An automated assembly process planning system

Charisis Bikas, Angelos Argyrou, George Pintzos, Christos Giannoulis, Kostantinos Sipsas, Nikolaos Papakostas, George Chryssoulouris

Abstract

The generation of assembly process plans of complex products is a challenging task. An assembly processing planning system reducing the human intervention and the computational effort is discussed. The method utilizes the information stored into the CAD model of the assembly for the extraction of the part precedence information which is then complemented with technological priorities. Results are demonstrated in an industrial case study from the optoelectronics industry.

1. Introduction

A product’s assembly plan affects both the assembly process efficiency and the assembly line’s design. Planning and using efficient assembly processes can actively contribute to the reduction of a product’s manufacturing cost. Assembly Planning includes the determination of a feasible method and layout, in order for a product to be assembled from its components.

Mechanical products often have multiple assembly sequence plans, due to the complexity and the multiplicity of their components. Therefore, some assembly sequence plans are more efficient than others and the selection of the appropriate one requires a high level of expertise and experience from the planner’s side. Assembly process planning is a time-consuming procedure and, as a result, the automation of this procedure is necessary. The goal of the current study is the proposition of a method that can contribute to the reduction in time and effort on process planning generation.

To demonstrate the functionalities and the innovation potential an industrial pilot case was selected. The pilot case involves robotic assembly of solar panel cells, which can be produced in different variants (power output, shape, size etc.) in order to demonstrate the adaptability of the process planning method.

2. Current practices

2.1. Process planning

The Robotic equipment has found great application to a broad range of automatic assembly systems, specifically in the assembly lines of automotive industry, electronics, rubber/plastics and metal/machinery industrial sectors. The robots’ intrinsic characteristics, such as high accuracy, speed, repeatability, strength and reliability, have enabled production firms to invest in large scale installations that can work around the clock with minimal human intervention [1]. Nevertheless, technological limitations impose the contribution of human operators on the process, by providing support to the system [2].

The development of such complex systems and the variation of production conditions bring about new problems [3]. A plethora of such problems have been extensively analyzed by researchers [4][5][6][7][8], and include conflicts between process planning and scheduling, unbalanced resources in the production line and the problem of selecting suitable resources with respect to the given conditions. From
an optimization point of view, several computational methods have been proposed in order to address these issues. A novel approach is presented by Papakostas et al. [9][10][11], which includes a data model along with a set of rules for realizing the knowledge-enabled structuring of assembly process information, taking into account 3D specifications of both robotic manipulators and parts. Xinyu Li [5] experimentally investigated into the impact of multi agent modelling on Integrated Process Planning and Scheduling (IPPS) optimization. A further study was also performed, integrating game theory in order for multi-objective manufacturing problems, similar to assembly scheduling problems to be addressed [6]. In the same context, the Particle Swarm Optimization (PSO) algorithms were used by Guo et al. [6][13][14] to search the optimum solution for both scheduling and process planning. A PSO-based algorithm was also proposed by Papakostas et al [14], where both time and cost parameters were considered for the generation of alternative sequences. For the facilitation of the integration and optimization of process planning and scheduling, an annealing-based simulation was proposed by Li et al. [15], combined with a unified representation model. Heuristics is another approach to solving such problems. Pierre de Lit et al. [16] provided an original Ordering Genetic Algorithm (OGA), enabling the automatic generation and evaluation of assembly product trees. Similar practices were followed in a further study by Carmelo Del Valle et al. [17], where their solution was based on and/or diagrams. A more applied approach of genetic models to assembly was made by X.F. Zha et al. [18], who proposed and implemented an algorithm in standard modelling language EXPRESS/EXPRESS-G. A more sophisticated version of the aforementioned algorithm was also proposed by Greg C. Smith et al.[19], who presented a modified automatic generation of initial assembly sequence population. The performance of Artificial Immune System (AIS), Iterated Greedy algorithm (IG) and (AIS-IG) algorithms was investigated into the makespan flow shop scheduling problem in [20], Mohapatra et al.[21] evaluated the performance of different process plans in terms of minimizing the makespan, machining cost and idle time of machines. JR Li et al. [22][23] presented a study, matching the advantages of the Tabu search algorithm and the enhanced genetic algorithm for assembly. C.W. Leung et al. [24] presented a study by combining an agent based model and the ant colony optimization algorithms. However, the aforementioned studies seldom accommodate a rescheduling mechanism and take into consideration the fluctuation of the real production line conditions [8]. To this effect, Abumairaz and Svestka proposed an algorithm for rescheduling in job shops [25]. The experimental/simulation assessment of an online scheduling approach is demonstrated by Chih-Chiang Hsu et al. [26] for multiple mixed-parallel workflows in grid environments. A more recent research done by Elisabeth Gunther et al. [27] proposes the online computation of close to best competitive ratio solutions, using competitive-ratio approximation schemes. Meanwhile, Michalos et al. [1] presented a more applied method through using the hierarchical and decision making algorithms to obtain rotation alternative schedules and evaluate them, according to several criteria. With similar ethos, Michalos et al. [28] provided a web-based platform combined with intelligent search algorithms to generate job rotation schedules. Viera et al. [29]; Ayteg et al. [30] and Potts and Strusevich [31] provided a comprehensive review of research studies on various types of rescheduling problems.

