several research openings and future works. We propose in this chapter ten identified ones that can be of interest for industrial applications and academia as well.

9.1. Introduction

The proposed approach supporting automation decision making seems to be promising. In previous chapter, different contributions in various research axes were identified. The method implementation is performed though modules separately studies and developed. The modules result from an initial definition of the research methodology decomposing the global decision approach to: process modelling, time estimation, cost estimation, and an optimization model. The last module allows the implementation of previous ones and an exact optimization resolution using a mathematical linear programming model. An evaluation of the obtained global approach with regard to the initially defined requirements, previously used to evaluate literature decision methods, was also performed in previous chapter. The assessment of the method with regard to the requirements fulfillment was successful and gave favorable and promising results as the method satisfied all the defined requirements. Yet, the question that can be asked may be: Is this a proof that the approach is perfect? The answer is evidently no. Every research has limitations, can be upgraded, extended, and should lead to new questions to solve more complex problems, customized problems, or have other direct or indirect applications. In next section of this chapter, we enumerate openings and future works that we propose as possible future developments, applications, or extensions of this work to address identified limitations of the defined approach.

9.2. More validations of the approach on large complex industrial case studies

The LoA decision approach was implemented through different steps because of the decomposition of the global approach to different modules. The validations were also consequently performed separately. Validation examples were different according to the need of the module to be validated on its features and specificities. For example, to validate the modelling approach with the use of tasks and conversion to motions, a simple example of few tasks was needed so that the lowest granularity can be obtained and shown. That is the reason why we used an example of only 3 tasks. For the cost model validation, a complex model of tasks was needed so that the validation and values can provide interesting conclusions that make sense and to show the ability of the model to convert the large extent of data to results and simple curves allowing analyzing and comparing different alternatives from profitability point of view. For the optimization model, and because of the high number of inputs and matrices that can be quickly increased for large size problems, we used a medium complexity problem to be pedagogically understandable and enough complex so that the resolution can be manually impossible in order to highlight the model usefulness. Consequently, for this proposal, we thought better to use different examples with adapted degrees of complexities to be coherent with the issue to be validated. At this stage, as all the modules are separately validated, there can be a need to validate the global approach with the different developed modules on a same complex industrial problem.

9.3. Multi-objective optimization

In the optimization model, we used the developed cost model as an objective function to be minimized. From another side, automation decision criteria can be filled using constant matrices after evaluation of the model tasks with regard to criteria that the decider wants to consider as significant in his decision (quality, ergonomics, etc). The evaluation has to be performed, for each task, and with regard to the different possible LoAs. The evaluation provides values as automation level, uing a given scale, can be favorable or not favorable for a given task. This reasoning, even if allows handling the multiple criteria, is mono-objective, as the only function to be optimized is the cost. The work can be extended to handle multi-objective optimization for combined objectives problems optimizing.

9.4. Scales for LoA criteria consideration

We mention in previous section 9.3 how the approach allows handling the LoA decision criteria. The procedure is to fill for each task if it can be carried or not with regard to a given criterion in a given LoA. The process should be iterated for all LoAs and all selected decision criteria. Consequently, if only one criterion is not favorable for a given task to be performed in a given LoA, this LoA will be automatically forbidden. We think this approach of considering the criteria can be enhanced by consideration of scales for each LoA instead of binary evaluation of favorable and not favorable. A scale of levels can be better to enhance it. We can propose for example a scale of 5 levels from 'not favorable' ('0') to 'excellent' ('4'). The complete scale with intermediate values can be for example: not favorable ('0'), poor ('1'), medium ('2'), good ('3'), and excellent ('4'). The task should then be forbidden in the given LoA only if it is not favorable ('0'). A score can be then computed to evaluate the LoA for each task as, for example, the product of the different obtained scores with regard to all the criteria. In this case if one of the values is '0' (not favorable), the final score of the task evaluation in the LoA will be '0'. Then the LoA with a best score can be kept as an objective function, or one through a multi-objective optimization.

9.5. Heuristics for large size problems

The current optimization approach is an exact resolution technique using a mathematical formulation as an integer programming model. As the problem is NP-hard, the computational time will increase exponentially with the problem size increase. Even with the combinatorial explosion reduction using the high layer vocabulary of tasks, we think the approach cannot be able to handle problems with important size or extremely high number of tasks. Heuristics will then be needed to handle large size problems.

9.6. Automatic generation of models and integration to LoA decision and DFA

In the current implementation of the approach, the initial ASML model of tasks is manually built. Then, this model is transposed to matrices and entered in the optimization model in the CPLEX solver. Currently, some product design tools, such as AutoCAD, allow an automatic generation of assembly operations. To automate the whole process computerization from product design tool to the assembly system design reaching with optimal automation levels, the first step can be to automatically translate the generated assembly operations by the CAD to the controlled vocabularies of tasks and/or motions. Then, ASML architecture can be defined. The LoA approach can then be run based on this obtained model. But, other features should be also automated to make the tool completely computerized directly from the product CAD tool, such as the compatibilities between tasks or the evaluation of tasks with regard to the selected LoA decision criteria. If this can work, it may provide a powerful integrated environment of assembly systems design and automation decision in one click from the product design CAD tool. This should considerably enhance practical implementation of DFA reasoning and principles. In fact, it can provide the possibility to re-design and upgrade the product design to decrease the ease or cost of the optimal process that can assemble it. Then, automatic re-design of the product using user interactive GUI (Graphic User Interface) interfaces in the 2 senses, from product design to assembly system and vice-versa, can be also tackled.

9.7. Automatic generation of all assembly sequences and corresponding models

The LoA decision approach is based on the initial ASML model which corresponds to a fixed assembly sequence. The proposed solution will then depend on this initial sequence. Changing the sequence to another one should provide other automation solution that can be better or worse. Consequently, to provide the optimal solution for a given product, all possible sequences should be generated. The automatic generation of sequences represents a research topic. Multiple works can help to develop this axis. Once an algorithm is available, the proposed approach can be run on each of the sequences. The optimal through all sub-optimals can be kept as a global optimal for a given product design.

9.8. Mixed assembly lines problems

One of the initial assumptions in the LoA problem solving was to handle unique product assembly on the line as a priority. The approach can be extended to handle mixed assembly lines problems where multiple variants of products can be assembled. In the developed approach, this can be handled by the analysis of tasks with regard to the flexibility criterion. This should authorize or not inflexible LoAs for given assembly tasks performing more than a variant of a product. We think this needs practical validations and possible improvements analysis.

9.9. Complex resources and other assembly structures

In this work, we consider a resource as a mean able to perform different assembly tasks. This resource can consist in a human beings as a manual LoA (LoA=1), manual with automated assistance (LoA=2), an automatic dedicated machine (LoA=3), or an industrial robot (LoA=4). In this approach, we assume a resource cannot execute more than a task at a given same time, by analogy to humans, but also to almost major existing machines. We define complex resources as resources capable to execute more than a task, or more than a motion, at an exactly same time, for example, get (P1) and get (P2) simultaneously. This can be valid for machines with multiple actuators or multi-arms robots. This context, for us particular, can be studied as an opening for future processes. Yet, in current implementation, it can be possible to represent them as different resources, or sub-resources that belong to a same main resource, because physically different actuators are used but associated to a same station. This topic can then be developed by updating or adapting the current developments; especially as a long-term future research eventually for more technologically advanced processes. This topic should be better discussed and clarified by discussions with manufacturers. Another topic that can be related may consist in structures and architectures of assembly lines handling, others than serial, parallel, and choice sequences, such as U-lines. Even these systems are rather organizational related than automation decision related, the impact of such structures on the decision making, on performances, or on LoA criteria can be more studied and analyzed.

9.10. Value of currency and uncertainties handling

In the defined cost estimation model, the estimation provides an assembly cost per product with consideration of multiple cost input parameters. The aim was to provide early phase estimation with consideration of as much as possible of cost drivers that can be available during the early phase. Then, other cost indicators that are of high importance for manufacturers in the automation decision making, according to discussions with our industrial partners, were proposed, such as initial investment, payback period, or the return on investment. Yet, one of important parameters to make the estimation more accurate, especially through the production life, which can be of several months or years, is the value of currency or value of money through the planned duration. The value can increase or decrease. The consideration of this parameter needs further research to use or define a stochastic model to predict the evolution of the currency through time. Once the parameter can be revealed as significantly influencing the decision, it can be integrated to the model. Another factor that can upgrade the model accuracy is the uncertainties of parameters that are currently considered as fixed. Error margins or probabilistic distributions of the parameters can be studied and eventually integrated.

9.11. Supporting geographical optimization and decision

Other possible extension of this work may be to integrate the possibility to handle optimizations with regard to different industrial localization. In the current implementation, parameters that depend on the location are fixed (such as labor rate, unitary cost of energy, of surface, etc). The extension of the work can start by the identification of the parameters that may vary on the geographical location. Then databases that depend on the country to be selected by the user can be built. This can be tested first for limited set of countries, such as for 5 countries (France, Germany, United-States, Japan, and China for example). Then, it can be interesting to compare, analyze, and find the reasons behind the results that can be found.

It should be also mentioned that the analyses with regard to criteria can differ according to the localization. Ergonomics as an issue for example can differ from a high labor rate country (more taken into account) to a low labor rate one, according to the local culture for example. If the analyses with regard to criteria would be automated, databases should be also upgraded. This can require multiple worldwide benchmarks and discussions. Such analyses can be interesting to evaluate the opportunity of a possible delocalization or keeping production in the country, particularly for high labor rate countries. This can provide a significant support to such important decision making. The studies can also provide additional signs or criteria that can drive new axes in the LoA decision making issue.

9.12. Conclusion

This research in automation decision making during the early phase of assembly systems design generates several perspectives and openings. Some of them can be driven in the short term. In this case results can be quickly obtained. Others seem to be more long-term researches and should take more time and effort to reach concrete and practical results. Almost major proposed developments require an alignment with the real industrial area and need continuous collaborations with industrial companies and stakeholders so that the developments can efficiently answer to their practical need. Such discussions and benchmarking are of a common interest for both of research academia and manufacturers. Yet, this would require more involvement of companies to facilitate collaborations so that experiments, data, and feedbacks from a first side; and research findings and results from the other side, can be quickly shared and enhanced. This can be facilitated only by reducing the gap between research and industry.

REFERENCES

Agnetis, A., & Arbib, C. (1997). Concurrent operations assignment and sequencing of particular assembly problems in flow lines. *Annals of Operations Research, 69*, 1-31.

Ağpak, K., & Gökçen, H. (2005). Assembly line balancing: Two resource constrained cases. *International Journal of Production Economics, 96*(1), 129-140.

Ahmad, F., Huang, H., & Wang, X. (2011). Analysis of the Petri Net Model of Parallel Manufacturing Processes with Shared Resources. *Information Sciences, 181*(23), 5249-5266.

Almannai, B., Greenough, R., & Kay, J. (2008). A Decision Support Tool Based on QFD and FMEA for the Selection of Manufacturing Automation Technologies. *Robotics and Computer-Integrated Manufacturing, 24*(4), 501-507.

Andrés, C., Miralles, C., & Pastor, R. (2008). Balancing and scheduling tasks in assembly lines with sequence-dependent setup times. *European Journal of Operational Research, 187*(3), 1212–1223.

Baldwin, D., Abell, T., Max Lui, M.-C., De Fazio, T., & Whitney, D. (1991). An Integrated Computer Aid for Generating and Evaluating Assembly Sequences for Mechanical Products. *IEEE Transactions on Robotics and Automation, 7*(1), 78-94.

Banerjee, A., & Banerjee, P. (2000). A behavioral scene graph for rule enforcement in interactive virtual assembly sequence planning. *Computers in Industry, 42*(2-3), 147-157.

Battaïa, O., & Dolgui, A. (2013). A taxonomy of line balancing problems and their solution approaches. *International Journal of Production Economics, 142*(2), 259-277.

Bauer, N., Huuck, R., Lukoschus, B., & Engell, S. (2004). Integration of Software Specification Techniques for Applications in Engineering. In *A Unifying Semantics for Sequential Function Charts* (pp. 400-418). Springer.

Baybars, I. (1986). A survey of exact algorithms for the simple assembly line balancing problem. *Management Science*(32), 909-932.

Bengtsson, K., Bergagård, P., Thorstensson, C., Lennartson, B., Åkesson, K., Yuan, C., et al. (2012). Sequence Planning Using Multiple and Coordinated Sequences of Operations. *IEEE Transactions on Automation Science and Engineering, 9*(2), 308-319.

Bernet, N., Wakeman, M., Bourban, P., & Månson, J. (2002). An integrated cost and consolidation model for commingled yarn based composites. *Composites Part A: Applied Science and Manufacturing, 33*(4), 495-506.

Bley, H., Reinhart, G., Seliger, G., Bernardi, M., & Korne, T. (2004). Appropriate Human Involvement in Assembly and Disassembly. *CIRP Annals - Manufacturing Technology, 53*(2), 487-509.

Boër, C., Pedrazzoli, P., Sacco, M., Rinaldi, R., De Pascale, G., & Avai, A. (2001). Integrated Computer Aided Design for Assembly Systems. *CIRP Annals - Manufacturing Technology, 50*(1), 17-20.

