A Graph Based Hybrid Approach of Offline Pre-planning and Online Re-planning for Efficient Assembly under Realtime Constraints

Daniel Ewert[1], Sebastian Thelen[1], Ralph Kunze[1], Marcel Mayer[2],
Daniel Schilberg[1], and Sabina Jeschke[1]

[1] RWTH Aachen University
Center for Learning and Knowledge Management and
Institute of Information Management in Mechanical Engineering
Dennewartstrasse 27, 52068 Aachen, Germany
`jeschke.office@zlw-ima.rwth-aachen.de`
[2] RWTH Aachen University
Chair and Institute of Industrial Engineering and Ergonomics
Bergdriesch 27, 52056 Aachen, Germany
`info@iaw.rwth-aachen.de`

Abstract. Assembly tasks, e.g. the assembly of an automobile headlight, are a big challenge for nowadays planning systems. Depending on the problem domain, a planner has to deal with a huge number of objects which can be combined in several ways. Uncertainty about the outcome of actions and the availability of parts to be assembled even worsens the problem. As a result, classic approaches have shown to be of little use for reactive (online) planning during assembly, due to the huge computational complexity. The approach proposed in this paper bypasses this problem by calculating the complex planning problems, e.g. which parts must be mounted in which sequence, prior to the actual assembly. During assembly the precalculated solutions are then used to provide fast decisions allowing an efficient execution of the assembly. Within this paper this online planning combined with offline planning and the assessment of realtime constraints during assembly could be executed in the future will be described.

1 Introduction

1.1 The Polylemma of Production Technology

In the last years, production in low-wage countries became popular with many companies by reason of low production costs. Hence, competitive production engineering is particularly important for high-wage countries such like Germany. Competition between manufacturing companies in high-wage and low-wage countries typically occurs within two dimensions: the production-oriented economy and the planning-oriented economy [1].

H. Liu et al. (Eds.): ICIRA 2010, Part II, LNAI 6425, pp. 44–55, 2010.
© Springer-Verlag Berlin Heidelberg 2010

In the dimension of the production-oriented economy low-wage countries production compensates possible economic disadvantages within process times, factor consumption and process mastering by low productive factor cost. Companies in high-wage countries try to maximize the economies-of-scale by the usage of relatively expensive productivity factors. These disadvantages of relatively high-unit cost high-wage countries try to compensate through customizing and fast adaptations to market needs - the economies-of-scope. In addition the share of production within the value chain decreases more and more. Thus, the realizable economies-of-scale decreases, too. The escape into sophisticated niche markets is not promising, either.

Within the second dimension - the planning-oriented economy - companies in high-wages countries try to optimize processes with sophisticated, investment-intensive planning and production systems. Since processes and production systems do not reach their limit of optimal operating points, additional competitive disadvantages for high-wage countries emerge. In contrast, companies in low-wage countries implement simple, robust value-stream-oriented process chains [2].

The production industry in high-wage countries is confronted with these two dichotomies mentioned above. On the one hand there is the dilemma value orientation vs. planning orientation and on the other hand the dilemma scale vs. scope. These two dilemmas span the so called polylemma of production technology as shown in figure 1 [1].

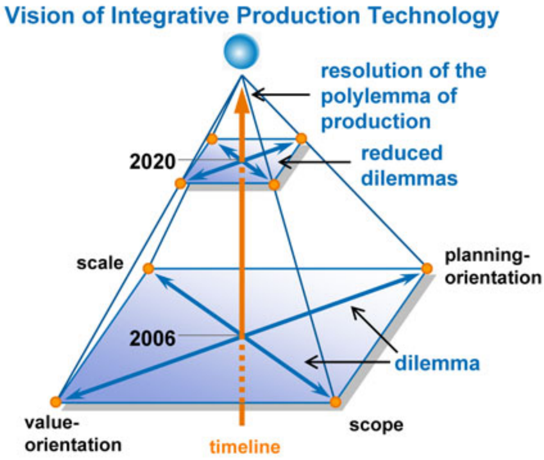

Fig. 1. The polylemma of production

1.2 Cluster of Excellence "Integrative Production Technology for High-Wage Countries"

The Cluster of Excellence "Integrative production technology for high-wage countries" unites several institutes of the RWTH Aachen University, which are dealing with production engineering. 19 professors of RWTH Aachen University, who work in materials and production technology, as well as affiliated institutes such as the Fraunhofer Institutes, work in this cluster with the objective to contribute to the maintenance of production in high-wage countries which is relevant for the labor market. With regard to economy, focus is on products that address both niche markets and volume markets. The solution for the problems addressed sometimes require a fundamental new understanding of product and production technological contexts.

The Cluster of Excellence aims for the development of basics for sustainable production-scientific strategy and theory, as well as for the necessary technological approaches. Based on the scientific analysis of the polylemma of production technology, the individualisation, virtualisation and hybridisation, as well as self-optimisation of production technology were identified as areas in need for action. Therefore, four major fields of research have been defined: Individualised Production Systems, Virtual Production Systems, Hybrid Production Systems and Self-optimising Production Systems. Institutions and participants of the same field of research become strategically consolidated in so called Integrative Cluster Domains (ICD) related to their competences in production technology at RWTH Aachen University. These ICD form the Aachen House of Integrative Production (see figure 2).