2.2. Assembly planning

The Assembly Planning aims to identify and evaluate the different ways of constructing a mechanical object from its components. “Given a geometrical and technological description of a product, find an assembly sequence that satisfies the precedence relations between operations and meets certain optimization criteria.” [9]. In the last decade, several approaches have been proposed to automatically generate assembly sequences. Summarizing, the existing approaches for the generation of assembly plans, can be roughly classified into three main approaches [9][10][11]:

- human-interaction,
- geometry-based reasoning and
- knowledge-based reasoning.

The automatic generation of assembly sequences, which is the key topic of computer aided assembly process planning (CAPP), has been an object of research for the past 30 years [33]. The assembly sequence generation is a part of the wider problem of Assembly Planning (ASP), especially when a large number of potential assembly sequences exist, as in the case of complex assemblies. Therefore, there is a growing need for the generation of assembly sequences to be systematized and computerized. Thus, many research activities have focused on various aspects of assembly sequence planning, such as assembly modelling, assembly sequence representation and assembly sequence generation algorithms. However, these methods and algorithms are less interactive and require more space to store the representation of assembly sequences and process time for complex problems [14].

Directed graph methods have been widely used to represent the ASP problem [11], [14]. A directed graph D = (P, C) can describe the assembly, where each vertex (P) represents a component, and each edge (C) represents a relationship between two components. In some cases, the contact-base feature is employed to represent the precedence relationships of the product, consequently, the directed graph is also called precedence diagram. However, the contact-base precedence diagram cannot effectively express the complexity of the assigned assembly relations [32].

A summary of the aforementioned methodologies, including their advantages and disadvantages can be found in Table 1. However, these approaches are often purely academic, without providing an implemented tool for the end user to enable automated process planning, based on the CAD file of the product.
Table 1. Advantages and disadvantages of the most popular assembly planning methodologies.

<table>
<thead>
<tr>
<th>Method</th>
<th>Advantages</th>
<th>Disadvantages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Graph theory</td>
<td>Geometric reasoning based on mathematical reasoning to construct assembly modelling and assembly sequences, which can express the entire sequences of the solution space</td>
<td>All the assembly components and the assembly operations have to be formulated, which will lead to combinatorial explosion problem.</td>
</tr>
<tr>
<td>Knowledge based approach</td>
<td>Takes advantage of the human experience and knowledge and is able to consider more engineer information</td>
<td>Should analyse geometry and assembly relations, which will become difficult with the increasing part quantity.</td>
</tr>
<tr>
<td>Heuristic search-based algorithm</td>
<td>Can intelligently compute an optimized assembly sequence on basis of the effective and reasonable creation of assembly information modelling</td>
<td>When a complex product needs to use assembly fixtures and tools, construction for intelligent planning algorithms will become extremely difficult.</td>
</tr>
<tr>
<td>Virtual reality based approach</td>
<td>Makes good use of advantages of virtual reality technology and human experience</td>
<td>Difficult to obtain optimized assembly sequence, when the part quantity is very large.</td>
</tr>
</tbody>
</table>