Boothroyd, G. (1987). Design for assembly—the key to design for manufacture. *The International Journal of Advanced Manufacturing Technology, 2*(3), 3-11.

Boothroyd, G. (2005). *Assembly Automation and Product Design.* Boca Raton, FL: CRC Press; Taylor & Francis Group.

Boothroyd, G., & Dewhurst, P. (1983). *Design For Assembly Handbook.* Amherst, MA: University of Massachussets, Department of Mechanical Engineering.

Boothroyd, G., Dewhurst, P., & Knight, W. (2011). *Product Design for Manufacture and Assembly – Third Edition.* Boca Raton, FL: CRC Press; Taylor & Francis Group.

Bornschlegl, M., Kreitlein, S., Bregulla, M., & Franke, J. (2015). A Method for Forecasting the Running Costs of Manufacturing. *Procedia CIRP - 12th Global Conference on Sustainable Manufacturing*, pp. 412-417.

Bowman, E. (1960). Assembly line balancing by linear programming. *Operations Research, 8*(3), 385-389.

Brainbridge, L. (1983). Ironies of automation. *Automatica, 19*(6), 775-779.

Bukchin, J., & Tzur, M. (2000). Design of flexible assembly line to minimize equipment cost. *IIE Transactions, 32*, 585-598.

Cavalieri, S., Maccarrone, P., & Pinto, R. (2004). Parametric vs. neural network models for the estimation of production costs: A case study in the automotive industry. *International Journal of Production Economics, 91*(2), 165-177.

Cecil, J., Srihari, K., & Emerson, C. (1992). A review of Petri Net Applications in manufacturing. *The International Journal of Advanced Manufacturing Technology, 7*(3), 168-177.

Chung, C. (1996). Human issues influencing the successful implementation of advanced manufacturing technology. *Journal of Engineering and Technology Management, 13*(3-4), 283-299.

Clark, W., & Gantt, H. (1923). *The Gantt Chart, a Working Tool of Management.* New York: Ronald Press.

Cleland, D., & King, W. (1988). *Project Management Handbook.* New York: Van Nostrand-Reinhold.

Colledani, M., Bolognese, L., Ceglarek, D., Franchini, F., Marine, C., & Mistry, A. (2015). Multi-objective Early-stage Design of Automotive Hybrid Assembly Lines. *Procedia CIRP. 28*, pp. 125-130. Naples: 3rd CIRP Global Web Conference - Production Engineering Research.

Colledani, M., Franchini, F., Micchetti, F., Ratti, A., & Taurisano, A. (2015). A software platform for the multi-objective early-stage design of automotive assembly lines. *15th IFAC Symposium onInformation Control Problems inManufacturing — INCOM 2015, 48*, pp. 2287-2292. Ottawa.

De Fazio, T., & Whitney, D. (1987). Simplified generation of all mechanical assembly sequences. *IEEE Journal on Robotics and Automation, 3*(6), 640-658.

Dewhurst, P., & Boothroyd, G. (1988). Early cost estimating in product design. *Journal of Manufacturing Systems, 7*(3), 183-191.

Dong, T., Tong, R., Zhang, L., & Dong, J. (2005). A Collaborative Approach to Assembly Sequence Planning. *Advanced Engineering Informatics, 19*(2), 155-168.

Downs, K., & Trappey, A. (1992). A sequential block approach to the design of a cost estimation system. *Computers & Industrial Engineering, 23*(1-4), 423-426.

Duverlie, P., & Castelain, J. (1999). Cost Estimation During Design Step: Parametric Method versus Case Based Reasoning Method. *The International Journal of Advanced Manufacturing Technology, 15*(12), 895–906.

Endsley, M. (1999). Level of automation effects on performance, situation awareness and workload in a dynamic control task. *Ergonomics, 42*, 462-492.

Endsley, M., & Kiris, E. (1995). The out-of-the-loop performance problem and level of control in automation. *Human Factors: The Journal of the Human Factors and Ergonomics Society, 37*(2), 381-394.

Fasth, Å., & Stahre, J. (2008). Does Level of Automation Need to Be Changed in an Assembly System? A Case Study. *Proceedings of the 2nd Swedish Production Symposium.* Stockholm, Sweden.

Fasth, Å., Provost, J., Fabian, M., Stahre, J., & Lennartson, B. (2012). From Task Allocation Towards Resource Allocation when Optimising Assembly Systems. *Procedia CIRP, 3*, pp. 400-405. Athens.

Fasth, Å., Stahre, J., & Dencker, K. (2008). Analysing changeability and time parameters due to levels of Automation in an assembly system. *Flexible Automation and Intelligent Manufacturing*, (pp. 700-707). Skövde, Sweden.

Fasth, Å., Stahre, J., & Dencker, K. (2008). Measuring and analysing Levels of Automation in an assembly system. *The 41st CIRP Conference on Manufacturing Systems*, (pp. 169-172). Tokyo; Japan.

Feldmann, K., & Slama, S. (2001). Highly flexible Assembly – Scope and Justification. *CIRP Annals - Manufacturing Technology, 50*(2), 489–498.

Feldmann, K., Müller, B., & Haselmann, T. (1999). Automated Assembly of Lightweight Automotive Components. *CIRP Annals - Manufacturing Technology, 48*(1), 9-12.

Frohm, J. (2008). *Levels of Automation in production systems.* Chalmers University of Technology. Göteborg: Department of Production System.

Frohm, J., Lindstrom, V., Winroth, M., & Stahre, J. (2006). The industry's view on automation in manufacturing. *Proceedings of the 9th Symposium IFAC on Automated Systems Based on Human Skills and Knowledge, 9*, 453-458.

Gary Teng, S.-H., & Garimella, S. (1998). Manufacturing cost modeling in printing wiring board assembly. *Journal of Manufacturing Systems, 17*(2), 87-96.

Ghosh, S., & Gagnon, R. (1989). A comprehensive literature review and analysis of the design, balancing and scheduling of assembly systems. *International Journal of Production Research, 27*, 637-670.

Gökċen, H., & Erel, E. (1998). Binary Integer Formulation for Mixed-Model Assembly Line Balancing Problem. *Computers & Industrial Engineering, 34*(2), 451–461.

Gorlach, I., & Wessel, O. (2008). Optimal Level of Automation in the Automotive Industry. *Engineering Letters, 16*(1), 141–149.

Granell, V. (2007). *Levels of automation in manufacturing systems: aligning strategic and tactical decisions by means of operational measurement.* Göteborg: Research Series from Chalmers University of Technology, Report no. 18.

Graves, S., & Holmes Redfield, C. (1988). Equipment selection and task assignment for multi-product assembly system design. *International Journal of Flexible Manufacturing Systems, 1*, 31-50.

Graves, S., & Lamar, B. (1983). An integer programming procedure for assembly system design problems. *Operations Research, 31*(3), 522-545.

Graves, S., & Whitney, D. (1979). A mathematical programming procedure for equipment selection and system evaluation in programmable assembly. *Proceedings of the IEEE Decision and Control*, 531-536.

Groover, M. (2015). *Automation, Production Systems, and Computer-integrated Manufacturing - 4th Edition.* London: Pearson.

Hill, T. (1999). *Manufacturing Strategy: Text and Cases, 3rd edition.* Homewood, IL: McGraw-Hill.

H'mida, F., Martin, P., & Vernadat, F. (2006). Cost estimation in mechanical production: The Cost Entity approach applied to integrated product engineering. *International Journal of Production Economics, 103*(1), 17-35.

Homem de Mello, L., & Sanderson, A. (1991). A Correct and Complete Algorithm for the Generation of Mechanical Assembly Sequences. *IEEE Transactions on Robotics and Automation, 7*(2), 228-240.

Homem De Morello, L., & Sanderson, A. (1990). AND/OR Graph Representation of Assembly Plans. *IEEE Transactions on Robotics and Automation, 6*(2), 188-199.

Homem De Morello, L., & Sanderson, A. (1991). Representation of Mechanical Assembly Sequences. *IEEE Transactions on Robotics and Automation, 7*(2), 221-227.

Hsieh, F. (2006). Robustness analysis of Petri nets for assembly/disassembly processes with unreliable resources. *Automatica, 42*(7), 1159-1166.

Hu, S., Zhu, X., Wang, H., & Koren, Y. (2008). Product variety and manufacturing complexity in assembly systems and supply chains. *CIRP Annals - Manufacturing Technology, 57*(1), 45-48.

Jackson, J. (1956). A computing procedure for the line balancing problem. *Management Science, 2*, 261-271.

Jha, N. (1992). Stochastic mathematical modelling and manufacturing cost estimation in uncertain industrial environment. *International Journal of Production Research, 30*(12), 2755-2771.

Jha, N. (1996). Probabilistic cost estimation in advance of production in a computerized manufacturing system through stochastic geometric programming. *Computers & Industrial Engineering, 30*(4), 809-821.

Jönsson, M., Andersson, C., & Ståhl, J.-E. (2011). Relations between volume flexibility and part cost in assembly lines. *Robotics and Computer-Integrated Manufacturing, 27*(4), 669–673.

Jung, J.-Y. (2002). Manufacturing cost estimation for machined parts based on manufacturing features. *Journal of Intelligent Manufacturing, 13*(4), 227-238.

Kapp, K. (1997). *The USA Principle.* APICS—The Performance Advantage.

Kelton, W., Sadowski, R., & Zupick, N. (2010). *Simulation with Arena – 5th Edition.* New York: McGraw-Hill.

Kolberg, D., & Zuehlke, D. (May 11-13, 2015). Lean Automation Enable by Industry 4.0 Technologies. *Proceedings of the 15th IFAC INCOM Symposium on Information Control Problems in Manufacturing.* Ottawa.

Konold, P., & Reger, H. (2003). *Praxis Der Montagetechnik [Practice the Assembly Technology]*. Wiesbaden: Vieweg Praxiswissen.

Koonce, D., Judd, R., Sormaz, D., & Masel, D. (2003). A hierarchical cost estimation tool. *Computers in Industry, 50*(3), 293-302.

Krüger, J., Nickolay, B., Heyer, P., & Seliger, G. (2005). Image based 3D Surveillance for flexible Man-Robot-Cooperation. *CIRP Annals - Manufacturing Technology, 54*(1), 19-22.

Lacouture, E. (2012). *Study to determine the optimal level of automation for the assembly.* Grenoble Institute of Technology.

Lay, G., & Schirrmeister, E. (2001). *Sackgasse Hochautomatisierung? Praxis Des Abbaus Von Overengineering in Der Produktion [Is High Automation an Impass? In Practice, Reduce Over-engineering in the Production]*. Karlsruhe: Fraunhofer Institute for Systems and Innovation Research (ISI).

Layer, A., Brinke, E., Van Houten, F., Kals, H., & Haasis, S. (2002). Recent and future trends in cost estimation. *International Journal of Computer Integrated Manufacturing, 15*(6), 499-510.

Lee, H., & Johnson, R. (1991). A line balancing strategy for designing flexible assembly systems. *International Journal of Flexible Manufacturing Systems, 3*, 91-120.

Lennartson, B., Bengtsson, K., Yuan, C., Andersson, K., Fabian, M., Falkman, P., et al. (2010). Sequence Planning for Integrated Product, Process and Automation Design. *IEEE Transactions on Automation Science and Engineering, 7*.

Liebers, A., & Kals, H. (1997). Cost Decision Support in Product Design. *CIRP Annals - Manufacturing Technology, 46*(1), 107-112.

Lindström, V. (2008). *Formulation of automation strategy in manufacturing systems – Developing a methodology for analysing and choosing levels of automation.* Chalmers University of Technology, PhD dissertation, Göteborg, Sweden.

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.

Lotter, B., & Wiendahl, H. P. (2006). *Montage in der Industriellen Produktion [Assembly in industrial production]*. Berlin: Springer.

Maynard, H., & Stegemerten, G. (1948). *Methods Time Measurements.* New York: McGRAW-Hill.

McMullen, P., & Frazier, G. (1998). Using simulated annealing to solve a multiobjective assembly line balancing problem with parallel stations. *International Journal of Production Research, 36*, 2717-2741.

Michalos, G., Fysikopoulos, A., Makris, S., Mourtzis, D., & Chryssolouris, G. (2015). Multi criteria assembly line design and configuration – An automotive case study. *CIRP Journal of Manufacturing Science and Technology, 9*, 69-87.

Michalos, G., Makris, S., & Mourtzis, D. (2012). An intelligent search algorithm-based method to derive assembly line design alternatives. *International Journal of Computer Integrated Manufacturing, 25*(3), 211-229.

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.

Mital, A., & Pennathur, A. (2002). Getting the Most out of Advanced Manufacturing Technology (AMT)-based Systems: Part I: Recognizing and Managing Technology Limitations. *International Journal of Technology and Management, 4*(1-2), 96-118.

Murata, T. (1989). Petri Nets: Properties, Analysis and Applications. *Proceedings of the IEEE, 77*(4), 541-580.

