Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 57 (2016) 516 – 521

www.elsevier.com/locate/procedia

49th CIRP Conference on Manufacturing Systems (CIRP-CMS 2016)

Towards feature-based human-robot assembly process planning

Csaba Kardos[a,b,*], András Kovács[a], József Váncza[a,b]

[a]Fraunhofer Project Center for Production Management and Informatics
Institute for Computer Science and Control, Hungarian Academy of Sciences
[b]Department of Manufacturing Science and Technology, Budapest University of Technology and Economics

* Corresponding author. Tel.: +36-1-279-6181; *E-mail address:* csaba.kardos@sztaki.mta.hu

Abstract

The paper proposes a generic approach to automated robotic assembly process planning. Such a novel feature-based model of the assembly process is presented which can be synthesized from the standard CAD model of the product and the description of the applicable resources. As a first step towards automated planning, the paper focuses on generating constraints that ensure plan feasibility, as well as on the formal verification of fully specified plans. Examples are given from the domains of robotic remote laser welding as well as collaborative human-robot mechanical assembly.

Keywords: Assembly planning; Robot; Feature

1. Introduction

Robots are becoming crucial, more and more indispensable elements of today's production and logistics systems, thanks to their *flexibility*, reliability, and warranted high quality of work. Together with this trend in industrial automation there increases the need for production *efficiency*. Hence the challenges are manifold: the typically conflicting requirements for flexibility and efficiency should be consolidated along with observing all the technological and geometrical constraints that are implied when using robots in a particular application domain. Designing the structure, planning and verifying the behaviour, as well as controlling and monitoring task execution of a robotic system should go hand in hand, in close interaction, facilitated by decision support tools that use generic models of products, robots as well as other resources (like workcells, workers, fixtures, tools) that take part in actual production.

Our specific domain of interest is *assembly* where robots inhabited mass production environments, e.g., in the automotive industry, for a long time. However, one of our main concerns here is to find a resolution to the flexibility vs. efficiency dilemma in small-scale, even personalized production that calls for new models and methods of automated *assembly planning* [1,2]. Secondly, in robotic assembly one can observe a shift from complete automation towards human-robot collaboration in shared workspaces [3]. Provided safety requirements can be warranted (e.g., by vision-guided active collision avoidance [4]), the scope of potential applications will grow to a large extent. The ultimate *goal* of this research is to develop such automated *process planning* tools and technologies for supporting robotic assembly that are generic across a number of domains.

Our current research centers around symbiotic acting together of human workers and robots in engine assembly, where operations on mechanical parts (such as placing, insertion, fitting, screwing, etc.) can be performed both by humans or robots. However, the scope includes, as an extreme, also fully robotic assembly like *remote laser welding* (RLW) where welding tasks are accomplished by a laser beam emitted from a scanner that acts as the end-effector of a robot [5–7].

Two general approaches are unanimously taken to cope with the inherent complexity of assembly process planning: (1) *aggregation* that suggests a hierarchical decision scheme separating macro and micro planning [1], and (2) feature-based *decomposition* that helps structuring domain knowledge around local assembly features. *Assembly features* that are derived from the CAD model of the product [8] imply tasks, the use of specific resources, and modes of tasks execution [2]. While *macro planning* is responsible for (re-)configuring assembly workcells, ordering the tasks and assigning resources, *micro planning* involves motion, path and trajectory planning, generation of work instructions and the determination of process parameters. In robotic assembly micro planning is especially challenging since feasible, collision-free trajectory of the robot has to be generated while striving for minimal cycle time. Nowadays, thanks

doi:10.1016/j.procir.2016.11.089

to advanced digital data acquisition, motion capture and visualization methods assembly planning is accompanied with virtual evaluation, testing and simulation [8–10]. However, simulation of virtual assembly cannot support completely the planning process [10]. In fact, *geometric reasoning* combined with motion planning should be used for ensuring feasibility of robotic assembly sequences. Furthermore, recognized assembly features can provide the basis also for generating human work instructions [11].