3. Process planning system

In the context of the EU-funded project “white-R”, an automated computer aided process planning system was developed. The developments can be divided into three main phases. The first phase regards the development of the Process Planning component, where the process plan is generated in a high abstraction level programing language, followed by the resource programming phase and the evaluation phase. A schematic overview of the developed process planning system is presented in Fig. 1. The present study will be dealing only with the Process Planning component. The purpose of the Process Planning Component is to obtain a plan of operations for each assembly, starting from the CAD file of the assembly. The Process Planning Component consists of two software modules, namely the Assembly Sequence Generation (ASG) module and the Joining Planning Algorithm module. The two modules are controlled via a web-based Graphical User Interface. The developments of each module will be detailed in the following sections.

3.1. Assembly Sequence Generation module

The Assembly Sequence Generation (ASG) module is implemented within a commercial software package, utilizing macros written in the VBA macro language. This module is based on a collision detection model that performs a disassembly process on the product assembly CAD file. It generates the assembly sequence by essentially reversing the disassembly process.

Based on the assembly CAD model, the total number of the assembly’s first-level components is counted and the system retrieves and stores the identity and geometrical information of the assembly’s components. The latter being identified as joining elements are removed from the assembly, and it is only the functional parts that remain. A user’s intervention is required for the selection of the assembly’s Base part(s) namely the first part to be assembled. After the Base selection, the system automatically retrieves the geometrical constraints of the assembly model and prompts the user to manually insert any further constraints that should be applied to the assembly. Once the user constraints have been defined, the disassembly process begins. Intersection tests are applied to each first-level component of the assembly (except for the Base part) and towards each one of the global (+x, +y, +z) and local (+u, ±v, ±w) directions of each component. The parts with non-zero number of possible disassembly directions are removed from the assembly. The possible disassembly directions, together with the tier number, are stored into the disassembly matrix for each part subtracted. After the removal of each tier’s parts, new intersection tests are applied to the remaining parts and the above process is sequentially repeated in each step of the disassembly process until only the Base part remains.

All the removed parts, together with the joining elements, are reintroduced to the active CAD file, forming again the initial assembly, whilst the user is prompted to select, for each part, the preferred disassembly direction among the possible ones. Based on the chosen disassembly directions of the parts, the reference part for each joining element as well as further precedence relations among the components are determined.
At this step, the assembly matrix is formed on the basis of tiers. Once the assembly matrix has been generated, it is exported to a workbook containing the essential information for the precedence relations of the components. The discrete steps of the algorithm for the generation of the assembly matrix based on tiers is illustrated in the flowchart of the implementation stages, as seen in Fig. 2.

3.2. Joining Planning Algorithm module

After the ASG module has generated the part precedence diagram, the output has to be enhanced for a plan of operations. In certain steps, it is necessary that a user intervention be included in order to include operations or characteristics, unique to each product, such as multiple operations in a single step. To better understand the process, an example output of the ASG module will be included hereafter, in order for the reader to be able to follow the process, using a simplified demonstration part. The latter is a simplified solar panel assembly, consisting of 2 solar cells and 3 tabbing wires in a row, as seen in Fig. 3.

Every task comprises a certain set of operations. For instance, the “Pick and place” task consists of the following operations:

The generic “connect” task, namely screwing, bonding, welding, soldering etc. has to be specified for each case. Each one of the aforementioned “connect” tasks could be then broken down to a set of operations. There are two ways of establishing the type of the “connect” task to be used. The user is either allowed to select the appropriate connecting method manually, or assign part names (or ID’s) with specific connecting methods, via a pre-defined lookup table.
It is assumed that the “connect” task to be used is the “soldering” one. The “solder” task consists only of one operation (“solder”). Therefore, the table containing tasks can now be transformed into one containing specific operations. At this point, it would be beneficial that the result be displayed to the user in order for him to confirm or manually edit the plan of operations, if necessary. Now, the “Part Type” column is redundant and can be omitted. Moreover, part names and operations can now be assigned to their respective IDs. The substitution is made in this specific step, in order for the user to be dealing with more “user-friendly” part labels, throughout the process that requires human supervision and possible intervention, while the process planning module output, in order to be useful, should use label IDs. The end result is a sequence of operations for the specific product.

Table 5. Enhanced ASG output in operation level (step 4).