Needy, K., Billo, R., & Warner, R. (1998). A cost model for the evaluation of alternative cellular manufacturing configurations. *Computers & Industrial Engineering, 34*(1), 119-134.

Niazi, A., Dai, J. S., Balabani, S., & Seneviratne, L. (2005). Product cost estimation: Technique classification and methodology review. *Journal of Manufacturing Science and Engineering, 128*(2), 563-575.

Niu, X., Ding, H., & Xiong, Y. (1987). A hierarchical approach to generating precedence graphs for assembly planning. *International Journal of Machine Tools & Manufacture, 43*(14), 1473-1486.

Ostwald, P. (1983). *American Machinists Manufacturing Cost Estimating Guide.* New York: McGraw-Hill.

Ostwald, P. (1988). *Am Cost Estimator 4 th edition.* Cleveland, Ohio: Penton.

Ou-Yang, C., & Lin, T. (1997). Developing an Integrated Framework for Feature-Based Early Manufacturing Cost Estimation. *The International Journal of Advanced Manufacturing Technology, 13*(9), 618-629.

Pang, H., Fang, Z., Li, H., & Yang, X. (2011). Research and Application of Time Workflow Model Based on Timing Constraint Petri Nets. *Procedia Engineering, 15*, 1371-1376.

Parasuraman, R., Sheridan, T., & Wickens, C. (2000). A Model for Types and Levels of Human Interaction with Automation. *IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans, 30*(3), 286-297.

Pehrsson, L., Ng, A., & Bernedixen, J. (2011). *Multi-objective Production Systems Optimisation with Investment and Running Cost.* London: Springer.

Pehrsson, L., Ng, A., & Stockton, D. (2013). Industrial cost modelling and multi-objective optimisation for decision support in production systems development. *Computers & Industrial Engineering, 66*(4), 1036-1048.

Pianne, A. (2012). *Study to determine the optimum level of automation in assembly lines at ZF.* Grenoble Institute of Technology.

Ping, L., Yongtong, H., Bode, J., & Shouju, R. (1996). Multi-agent system for cost estimation. *Computers & Industrial Engineering, 31*(3-4), 731-735.

Quintana, G., & Ciurana, J. (2011). Cost estimation support tool for vertical high speed machines based on product characteristics and productivity requirements. *International Journal of Production Economics, 134*(1), 188-195.

Ramezanian, R., & Ezzatpanah, A. (2015). Modeling and solving multi-objective mixed-model assembly line balancing and worker assignment problem. *Computers & Industrial Engineering, 87*, 74-80.

Rekiek, B., & Delchambre, A. (1998). Ordering variants and simulation in multiproduct assembly lines. *Proceedings of VR-Mech'98*, (pp. 49-54). Brussels.

Rekiek, B., Dolgui, A., Delchambre, A., & Bratcu, A. (2002). State of art of optimization methods for assembly line design. *Annual Reviews in Control, 26*(2), 163-174.

Renu, R., Mocko, G., & Koneru, A. (2013). Use of Big Data and Knowledge Discovery to Create Data. *Procedia Computer Science, 20*, 446–453.

Ross, P. (2002). *Bestimmung Der Wirtschaftlichen Automatisierungsgrades Von Montageprozessen in Der frühen Phase Der Montageplanung [Determining the Degree of Automation of Assembly Processes in the Early Stage of Assembly Planning].* Munich: IWB Forschungsberichte N°170.

Säfsten, K., Winroth, M., & Stahre, J. (2007). The content and process of automation strategies. *International Journal of Production Economics, 110*(1-2), 25-38.

Salmi, A. (2013). *Deciding the Level of Automation for the Design of Assembly Manufacturing Systems.* Grenoble Institute of Technology.

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 Symposium on Computer & Industrial Engineering, October 28-30, 2015.* Metz.

Salmi, A., David, P., Blanco, E., & Summers, J. (2016). A review of cost estimation models for determining assembly automation level. *Computers & Industrial Engineering, 98*, 246-259.

Salmi, A., David, P., Blanco, E., & Summers, J. (2016). Standardized vocabularies for assembly systems modelling and automation alternatives description. *Proceedings of the ASME 2016 International Design & Engineering Technical Conference.* Charlotte.

Salmi, A., David, P., Blanco, E., & Summers, J. (In submission). An integrated cost estimation approach to support automation decision in assembly systems design.

Salmi, A., David, P., Blanco, E., Summers, J., & Briant, O. (Under review). An Optimization Model to Support Early phase Assembly Systems Design & Automation Decision. *Computers & Industrial Engineering*.

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.

Salmi, A., Dhulia, J., Summers, J., David, P., & Blanco, E. (2015). Methods for Selecting Level of Automation: A Critical Comparison of Approaches and Integrated Proposal. *Proceedings of the ASME 2015 International Design & Engineering Technical Conference.* Boston.

Salveson, M. (1955). The assembly line balancing problem. *Journal of Industrial Engineering, 6*(3), 18-25.

Satchell, P. (1998). *Innovation and Automation.* Aldershot; UK: Ashgate.

Saunders, K. (2004). Automation and robotics in ADME screening. *Drug Discovery Today: Technologies, 1*(4), 373-380.

Scholz-Reiter, B., & Freitag, M. (2007). Autonomous Processes in Assembly System. *CIRP Annals, 56*(2), 712-729.

Shehab, E., & Abdalla, H. (2002). A design to cost system for innovative product development. *Journal of Engineering Manufacture, 216*(7), 999-1019.

Sheridan, T. (1995). Human centered automation: oxymoron or common sense? *IEEE International Conference on Systems Man and Cybernetics*, (pp. 823-828). Vancouver; Canada.

Sheridan, T. (1997). Task analysis, task allocation and supervisory control. In M. Helander, T. Landauer, & P. Prabhu, *Handbook of Human-Computer Interaction, Second Edition* (pp. 87-105). Amsterdam: Elsevier.

Sheridan, T. (2002). *Humans and Automation: System Design and Research Issues.* New York; USA: John Wiley & Sons.

Sheridan, T., & Verplanck, W. (1978). *Human and computer control of undersea teleoperators.* Cambridge, MA, USA: Technical report, MIT Man-Machine Laboratory.

Sim, K. (2001). An empirical examination of successive incremental improvement techniques and investment in manufacturing technology. *International Journal of Operations & Production Management, 21*(3), 373-399.

Slack, N., & Lewis, M. (2002). *Operations Strategy.* Harlow: Prentice Hall.

Son, Y. (1991). A cost estimation model for advanced manufacturing systems. *International Journal of Production Research, 29*(3), 441-452.

Srihari, K., & Emerson, C. (1990). Petri Nets in Dynamic Process Planning. *Computers & Industrial Engineering, 19*, 447-451.

Stewart, R. (1991). *Cost Estimating, second edition.* New York: Wiley-Interscience.

Swift, K., & Booker, J. (2013). *Manufacturing process selection handbook.* Waltham, MA: Elsevier.

Tuncel, G., & Topaloglu, S. (2013). Assembly line balancing with positional constraints, task assignment restrictions and station paralleling: A case in an electronics company. *Computers & Industrial Engineering, 64*(2), 602-609.

Udo, G., & Ebiefung, A. (1999). Human factors affecting the success of advanced manufacturing systems. *Computers & Industrial Engineering, 37*(1-2), 297-300.

Wickens, C., Lee, J., Liu, Y., & Gordon-Becker, S. (2014). *An Introduction to Human Factors Engineering - 2nd Edition.* New York: Pearson.

Windmark, C., Gabrielson, P., Andersson, C., & StŒhl, J. E. (2012). A Cost Model for Determining an Optimal Automation Level in Discrete Batch Manufacturing. *Procedia CIRP*, *3*, 73-78.

Winroth, M., Säfsten, K., Stahre, J., Granell, V., & Frohm, J. (2007). Strategic automation—refinement of classical manufacturing strategy. *Proceedings of the Swedish Production Symposium.* Göteborg.

Wong, J., Imam, I., Khosravi-Kamrani, A., Parsaei, H., & Tayyari, F. (1992). *A Totally Integrated Manufacturing Cost Estimating System (TIMCES). In: Parsaei, H. R.; Mital, A., Economics of Advanced Manufacturing Systems* . NJ: Chapman & Hall.

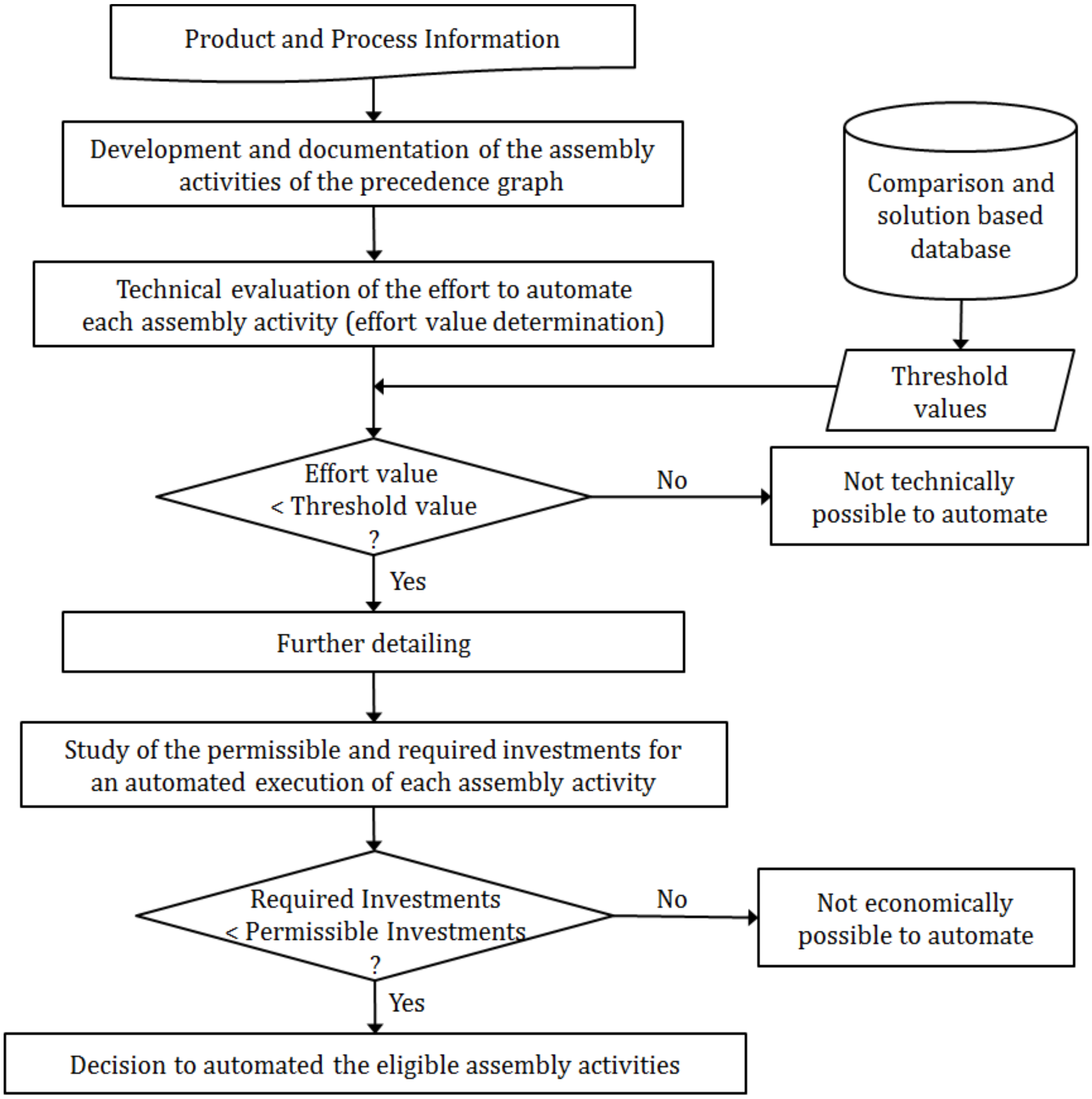
Zha, X., Du, H., & Lim, Y. (2001). Knowledge Intensive Petri Net Framework for Concurrent Intelligent Design of Automatic Assembly Systems. *Robotics and Computer-Integrated Manufacturing, 17*(5), 379-398.

Zha, X., Lim, Y., & Fok, S. (1998). Integrated knowledge-based assembly sequence planning. *The International Journal of Advanced Manufacturing Technology, 14*(1), 50-64.