Automated process planning in general is one of the hardest problems in production engineering because it has to concern both the worlds of design and production. Still, based on our experience in planning in the machining [12,13], sheet metal bending [14] and recently, the RLW [5–7] domains we believe that while process planning requires observing a wide variety of domain specific constraints (on tools, setups, operations and their ordering, movements, etc.), there can be defined an underlying *generic representation* for capturing all the essential elements, relations and criteria of the process planning problem. This paper presents the first steps towards such a generic model in robotic assembly, together with a proposed methodology that handles the *verification* of feature-based robotic assembly plans. Examples from both the human-robot mechanical assembly and the RLW domains will be provided.

2. Problem definition

This paper looks at assembly process planning as part of the workstation configuration problem, as depicted in Fig. 1. The initial steps of this workflow *extract assembly features* from standard CAD product models, and generate one or more *assembly tasks* for each feature. Each task is allocated to a workcell of the assembly system during *workcell allocation* (line balancing). Workcell configuration focuses on designing the layout and the behavior of an individual workcell, given the set of task to be executed in it. *Assembly process planning* is responsible for generating the optimal behavior: *sequencing the tasks* and *assigning them to resources* in such a way that a certain performance measure (e.g., the cycle time) is minimized. The computed plans are submitted to motion planning, and work instructions are generated for all resources: program code for robots, and work instructions for human workers.

In the sequel, it is assumed that a task can be executed by a robot, a human worker, or a combination of these two. In addition to the robot or human resources, appropriate tools and fixtures might be assigned to the task as needed.

In order to make a step towards automated assembly planning, this paper proposes a formal model of the assembly process, and presents an approach to the formal verification of the feasibility of assembly process plans from all points of view, including technological and geometric feasibility of the process.

3. Feature-based planning approach

During assembly two or more parts or sub-assemblies are joined in order to create a product or new sub-assembly. Various types of assembly operations are applied in present days production systems and most of them can be executed both by robots or manually. This section introduces the models of the assembly features in scope, the geometry, the surrounding environment (workcell) and the applied resources.

3.1. Modeling of part geometry

During planning part geometry will be modeled as triangle meshes. This approach does not utilize the advantages of descriptive CAD representations (e.g., native formats of CAD systems), however triangle meshes can be used efficiently for proximity queries in collision avoidance [15,16]. In addition, a common limitation on using native CAD formats is that they usually define constraints by using mating pairs and therefore assembly features with more than two components are not captured as one.

Considering rigid, homogeneous parts the volume, the mass, the center of mass can be calculated by using the mesh model. These physical properties of the part geometry have to be linked to the geometric model.

3.2. Modeling of assembly features

Assembly features implement *kinematic constraints* to join components. Since in the presented approach only rigid components are considered therefore only features that implement fixed kinematic pairs are in the scope, while gears, belt drives, etc. are excluded. It is assumed that the components to be assembled within a task do not affect the feasibility of it, i.e., the components are compatible. The approach presented in this paper aims to be generic and extendible, thus besides placing, insertion and screwing, RLW tasks are also modelled. The currently included features are shown in Fig. 2.

Placing and insertion determine the relative position of parts that were earlier independent. These will be referred to as *relative positioning feature types*. Other feature types (e.g., screwing, welding, etc.) create a permanent link between parts with momentarily fixed position. These will be named *permanent positioning feature types*. All permanent positioning features must be preceded by the relative positioning features between the parts that they join together.

We also assume that the sequence of tasks describes a monotonous assembly, i.e., there are no disassembly tasks (not even temporarily). Auxiliary tasks, such as put-away, material handling, etc. are ignored here, since these can be generated only after the assignment of assembly tasks to the workcells.

3.3. Modeling of technological parameters

Placing requires the end position of the component to be placed, which is described by the location and the orientation as a six-dimensional vector ($x, y, z, \alpha, \beta, \gamma \in \mathbb{R}^3$). The path of the component can be any collision-free path.

Insertion is described with the same parameters as placing, however the path is decomposed into two segments: the first segment is *placing* the component into a position which allows moving the component into the receiving component along a single axis movement. The reference frame attached to the component is defined so that the second segment of the movement (the actual insertion) is carried out parallel to its z axis. A safety distance d defines a clearance that separates the receiving geometry and the end of the first movement segment.