<table>
<thead>
<tr>
<th>Index</th>
<th>Part name</th>
<th>Part Type</th>
<th>Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Tabbing wire_Type A.1</td>
<td>Auxiliary</td>
<td>Pick</td>
</tr>
<tr>
<td>2</td>
<td>Tabbing wire_Type A.1</td>
<td>Auxiliary</td>
<td>Transport</td>
</tr>
<tr>
<td>3</td>
<td>Tabbing wire_Type A.1</td>
<td>Auxiliary</td>
<td>Place</td>
</tr>
<tr>
<td>4</td>
<td>Tabbing wire_Type A.1</td>
<td>Auxiliary</td>
<td>Block</td>
</tr>
<tr>
<td>5</td>
<td>Tabbing wire_Type A.1</td>
<td>Auxiliary</td>
<td>Release</td>
</tr>
<tr>
<td>6</td>
<td>Cell Type A.1</td>
<td>Main</td>
<td>Pick</td>
</tr>
<tr>
<td>7</td>
<td>Cell Type A.1</td>
<td>Main</td>
<td>Transport</td>
</tr>
<tr>
<td>8</td>
<td>Cell Type A.1</td>
<td>Main</td>
<td>Place</td>
</tr>
<tr>
<td>9</td>
<td>Cell Type A.1</td>
<td>Main</td>
<td>Block</td>
</tr>
<tr>
<td>10</td>
<td>Cell Type A.1</td>
<td>Main</td>
<td>Release</td>
</tr>
<tr>
<td>11</td>
<td>Tabbing wire_Type A.2</td>
<td>Auxiliary</td>
<td>Pick</td>
</tr>
<tr>
<td>12</td>
<td>Tabbing wire_Type A.2</td>
<td>Auxiliary</td>
<td>Transport</td>
</tr>
<tr>
<td>13</td>
<td>Tabbing wire_Type A.2</td>
<td>Auxiliary</td>
<td>Place</td>
</tr>
<tr>
<td>14</td>
<td>Tabbing wire_Type A.2</td>
<td>Auxiliary</td>
<td>Block</td>
</tr>
<tr>
<td>15</td>
<td>Tabbing wire_Type A.2</td>
<td>Auxiliary</td>
<td>Release</td>
</tr>
<tr>
<td>16</td>
<td>Tabbing wire_Type A.2</td>
<td>Auxiliary</td>
<td>Solder</td>
</tr>
<tr>
<td>17</td>
<td>Cell Type A.2</td>
<td>Main</td>
<td>Pick</td>
</tr>
<tr>
<td>18</td>
<td>...</td>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>

4. Conclusions and Outlook

An automated process planning system was developed and presented. In order for the systems’ correctness to be checked, a simplified demonstration part was used, having only 2 solar cells and 3 tabbing wires in a row. An early implementation of the ASG algorithm was tested on this specific part and its output was used for the generation of the plan of operations. All the logic steps described by the algorithm were implemented, albeit not by software, for the generation of a primitive output to confirm the methodology’s correctness and functionality along with spotting any other issues early in the development phase.

A number of issues, during the algorithm’s development and testing, were noted and had to be dealt with before the final implementation of the system. To begin with, the ASG algorithm associates a name to each part (instance) within the assembly. The names on the ASG algorithm output file are the same as those of the individual instances in the CAD assembly. Therefore, if the designer has not used the appropriate names, in no way will the software associate those parts with the ones described in the white’R Preliminary Project reference framework, which in turn, will pose a series of difficulties upon trying to generate the actual part program out of the plan of operations.
It would be useful that the developed framework be extended so as to include coordinates of each part within the cell. This information could be stored into a separate database; however, it would be also useful to have a dynamic coordinates matrix for each part instance, in order for its position to be tracked every single moment within the robotic island. In addition, the process planning module output could be enhanced with information, regarding the handlers carrying out a specific operation, in order for the actual equipment programming to be enhanced. This information could be included as a “handler capability index”, where each handler, available in the robotic cell inventory, would be assigned to specific parts and operations.

Acknowledgment

The work presented in the current paper is partially funded by the FP7 EU project White’R (http://whiterproject.eu/).

References

[23] The top of a round wooden table has the shape shown. Determine how; 2007; p. 5000.