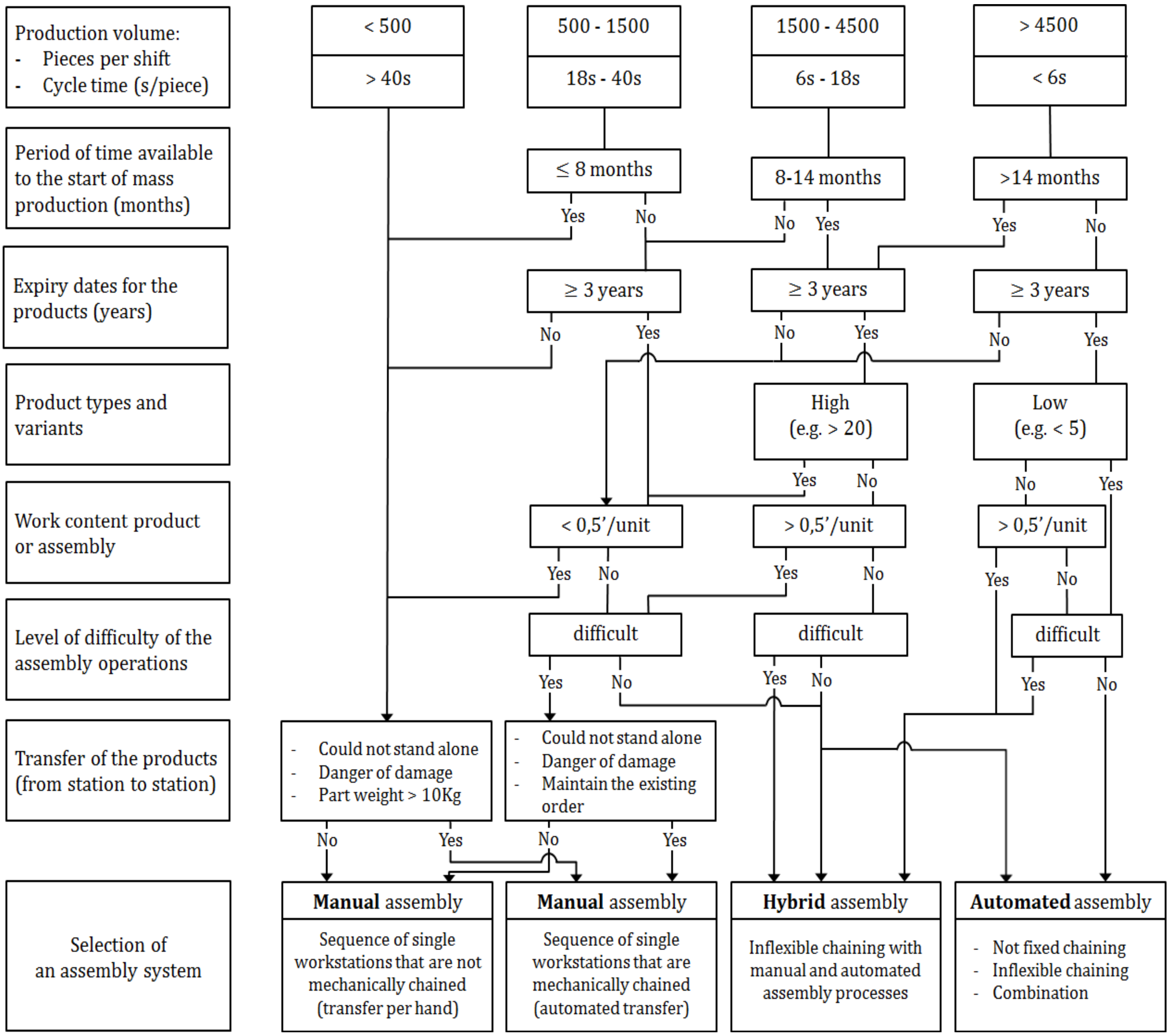
Zhang, W., Freiheit, T., & Yang, H. (2005). Dynamic Scheduling in Flexible Assembly System Based on Timed Petri Nets Model. *Robotics and Computer-Integrated Manufacturing, 21*(6), 550-558.

APPENDICES

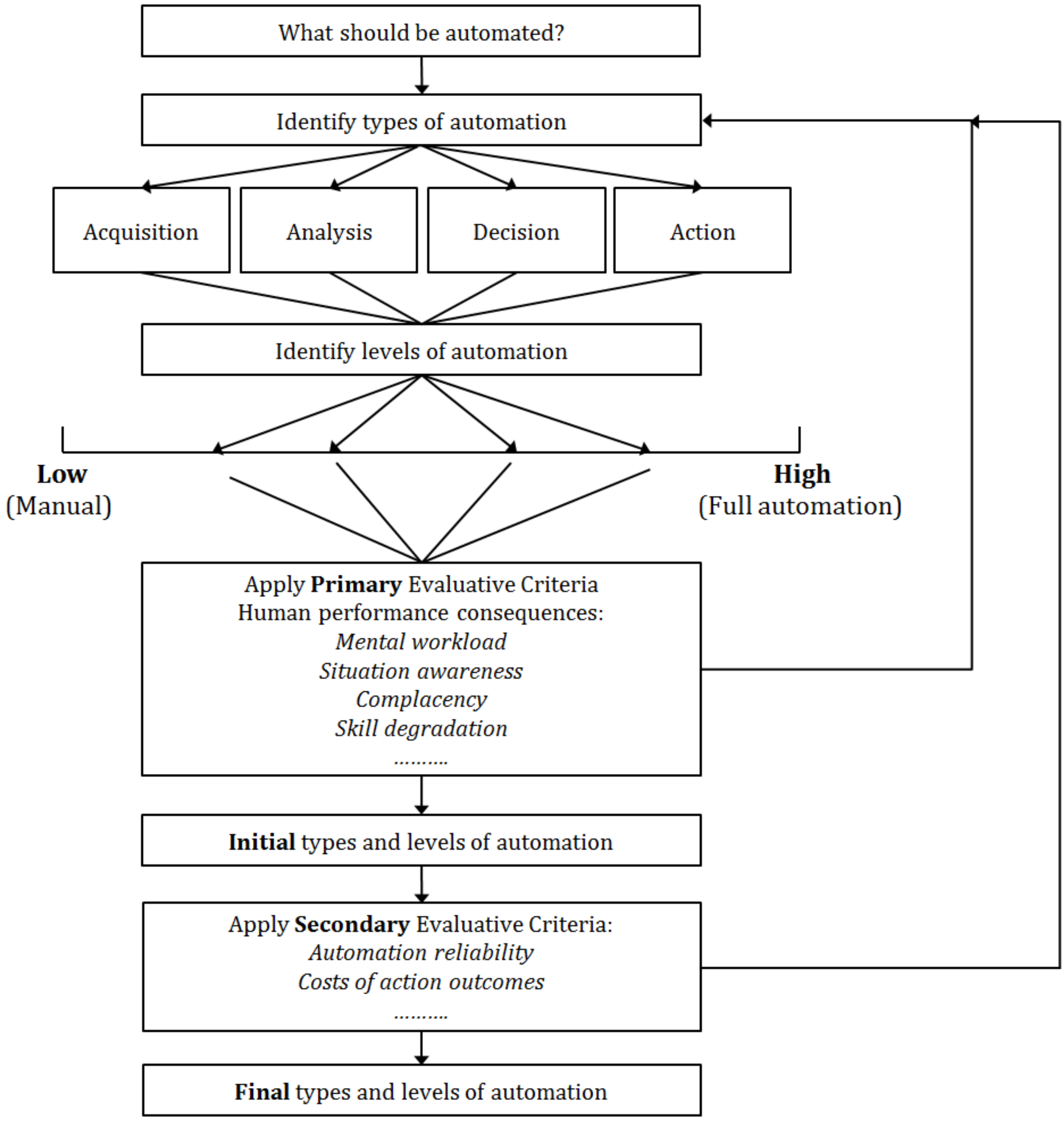
Annex A. Automation Decision Literature Methods Outlines



Annex A - Figure 1: The method M1 (Ross, 2002) description



Annex A - Figure 2: The method M2 (Konold & Reger, 2003) description



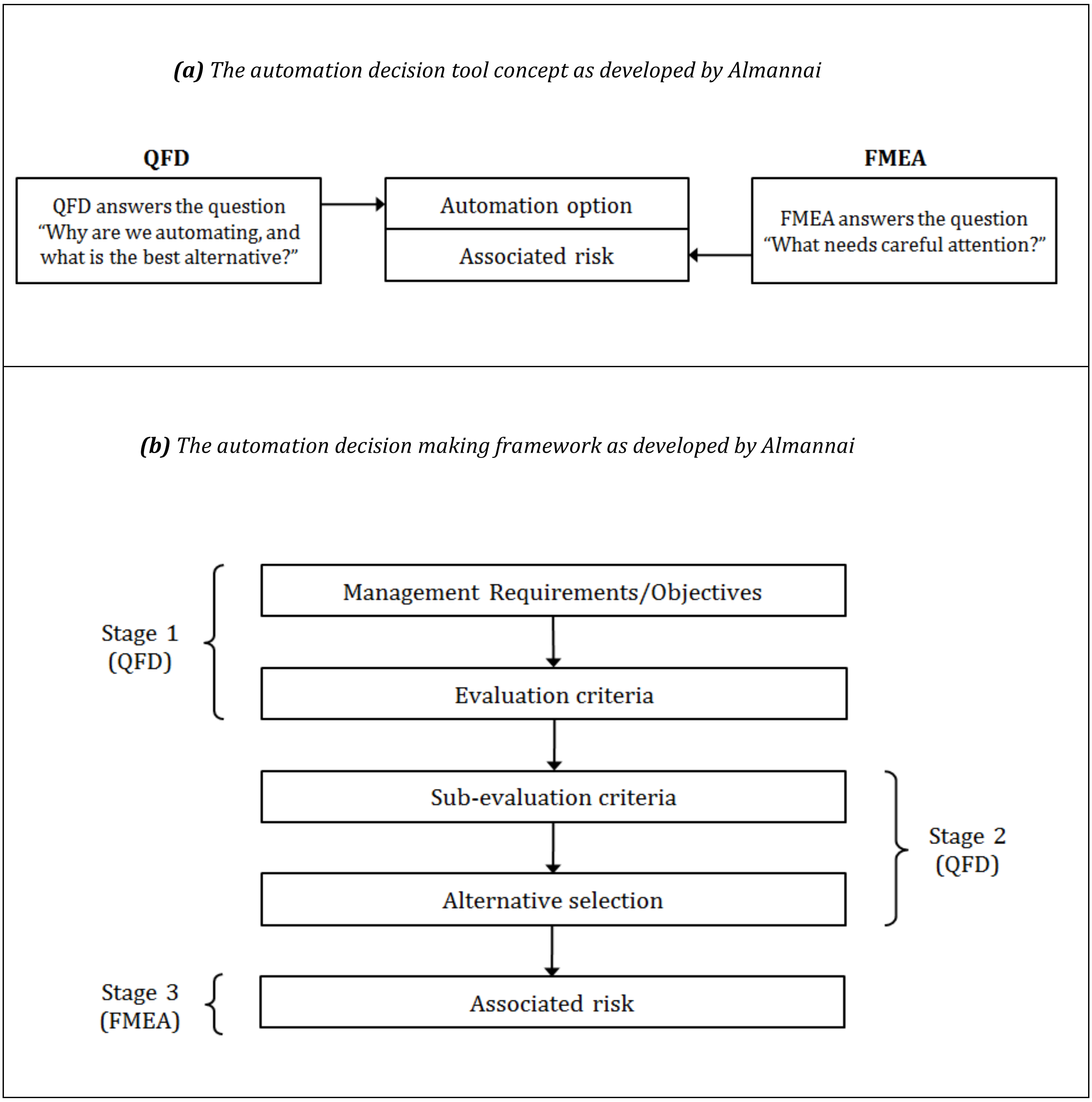
Annex A - Figure 3: The method M4 (Parasuraman & Sheridan, 2000) description

Step	*Description*
1	Plan ahead before the measurement
2	On site, start with a pre-study to identify the process
3	Visualize and document the production flow
4	Identify the main task for each section/cell
5	Identify the sub-tasks for each section/cell
6	Measure LoA
7	Assess LoA, set relevant max and min levels
8	Results analysis

Annex A - Figure 4: The DYNAMO method (M5) (Lindström & Winroth, 2010) description

Step	*Description*
	Pre-Study
1	Identify the system to improve onsite
2	Walk the process
3	Identify flow and time parameters by Value Stream Mapping (VSM) building
	Measurement
4	Identify the main operations and subtasks for selected area by Hierarchical Tasks Analysis (HTA) designing
5	Measure LoA using the LoA mechanical and information scales
6	Results documentation
	Analysis
7	Decide min and max LoA for the different tasks by Workshop
8	Design Square of Possible Improvements (SoPI) based on workshop results
9	SoPI analysis
	Implementation
10	Write / visualize the suggestions of improvements
11	Implementation of the decision suggestions
12	Follow-up when the suggestions have been implemented and analyses their effects on time and flow

Annex A - Figure 5: The DYNAMO++ method (M6) (Fasth & Stahre, 2008) description



Annex A - Figure 6: The method M7 (Almannai, Greenough, & Kay, 2008) description

			NP = 1								A	B
			(NT < 1.5 NA) U (ND < 0.5 NA)				(NT >= 1.5 NA) ∩ (ND >= 0.5 NA)					
			RI>=5	5>RI>2	2>=RI>=1	1>RI	RI>=5	5>RI>2	2>=RI>=1	1>RI		
			0	1	2	3	4	5	6	7	8	9
VS> 0.65	NA=>16	0	AF	AF	AF	MM (AF)	AP	AP	AP (MM)	MM	MA (AP)	MA
	16>NA=>7	1	AF	AF (AI)	AI (AF)	MM (AI)	AP	AP	MM (AP)	MM	MA	MA
	6>NA	2	AI	AI	AI	AI	AI	AI (AP)	MM	MM	MA	MA
0.65>=V S>0.4	NA=>16	3	AP	AP	MM (AP)	MM	AP	AP	AP	MA (MM)	MA	MA
	16>NA=>7	4	AI	AI	AI	MM	AP	AP	MM (AP)	MA (MM)	MA	MA
	6>NA	5	AI	AI	MM (AI)	MM	AI (MM)	MM	MM	MA (MM)	MA	MA
0.4>=VS >0.2	NA=>16	6	AP	AP	MM	MM	AP	AP	AP	MA	MA	MA
	16>NA=>7	7	AI (MM)	MM	MM	MM	AP	MM	MA (MM)	MA	MA	MA
	6>NA	8	MM	MM	MM	MM	MM	MM	MA (MM)	MA	MA	MA
VS <= 0.2		9	MM	MM	MM (MA)	MA	MM	MA	MA	MA	MA	MA

ND	Number of parts whose design changes during the first 3 years (major changes, e.g imposing new feeders/workheads)	**NP**	Number of different products to be assembled using the same basic system during the first three years
VS	Annual production volume per shift	**NT**	Total number of parts available for building different product styles
A	A variety of different but similar products	**NA**	Number of parts in the complete assembly
B	A variety of different products	**RI**	Company Investment potential or Investment ratio
MA	A worker with a transfer device	**MM**	A worker with mechanical assistance
AI	An indexing machine	**AF**	A free transfer machine
AP	Programmable assembly machine	**AR**	A two arm robot (AP in case of a unique station needed)

Annex A - Figure 7: The method M8 (Boothroyd & Dewhurst, 1983) description

Annex B. *Cost estimation literature for automation decision: models & classifications*

Annex B – Table 1: Literature cost estimation approaches classification (Salmi A. , David, Blanco, & Summers, 2016)

Model N°	Literature cost approach reference	A/M*	Approach type (I)	Granularity level (II)	Applicability Phase (III)
1	(Windmark, et al. 2012)	*M*	*Parametric*	*Bottom-up*	*Late-phase*
2	(Gorlach et Wessel 2008)	*A*	*Analytical*	*Bottom-up*	*Late-phase*
3	(Jönsson, Andersson et Ståhl 2011)	*A*	*Parametric*	*Bottom-up*	*Late-phase*
4	(Dewhurst et Boothroyd 1988)	*M*	*Parametric*	*Bottom-up*	*Early-phase*
5	(Zhang et Fuh 1998)	*M*	*Analogical*	*Top-down*	*Early-phase*
6	(Jung 2002)	*M*	*Parametric*	*Bottom-up*	*Early-phase*
7	(Son 1991)	*AM*	*Parametric*	*Bottom-up*	*Early-phase*
8	(P. Ostwald 1988)	*M*	*Parametric*	*Top-down*	*Early-phase*
9	(Gary Teng et Garimella 1998)	*A*	*Analytical*	*Bottom-up*	*Late-phase*
10	(Boothroyd 1984)	*A*	*Parametric*	*Top-down*	*Early-phase*
11-12	(Jahan-Shahi, Shayan et Masood 1999) (Jahan-Shahi, Shayan et Masood 1998)	*M*	*Intuitive*	*Bottom-up*	*Late-phase*
13	(Pehrsson, Ng et Stockton 2013)	*M*	*Analogical*	*Top-down*	*Early-phase*
14	(Pehrsson, Ng et Bernedixen 2011)	*M*	*Analytical*	*Bottom-up*	*Early-phase*
15	(Cavalieri, Maccarrone et Pinto 2004)	*M*	*Analogical*	*Top-down*	*Early-phase*
16-17	(Jha 1996) (Jha 1992)	*M*	*Analytical*	*Bottom-up*	*Early-phase*
18	(H'mida, Martin et Vernadat 2006)	*M*	*Analytical*	*Bottom-up*	*Early-phase*
19	(Boothroyd, Dewhurst et Knight 2011)	*A*	*Parametric*	*Bottom-up*	*Early-phase*
20	(Creese, Adithan et Pabla 1992)	*M*	*Parametric*	*Top-down*	*Early-phase*
21	(Bernet, et al. 2002)	*M*	*Analytical*	*Bottom-up*	*Late-phase*
22	(Zhang, Fuh et Chan 1996)	*M*	*Analogical*	*Bottom-up*	*Early-phase*
23	(Turunen, Järveläinen et Dohnal 1984)	*M*	*Intuitive*	*Top-down*	*Early-phase*
24	(Eklin, Arzi et Shtub 2009)	*M*	*Parametric*	*Bottom-up*	*Late-phase*
25	(Downs et Trappey 1992)	*M*	*Analytical*	*Top-down*	*Late-phase*
26	(Quintana et Ciurana 2011)	*M*	*Parametric*	*Bottom-up*	*Early-phase*
27	(Gayretli et Abdalla 1999)	*M*	*Analytical*	*Top-down*	*Early-phase*
28	(Tosun, Turhan et Bener 2009)	*M*	*Analogical*	*Top-down*	*Early-phase*
29-30	(Shehab et Abdalla 2002) (Ou-Yang et Lin 1997)	*M*	*Analytical*	*Bottom-up*	*Early-phase*
31	(Swift et Booker 2013)	*AM*	*Parametric*	*Bottom-up*	*Early-phase*
32	(Bornschlegl, et al. 2015)	*AM*	*Parametric*	*Top-down*	*Early-phase*

** **A**: The cost model is dedicated to **Assembly** / **M**: The model is dedicated to **Manufacturing** / **AM:** The model(s) can be used for both of **A** and **M***

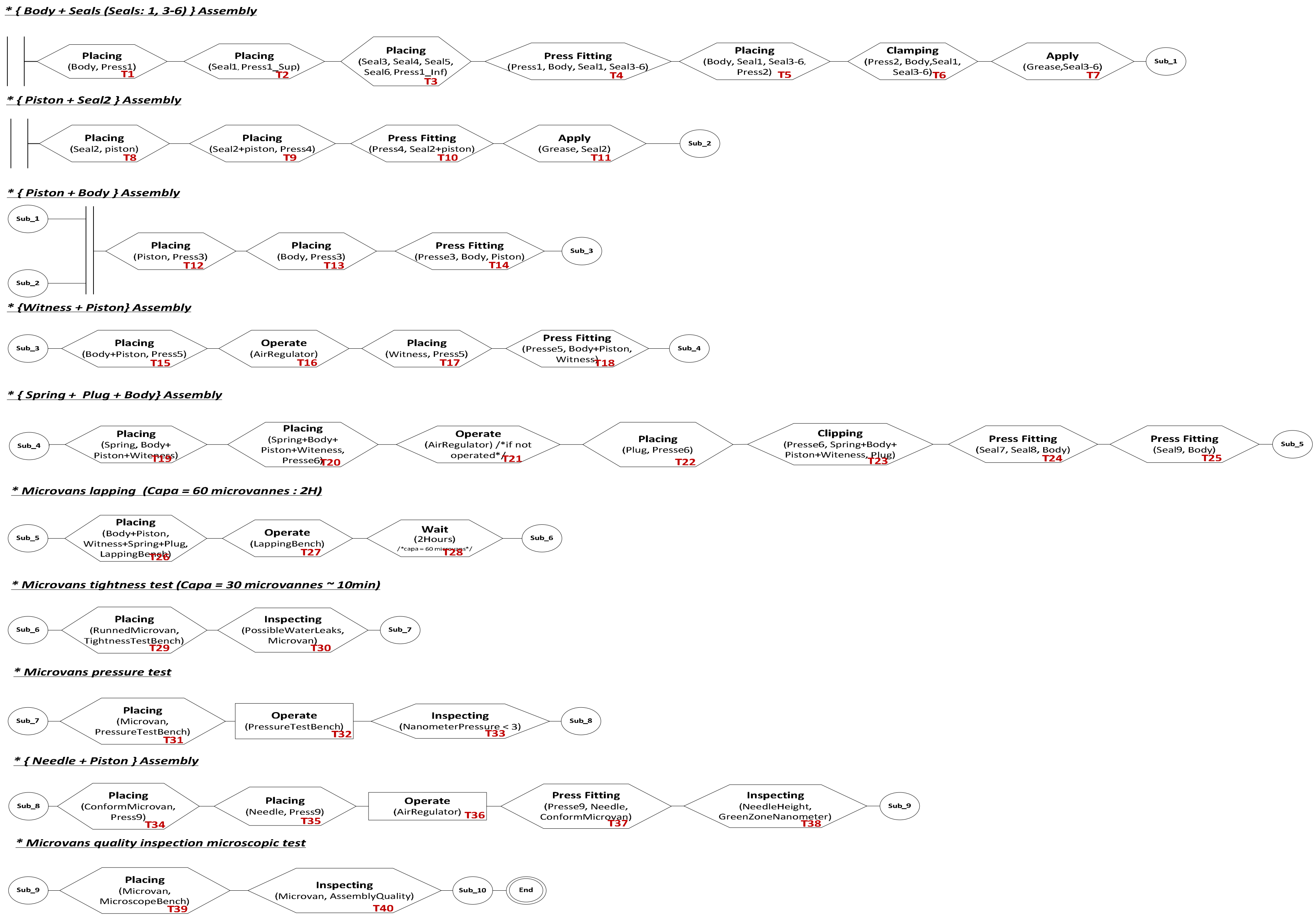
Annex C. Cost estimation case study input information

Annex C – Table 1: The planned production strategic input information

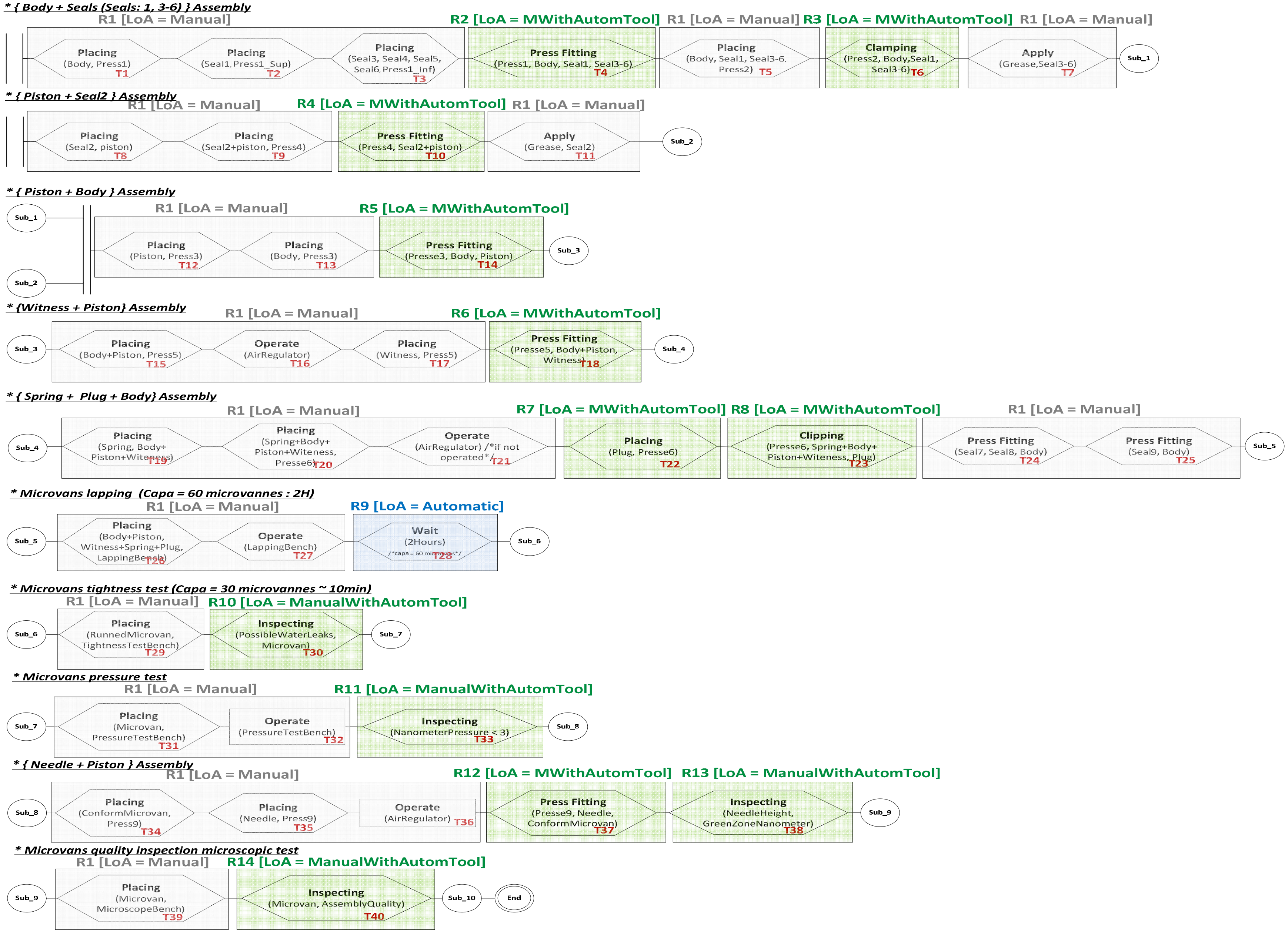
Parameter		Value	Description/ Equation
YearlyWrkD		230 [Days]	P7
StatHolid		20 [Days]	P8
DailyWrkH1S		8 [Hours]	P9
YearlyWrkDur1W1S		1680 [Hours] = 6048.10^3 [sec]	P6, 8
ProdLife		6 [Years] = 72 [Months]	P18
YearlyWrkMonths		12 [Months/Year]	P16
Volume	*YearlyVolume*	30 000 [Microvans/Year]	P15, 12
	TotalProdVolume	180 000 [Microvans]	P11
	MonthlyVolume	2500 [Microvans/Month]	P17
NbShiftsDay		1 [Shift/Day]	P14
$T_{Req\_TaktTime}$		201,6 [Sec]	P13, 11
C_{KWH}		0.152 [€/KWH]	P25
HourlyStopProdCost		1000 [€/Hour]	P31
$C_{MonthlySqrMeter}$		9 [€/m^2/Month]	P36

Annex C – Table 2: The parts to be assembled obtaining costs

Part	*Part label*	*Part obtaining cost* [€]
Body	BD	6,65
Piston	PS	11,5
Spring	SP	4,35
Plug	PL	2,5
Witness	WTN	2,1
Needle	NDL	4,5
Seal1	SL1	3,45
Seal2	SL2	3,17
Seal3	SL3	3,38
Seal4	SL4	3,6
Seal5	SL5	3,58
Seal6	SL6	3,1
Seal7	SL7	2,85
Seal8	SL8	3,15
Seal9	SL9	2,5



Annex C – Figure 1: The assembly generic process model



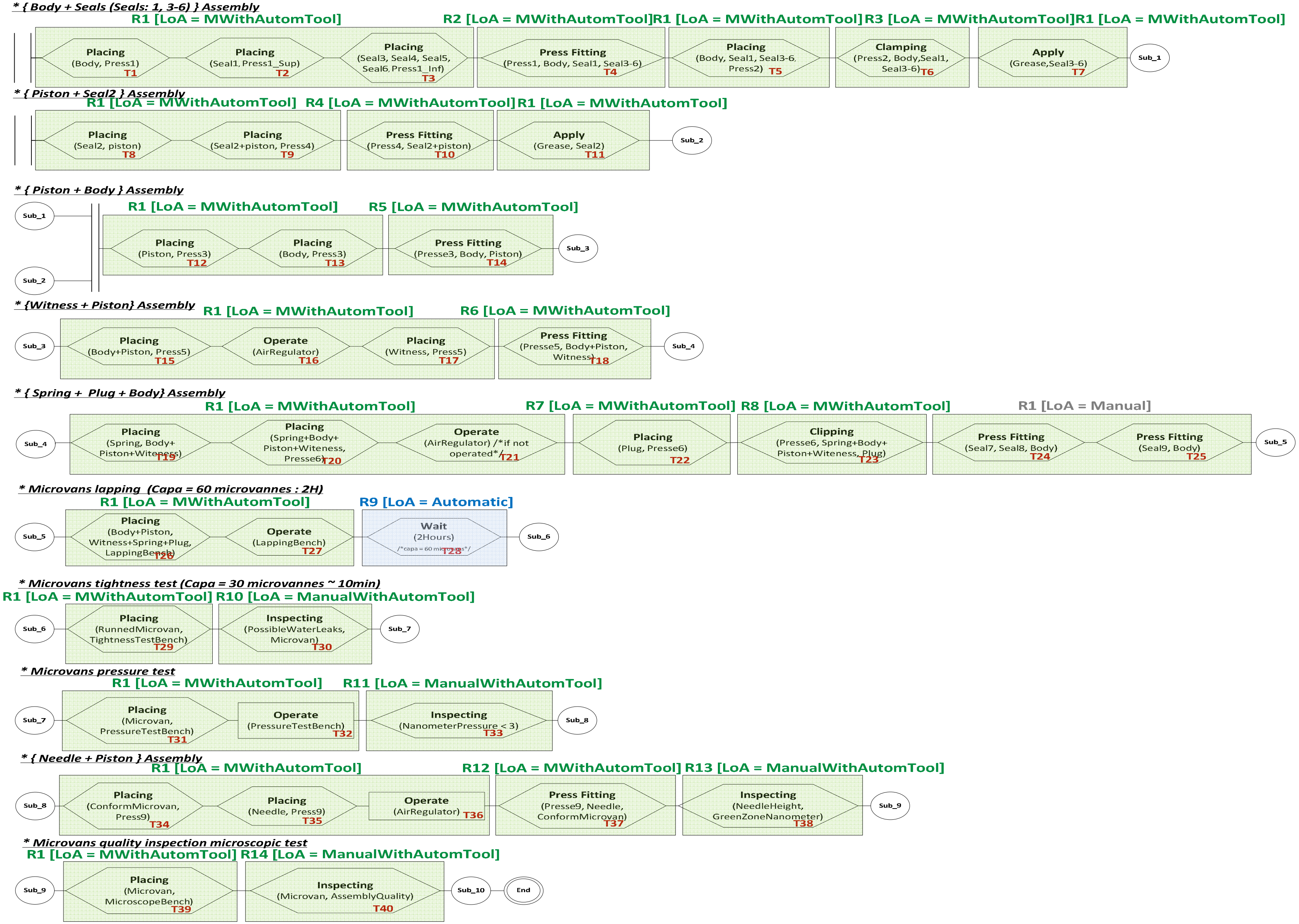
Annex C – Figure 2: The current assembly system model: an alternative with a minimum of investment

Annex C – Table 3: The first proposed alternative resources allocations, resources time estimation, and resources concerned parts and their costs

Resource (R_i)	**LoA**	**Description**	**Set of Assigned tasks (S_{Ri})**	**Total time estimate (T_{Ri}) [sec]**	**Concerned parts if rejec. non-conformity caused by the Res.**	**Concerned parts costs (C_{Parts}) [€]**
R1	Manual	Operator(s) + Automated system for operator(s) assembly components selection assistance	T1-T3; T5; T7-T9;T11-T13; T15-T17; T19-T21; T24-T27; T29 ; T31-T32 ;T34-T36 ; T39	183 [359 if consideration of human-machine cooperated operations (LoA=2): R2-R8;R10-R14]	All parts concerned	60,38
R2	ManualWithAutomAssist	Press N°1 + Operator(s)	T4	3	BD, SL1, SL3-6	23,76
R3	ManualWithAutomAssist	Press N°2 + Operator(s)	T6	10	BD, SL1, SL3-6	23,76
R4	ManualWithAutomAssist	Press N°4 + Operator(s)	T10	3	SL2+piston	14,67
R5	ManualWithAutomAssist	Press N°3 + Operator(s)	T14	3	BD+piston (+prev assembled parts: SL1-6)	38,43
R7	ManualWithAutomAssist	AirCompressor + Operator(s)	T22	5	PL(+BD +PS+WTN + SL1-6)	43,03
R8	ManualWithAutomAssist	Press N°6 + R1	T23	4	SP (+BD +PS+WTN+ PL+ SL1-6)	47,38
R9	Automatic	Lapping bench [*capacity = 60*]	T28	7200 (2H for 60)→120 (for 1)	SP + BD + PS+WTN+ PL+ SL1-6	47,38
R10	ManualWithAutomAssist	Tightness test bench [*capacity = 30*] + Operator(s)	T30	60 (10min for 30)→20 (for 1)	SP + BD + PS+WTN+ PL+ SL1-6	47,38
R11	ManualWithAutomAssist	Pressure test bench + Operator(s)	T33	40	SP+BD+PS+WTN+PL+SL1-9	55,88
R12	ManualWithAutomAssist	Press N°9 + Operator(s)	T37	4	NDL(+SP+ BD +PS+WTN+ PL+ SL1-9)	60,38
R13	ManualWithAutomAssist	Needle Assembly length test tool + Operator(s)	T38	20	NDL +SP+ BD +PS+WTN+ PL+ SL1-9	60,38
R14	ManualWithAutomAssist	Microscope quality inspection bench + Operator(s)	T40	60	NDL +SP+ BD +PS+WTN+ PL+ SL1-9	60,38

Annex C – Table 4: Resources input information for the existing process

Resource		**R1**	**R2**	**R3**	**R4**	**R5**	**R6**	**R7**	**R8**	**R9**	**R10**	**R11**	**R12**	**R13**	**R14**
Specific inputs to Manual	$AnnualLaborCost1W_R$[€]	30000	30000	30000	30000	30000	30000	30000	30000		30000	30000	30000	30000	30000
	C_{Rate_R} [€/sec]	0,00496	0,00496	0,00496	0,00496	0,00496	0,00496	0,00496	0,00496		0,00496	0,00496	0,00496	0,00496	0,00496
	$AverageRecruitPeriod1W_R$[Month]	24	24	24	24	24	24	24	24		24	24	24	24	24
	$ToolsEquipInvestCost_R$[€]	300	10	0	10	10	0	0	0		0	0	10	0	0
	$TotalTrainingCosts1W_R$[€]	1000	100	100	100	100	100	50	100		100	150	100	50	150
Specific inputs to Machined	Ren_R		1	2	1	1	1	1	1	1	1	1	1	1	1
	$ResWorkingLife_R$ [Hours]		400	400	400	400	400	25000	400	70000	70000	30000	400	400000	200000
	$ResInvestCost_R$[€]		1500	1500	1500	1500	1500	1000	1500	3500	2000	2500	1500	300	1200
	$C_{AnnualSP_R}$ [€]		50	50	50	50	50	0	50	100	100	100	50	0	0
	$C_{RateMaint_R}$ [€]		0,006	0,006	0,006	0,006	0,006	0,006	0,006	0,006	0,006	0,006	0,006	0,006	0,006
	$MTBF_R$ [Hours]		100	100	100	100	100	300	100	1000	1000	500	100	*Infinite*	*infinite*
	$MTTR_R$ [Hours]		5	5	5	5	5	7	5	25	25	20	5	0	0
	$\%Failure_R$		0.005	0.005	0.005	0.005	0.005	0.003	0.005	0.002	0.01	0.01	0.005	0	0
	$Power_R$ [KW]		3	3	3	3	3	5	3	1,5	1,5	1,5	3	0,05	0,2
Common inputs to Manual and Machined	$Dupli_R$	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	T_{Ri} [sec]	183	3	10	3	3	4	5	4	120	20	40	4	20	60
	T_{Setup_R}[sec]	30	2	2	2	2	2	0	2	0	3	3	2	0	3
	$C_{RateSetup_R}$[€/sec]	0,00496	0,00496	0,00496	0,00496	0,00496	0,00496	0,00496	0,00496	0,00496	0,00496	0,00496	0,00496	0,00496	0,00496
	$BatchSize_R$	60	60	60	60	60	60	60	60	60	60	60	60	60	60
	C_{Parts_R}[€]	60,38	23,76	23,76	14,67	38,43	40,53	43,03	47,38	47,38	47,38	55,88	60,38	60,38	60,38
	$\%Non\_Conform_R$	0.05	0.015	0.015	0.015	0.015	0.015	0.008	0.015	0.001	0	0.004	0.015	0.005	0.006
	$\%Reject_R$	0.02	0.001	0.001	0.001	0.001	0.001	0.005	0.001	0.001	0	0.003	0.001	0.002	0.002
	$Surface_R$ [m²]	30	4	4	4	4	4	2	4	9	7	6	4	2	5



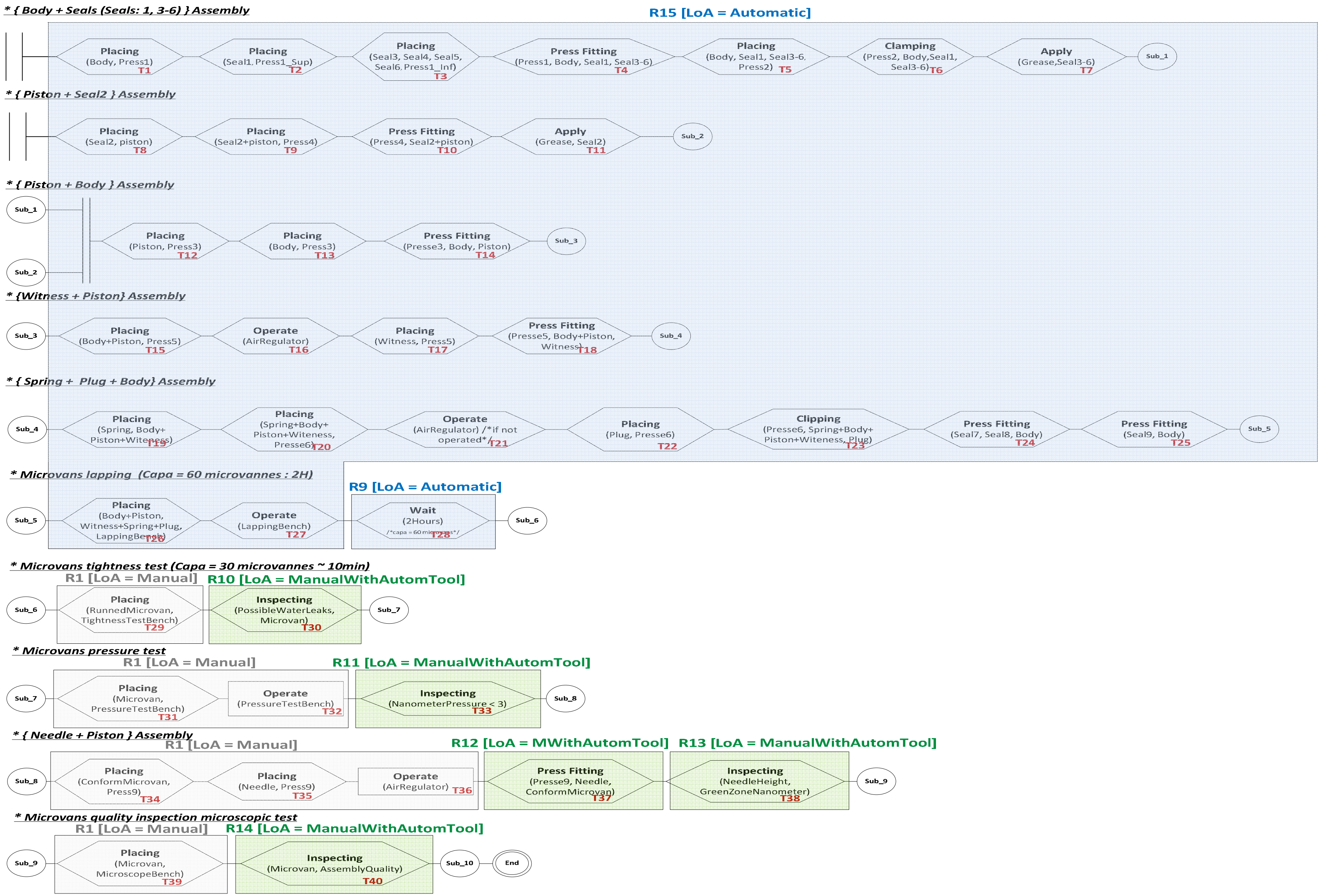
Annex C – Figure 3: The first alternative model: an automated assistance to guide and support operators during assembly

Annex C – Table 5: The first proposed alternative resources allocations, resources time estimation, and resources concerned parts and their costs

Resource (R_i)	**LoA**	**Description**	**Set of Assigned tasks (S_{Ri})**	**Total time estimate (T_{Ri}) [sec]**	**Concerned parts if rejec. non-conformity caused by the Res.**	**Concerned parts costs (C_{Parts}) [€]**
R1	ManualWithAutomAssist	Operator(s) + Automated system for operator(s) assembly components selection assistance	T1-T3; T5; T7-T9;T11-T13; T15-T17; T19-T21; T24-T27; T29 ; T31-T32 ;T34-T36 ; T39	183 [359 if consideration of human-machine cooperated operations (LoA=2): R2-R8;R10-R14]	All parts concerned	60,38
R2	ManualWithAutomAssist	Press N°1 + Operator(s)	T4	3	BD, SL1, SL3-6	23,76
R3	ManualWithAutomAssist	Press N°2 + Operator(s)	T6	10	BD, SL1, SL3-6	23,76
R4	ManualWithAutomAssist	Press N°4 + Operator(s)	T10	3	SL2+piston	14,67
R5	ManualWithAutomAssist	Press N°3 + Operator(s)	T14	3	BD+piston (+prev assembled parts: SL1-6)	38,43
R7	ManualWithAutomAssist	AirCompressor + Operator(s)	T22	5	PL(+BD +PS+WTN + SL1-6)	43,03
R8	ManualWithAutomAssist	Press N°6 + R1	T23	4	SP (+BD +PS+WTN+ PL+ SL1-6)	47,38
R9	Automatic	Lapping bench [*capacity* = *60*]	T28	7200 (2H for 60)→120 (for 1)	SP + BD + PS+WTN+ PL+ SL1-6	47,38
R10	ManualWithAutomAssist	Tightness test bench [*capacity* = *30*] + Operator(s)	T30	60 (10min for 30)→20 (for 1)	SP + BD + PS+WTN+ PL+ SL1-6	47,38
R11	ManualWithAutomAssist	Pressure test bench + Operator(s)	T33	40	SP+BD+PS+WTN+PL+SL1-9	55,88
R12	ManualWithAutomAssist	Press N°9 + Operator(s)	T37	4	NDL(+SP+ BD +PS+WTN+ PL+ SL1-9)	60,38
R13	ManualWithAutomAssist	Needle Assembly length test tool + Operator(s)	T38	20	NDL +SP+ BD +PS+WTN+ PL+ SL1-9	60,38
R14	ManualWithAutomAssist	Microscope quality inspection bench + Operator(s)	T40	60	NDL +SP+ BD +PS+WTN+ PL+ SL1-9	60,38

Annex C – Table 6: Resources input information for the first proposed alternative

Resource		**R1**	**R2**	**R3**	**R4**	**R5**	**R6**	**R7**	**R8**	**R9**	**R10**	**R11**	**R12**	**R13**	**R14**
Specific inputs to Manual	$AnnualLaborCost1W_R$[€]	30000	30000	30000	30000	30000	30000	30000	30000		30000	30000	30000	30000	30000
	C_{Rate_R} [€/sec]	0,00496	0,00496	0,00496	0,00496	0,00496	0,00496	0,00496	0,00496		0,00496	0,00496	0,00496	0,00496	0,00496
	$AverageRecruitPeriod1W_R$[Month]	24	24	24	24	24	24	24	24		24	24	24	24	24
	$ToolsEquipInvestCost_R$[€]	300	10	0	10	10	0	0	0		0	0	10	0	0
	$TotalTrainingCosts1W_R$[€]	1000	100	100	100	100	100	50	100		100	150	100	50	150
Specific inputs to Machined	Ren_R	1	1	2	1	1	1	1	1	1	1	1	1	1	1
	$ResWorkingLife_R$ [Hours]	*infinite*	400	400	400	400	400	25000	400	70000	70000	30000	400	400000	200000
	$ResInvestCost_R$[€]	4000	1500	1500	1500	1500	1500	1000	1500	3500	2000	2500	1500	300	1200
	$C_{AnnualSP_R}$ [€]	30	50	50	50	50	50	0	50	100	100	100	50	0	0
	$C_{RateMaint_R}$ [€]	0.009	0,006	0,006	0,006	0,006	0,006	0,006	0,006	0,006	0,006	0,006	0,006	0,006	0,006
	$MTBF_R$ [Hours]	5000	100	100	100	100	100	300	100	1000	1000	500	100	*Infinite*	*infinite*
	$MTTR_R$ [Hours]	10	5	5	5	5	5	7	5	25	25	20	5	0	0
	$\%Failure_R$	0.001	0.005	0.005	0.005	0.005	0.005	0.003	0.005	0.002	0.01	0.01	0.005	0	0
	$Power_R$ [KW]	0.1	3	3	3	3	3	5	3	1,5	1,5	1,5	3	0,05	0,2
Common inputs to Manual and Machined	$Dupli_R$	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	T_{Ri} [sec]	183	3	10	3	3	4	5	4	120	20	40	4	20	60
	T_{Setup_R}[sec]	30	2	2	2	2	2	0	2	0	3	3	2	0	3
	$C_{RateSetup_R}$[€/sec]	0,00496	0,00496	0,00496	0,00496	0,00496	0,00496	0,00496	0,00496	0,00496	0,00496	0,00496	0,00496	0,00496	0,00496
	$BatchSize_R$	60	60	60	60	60	60	60	60	60	60	60	60	60	60
	C_{Parts_R}[€]	60,38	23,76	23,76	14,67	38,43	40,53	43,03	47,38	47,38	47,38	55,88	60,38	60,38	60,38
	$\%Non\_Conform_R$	0.025	0.015	0.015	0.015	0.015	0.015	0.008	0.015	0.001	0	0.004	0.015	0.005	0.006
	$\%Reject_R$	0.005	0.001	0.001	0.001	0.001	0.001	0.005	0.001	0.001	0	0.003	0.001	0.002	0.002
	$Surface_R$ [m²]	30	4	4	4	4	4	2	4	9	7	6	4	2	5



Annex C – Figure 4: A second alternative of a future assembly system: assembly operations automation

Annex C – Table 7: The second proposed alternative resources allocations, resources time estimation, and resources concerned parts and their costs

Resource (R_i)	**LoA**	**Description**	**Set of Assigned tasks (S_{Ri})**	**Total time estimate (T_{Ri}) [sec]**	**Concerned parts if rejec. non-conformity caused by the Res.**	**Concerned parts costs (C_{Parts}) [€]**
R15	Automatic	Automatic assembly line	T1-T27	120	All parts concerned except the Needle part (NDL)	55.88
R9	Automatic	Lapping bench [*capacity = 60*]	T28	7200 (2H for 60)→120 (for 1)	SP + BD + PS+WTN+ PL+ SL1-6	47,38
R1	Manual	Operator(s)	T29; T31-T32; T34-T36; T39	34	NDL	4.5
R10	ManualWithAutomAssist	Tightness test bench [*capacity = 30*] + Operator(s)	T30	60 (10min for 30)→20 (for 1)	SP + BD + PS+WTN+ PL+ SL1-6	47,38
R11	ManualWithAutomAssist	Pressure test bench + Operator(s)	T33	40	SP+BD+PS+WTN+PL+SL1-9	55,88
R12	ManualWithAutomAssist	Press N°9 + Operator(s)	T37	4	NDL(+SP+ BD +PS+WTN+ PL+ SL1-9)	60,38
R13	ManualWithAutomAssist	Needle Assembly length test tool + Operator(s)	T38	20	NDL +SP+ BD +PS+WTN+ PL+ SL1-9	60,38
R14	ManualWithAutomAssist	Microscope quality inspection bench + Operator(s)	T40	60	NDL +SP+ BD +PS+WTN+ PL+ SL1-9	60,38

Annex C – Table 8: Resources input information for the second proposed alternative

Resource		**R15**	**R9**	**R1**	**R10**	**R11**	**R12**	**R13**	**R14**
Specific inputs to Manual	$AnnualLaborCost1W_R$[€]			30000	30000	30000	30000	30000	30000
	C_{Rate_R} [€/sec]			0,00496	0,00496	0,00496	0,00496	0,00496	0,00496
	$AverageRecruitPeriod1W_R$[Month]			24	24	24	24	24	24
	ToolsEquipInvestCost$_R$ [€]			80	0	0	10	0	0
	TotalTrainingCosts1W$_R$ [€]			500	100	150	100	50	150
Specific inputs to Machined	Ren_R	1	1		1	1	1	1	1
	$ResWorkingLife_R$ [Hours]	200000	70000		70000	30000	400	400000	200000
	ResInvestCost$_R$ [€]	60000	3500		2000	2500	1500	300	1200
	$C_{AnnualSP_R}$ [€]	1000	100		100	100	50	0	0
	$C_{RateMaint_R}$ [€]	0.006	0,006		0,006	0,006	0,006	0,006	0,006
	$MTBF_R$ [Hours]	1000	1000		1000	500	100	*Infinite*	*infinite*
	$MTTR_R$ [Hours]	20	25		25	20	5	0	0
	$\%Failure_R$	0.2	0.002		0.01	0.01	0.005	0	0
	$Power_R$ [KW]	15	1,5		1,5	1,5	3	0,05	0,2
Common inputs to Manual and Machined	$Dupli_R$	1	1	1	1	1	1	1	1
	T_{Ri} [sec]	143	120	34	20	40	4	20	60
	T_{Setup_R}[sec]	50	0	30	3	3	2	0	3
	$C_{RateSetup_R}$[€/sec]	0,00496	0,00496	0,00496	0,00496	0,00496	0,00496	0,00496	0,00496
	$BatchSize_R$	60	60	60	60	60	60	60	60
	C_{Parts_R}[€]	55.88	47,38	4.5	47,38	55,88	60,38	60,38	60,38
	$\%Non\_Conform_R$	0.015	0.001	0.05	0	0.004	0.015	0.005	0.006
	%Reject$_R$	0.01	0.001	0.02	0	0.003	0.001	0.002	0.002
	$Surface_R$ [m^2]	20	9	10	7	6	4	2	5

Annex D. *Cost estimation case study output information*

Annex D – Table 1: Cost output information for the existing process solution

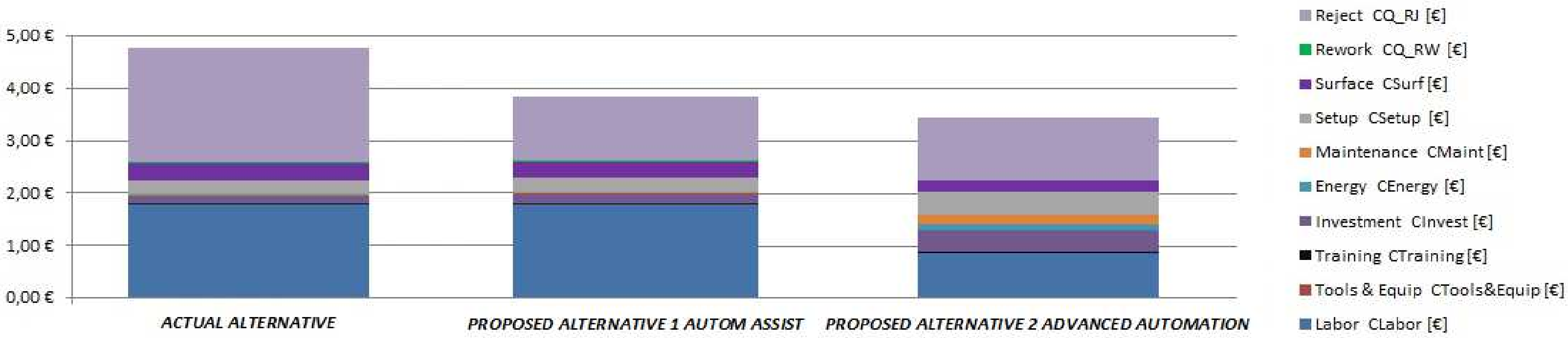
Resource			R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	R13	R14	*Total cost* [€]
Specific to ***Manual***	***Labor*** C_{Labor} [€]		0,907738095	0,014880952	0,049603175	0,01488095	0,01488095	0,01984127	0,02480159	0,01984127		0,09920635	0,1984127	0,01984127	0,09920635	0,29761905	1,780753968
	Tools & Equip $C_{Tools\&Equip}$ [€]		0,001666667	5,55556E-05	0	5,5556E-05	5,5556E-05	0	0	0		0	0	5,5556E-05	0	0	0,001888889
	Training $C_{Training}$ [€]		0,016666667	0,001666667	0,001666667	0,00166667	0,00166667	0,00166667	0,00083333	0,00166667		0,00166667	0,0025	0,00166667	0,00083333	0,0025	0,036666667
Specific to ***Machined***	***Investment*** C_{Invest} [€]			0,008333333	0,016666667	0,00833333	0,00833333	0,00833333	0,00555556	0,00833333	0,01944444	0,01111111	0,01388889	0,00833333	0,00166667	0,00666667	0,125
	Energy C_{Energy} [€]			0,00038	0,001266667	0,00038	0,00038	0,00050667	0,00105556	0,00050667	0,0076	0,00126667	0,00253333	0,00050667	4,2222E-05	0,00050667	0,016931111
	Maintenance C_{Maint} [€]			0,0018795	0,002376111	0,0018795	0,0018795	0,00195044	9,9322E-05	0,00195044	0,005036	0,00475222	0,00787378	0,00195044	0	0	0,031627267
Common *to* ***Manual*** *&* ***Machined***	***Setup*** C_{Setup} [€]		0,141369047	0,009424603	0,009424603	0,0094246	0,0094246	0,0094246	0	0,0094246	0	0,0141369	0,0141369	0,0094246	0	0,0141369	0,249751984
	Surface C_{Surf} [€]		0,108	0,0144	0,0144	0,0144	0,0144	0,0144	0,0072	0,0144	0,0324	0,0252	0,0216	0,0144	0,0072	0,018	0,3204
	Quality	***Rework*** C_{Q_RW} [€]	0,027232143	0,000213653	0,000712178	0,00021365	0,00021365	0,00028487	7,7571E-05	0,00028487	0	0	0,00020095	0,00028487	0,00029775	0,0011925	0,03120866
		Reject C_{Q_RJ} [€]	1,225754762	0,023775261	0,01472087	0,03844526	0,04054526	0,04305035	0,23702929	0,04740035	0,0473876	0	0,16824284	0,06040035	0,1209585	0,12135625	2,189066931
		Total Quality C_Q [€]	1,252986905	0,023988914	0,015433048	0,03865891	0,04075891	0,04333522	0,23710686	0,04768522	0,0473876	0	0,16844378	0,06068522	0,12125624	0,12254875	2,220275591
Cost per product per resource [€]			2,428427381	0,075009525	0,110836936	0,08967953	0,09177953	0,0994582	0,27665221	0,1038082	0,11186804	0,15733992	0,42938939	0,11686376	0,23020481	0,46197804	**4,783295476**

Annex D – Table 2: Cost output information for the existing process solution

Resource			R1	R2	R3	R4	R5	R6	R7	R8	R9	R10	R11	R12	R13	R14	*Total cost* [€]
Specific to ***Manual***	***Labor*** C_{Labor} [€]		0,907738095	0,014880952	0,049603175	0,01488095	0,01488095	0,01984127	0,02480159	0,01984127		0,09920635	0,1984127	0,01984127	0,09920635	0,29761905	1,780753968
	Tools & Equip $C_{Tools\&Equip}$ [€]		0,001666667	5,55556E-05	0	5,5556E-05	5,5556E-05	0	0	0		0	0	5,5556E-05	0	0	0,001888889
	Training $C_{Training}$ [€]		0,016666667	0,001666667	0,001666667	0,00166667	0,00166667	0,00166667	0,00083333	0,00166667		0,00166667	0,0025	0,00166667	0,00083333	0,0025	0,036666667
Specific to ***Machined***	***Investment*** C_{Invest} [€]		0,033333333	0,008333333	0,016666667	0,00833333	0,00833333	0,00833333	0,00555556	0,00833333	0,01944444	0,01111111	0,01388889	0,00833333	0,00166667	0,00666667	0,158333333
	Energy C_{Energy} [€]		0,000772667	0,00038	0,001266667	0,00038	0,00038	0,00050667	0,00105556	0,00050667	0,0076	0,00126667	0,00253333	0,00050667	4,2222E-05	0,00050667	0,017703778
	Maintenance C_{Maint} [€]		0,001104961	0,0018795	0,002376111	0,0018795	0,0018795	0,00195044	9,9322E-05	0,00195044	0,005036	0,00475222	0,00787378	0,00195044	0	0	0,032732227
Common *to* ***Manual*** *&* ***Machined***	***Setup*** C_{Setup} [€]		0,141369047	0,009424603	0,009424603	0,0094246	0,0094246	0,0094246	0	0,0094246	0	0,0141369	0,0141369	0,0094246	0	0,0141369	0,249751984
	Surface C_{Surf} [€]		0,108	0,0144	0,0144	0,0144	0,0144	0,0144	0,0072	0,0144	0,0324	0,0252	0,0216	0,0144	0,0072	0,018	0,3204
	Quality	***Rework*** C_{Q_RW} [€]	0,018170215	0,000213653	0,000712178	0,00021365	0,00021365	0,00028487	7,7571E-05	0,00028487	0	0	0,00020095	0,00028487	0,00029775	0,00149063	0,022444858
		Reject C_{Q_RJ} [€]	0,306442554	0,023775261	0,01472087	0,03844526	0,04054526	0,04305035	0,23702929	0,04740035	0,0473876	0	0,16824284	0,06040035	0,1209585	0,06067813	1,209076597
		Total Quality C_Q [€]	0,324612769	0,023988914	0,015433048	0,03865891	0,04075891	0,04333522	0,23710686	0,04768522	0,0473876	0	0,16844378	0,06068522	0,12125624	0,06216875	1,231521455
Cost per product per resource [€]			1,535264206	0,075009525	0,110836936	0,08967953	0,09177953	0,0994582	0,27665221	0,1038082	0,11186804	0,15733992	0,42938939	0,11686376	0,23020481	0,40159804	**3,829752301**

Annex D – Table 3: Cost output information for the existing process solution

Resource			R15	R9	R1	R10	R11	R12	R13	R14	*Total cost* [€]
Specific to ***Manual***	***Labor*** C_{Labor} [€]				0,168650794	0,09920635	0,1984127	0,01984127	0,09920635	0,29761905	0,882936508
	Tools & Equip $C_{Tools\&Equip}$ [€]				0,000444444	0	0	5,5556E-05	0	0	0,0005
	Training $C_{Training}$ [€]				0,008333333	0,00166667	0,0025	0,00166667	0,00083333	0,0025	0,0175
Specific to ***Machined***	***Investment*** C_{Invest} [€]		0,333333333	0,019444444		0,01111111	0,01388889	0,00833333	0,00166667	0,00666667	0,394444444
	Energy C_{Energy} [€]		0,090566667	0,0076		0,00126667	0,00253333	0,00050667	4,2222E-05	0,00050667	0,103022222
	Maintenance C_{Maint} [€]		0,195654222	0,005036		0,00475222	0,00787378	0,00195044	0	0	0,215266667
Common *to* ***Manual*** *&* ***Machined***	***Setup*** C_{Setup} [€]		0,235614815	0	0,141368889	0,01413689	0,01413689	0,00942459	0	0,01413689	0,428818963
	Surface C_{Surf} [€]		0,072	0,0324	0,036	0,0252	0,0216	0,0144	0,0072	0,018	0,2268
	Quality	***Rework*** C_{Q_RW} [€]	0,000452833	0	0,005059524	0	0,00020095	0,00028487	0,00029775	0,0011925	0,007488423
		Reject C_{Q_RJ} [€]	0,559705667	0,0473876	0,093373016	0	0,16824284	0,06040035	0,1209585	0,12135625	1,171424217
		Total Quality C_Q [€]	0,5601585	0,0473876	0,09843254	0	0,16844378	0,06068522	0,12125624	0,12254875	1,17891264
Cost per product per resource [€]			1,487327537	0,111868044	0,45323	0,1573399	0,42938937	0,11686375	0,23020481	0,46197802	**3,448201444**



Annex D Figure 1: Obtained assembly costs per product for the different assembly systems alternatives

Annex D Table 4: Obtained performance indicators for the 2 proposed alternatives compared to the actual process solution

Initial Investment (Vs Actual Process)	Total Investment including renov.	Total Surface		Total Power		Payback of additional invest. Vs actual process	Return On Investment
		Surface	Monthly Surface cost	Power	Monthly Pwr. cost		
			Proposed Alternative 1				
27340 € (-21340€)	27340€+1500€ (-21340€)	89 m^2	801 €/Month	30.85 KW	44.26 €/Month	2.51 Months	27 x
			Proposed Alternative 2				
71090 € (-21340€)	71090 € (-21340€)	63 m^2	567 €/Month	22.75 KW	257.55 €/Month	14.4559 Months	4 x

Résumé

Le contexte industriel s'avère de plus en plus difficile dans le cadre de la mondialisation et de la compétition internationale. Une meilleure conception des processus d'assemblage avec un choix judicieux en matière d'automatisation devient crucial afin d'assurer une compétitivité sur le marché. Cette thèse a pour objectif la détermination d'une méthode ainsi que d'un outil d'aide à la décision pour la modélisation et la conception de systèmes d'assemblage, en particulier, avec un niveau d'automatisation optimal. Dans ce cadre, l'état de l'art effectué a souligné une littérature non abondante et un manque de support objectif et méthodique des approches existantes. L'analyse approfondie des méthodes de décision a confirmé que les méthodes qui existent ne fournissent pas une aide significative qui soit alignée avec les exigences complexes et multicritères du sujet. Ceci a conduit à la définition d'une nouvelle approche pour adresser les exigences identifiées. L'approche, afin d'être concrètement applicable, nécessite le développement de différents axes. Ces axes concernent: un langage graphique standardisé de représentation des processus d'assemblage, une identification des critères impliqués dans la décision et leur intégration dans le processus de décision, une approche d'estimation des temps et des coûts d'assemblage, et une technique de génération de scénarios. Ces différents axes sont abordés dans cette thèse. Des propositions validées sur des exemples sont fournies, tout en étant accompagnées au préalable de revues et analyses approfondies de la littérature.

Mots-clés: *Conception de systèmes d'assemblage, modélisation de processus d'assemblage, décision du niveau d'automatisation, optimisation des systèmes de production, estimation de coût d'assemblage*

Abstract

The current industrial context is characterized by a globalized international competition imposing a need to master investments and costs. To be competitive, manufacturers should particularly optimize their processes and productions. Automation is observed as one of the most efficient tools to face this issue. From another side, manufacturers realized that an increased usage of automation does not necessarily result in increasing benefits. The appropriate level of automation should be consequently found. This research aims at defining a procedure and a tool to help assembly systems designers in the decision about the most appropriate level of automation. The purpose is to help to find the optimal Level of Automation (LoA) of the process since the early conceptual design phase. A state of the art of the topic was realized and has shown that the literature about LoA decision is not abundant. A need to an objective, concrete, and methodic approach to guide the decision was identified. A new approach is defined in this thesis. To make the approach practically usable, different axes have to be developed. These axes concern: a modelling language to represent automation alternatives, automation decision criteria identification and consideration, process time and cost estimation, automation alternatives generation and optimization. In this thesis, reviews and literature analysis are presented in each of the axes. Proposals are also presented for each. The proposals include theoretical and industrial validations.

Keywords: *Assembly systems design, assembly modelling, level of automation decision, assembly systems optimization, assembly cost estimation*