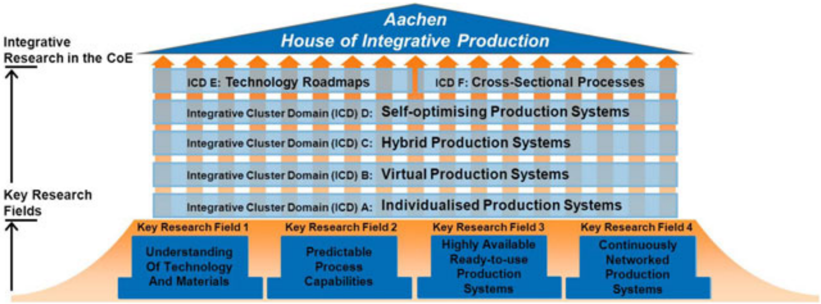

Fig. 2. The Integrative House of Production

This paper deals with cognitive planning and control systems of production which are addressed in a sub-project of the ICD-D “Self-optimising Production Systems”.

2 The Subproject “Cognitive Planning and Control System for Production”

2.1 Objective

An advanced rationalization of production systems and processes is typical for high-wage countries. A significant challenge is to involve the implementation of value stream-oriented approaches along with simultaneously increasing planning efficiency. A promising approach for the reduction of previous planning efforts consists of the development of a production system that is able to autonomously plan the production during a running process. Furthermore, such a system could autonomously react to changes in customer demands.

Within this sub-project a cognitive planning and control unit (CCU) is developed that aims to automate the assembly planning process, so that only a CAD description of the product which has to be fabricated is required as input to the system [3]. By means of this unit it will be possible to increase the flexibility of manufacturing and assembly systems and to decrease the planning effort in advance.

2.2 Approach

The task of the CCU is to plan and control the assembly of a product which is described solely by its CAD data. The system plans and executes the assembly autonomously, after receiving further descriptions, which are entered by a human operator. The CCU cooperates with human operators during the entire assembly process. While most of the assembly actions are executed by the assembly robot, certain tasks can only be accomplished by the operator.

The presented approach is evaluated within a scenario which is depicted in figure 3. The setup contains a robot cell with two robots, in which one robot (robot 2) is only controlled by the CCU [4], [5]. The other robot (robot 1) delivers separate parts for the final product in random sequence to a circulating conveyor belt. The CCU decides whether to pick up the delivered parts or to refuse them. If a part got picked up, the parts will be put into a buffer area or into the assembly area for immediate use.

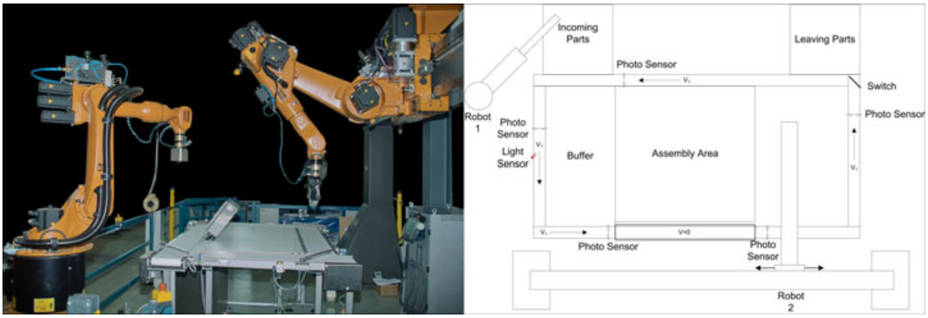

Fig. 3. Photo and diagram of the robot cell

Due to the random block delivery future states of the system cannot be predicted. The CCU is therefore facing a non-deterministic planning problem, requiring either an online re-planning during the assembly, whenever a not expected event occurs, or a plan in advance for all possible delivery sequences. Each of these strategies results in extensive computations, which lead either to slow responses during the assembly, or an unacceptable amount of pre-planning. The approach proposed in this paper therefore follows a hybrid approach, based on state graphs as described in the remainder of the paper.

3 Graph Based Planning under Realtime Constraints

3.1 Related Work

In the field of artificial intelligence, planning is of great interest. Several different approaches do exist, which are suitable for the application on planning problems. Hoffmann et al. developed the so called Fast-Forward Planning System, which is suitable to derive action sequences for given problems in deterministic domains [6]. Other planners are capable to deal with uncertainty [7], [8]. However, all these planners rely on a symbolic representation based on logic. Thus, the corresponding representations of geometric relations between objects and their transformations, which are needed for assembly planning, become very complex even for small tasks. As a result, these generic planners fail to compute any solution within acceptable periods of time.

Other planners have been designed especially for assembly planning. They directly work on geometric data to derive action sequences. A widely used approach is the Archimedes System [9] that uses AND/OR Graphs and an "Assembly by Disassembly" strategy to find optimal plans. U. Thomas follows the same strategy [10]. But where the Archimedes System relies on additional operator-provided data to find feasible sub-assemblies in his solution, Thomas uses geometric information of the final product as input only. However, both approaches are not capable of dealing with uncertainty.

3.2 Basic Idea

To allow fast reaction times during assembly the planning process is separated into an offline part(the Offline Planner, described in 3.3), executed prior to the assembly, and an online part(the Online Planner, described in 3.4), executed in a loop during the assembly. The resulting system is drafted in Figure 4. While the Offline Planner is allowed computation times of up to several hours, the Online Planner's computation time must not exceed several seconds. The task of the Offline Planner is the precalculation of all feasible assembly sequences leading from the single parts to the desired product. The output is a graph representing these sequences, which is transmitted to the Online Planner. During assembly the Online Planner then repeatedly maps the current system state to a state contained in the graph. Following, it extracts an assembly sequence that transforms this state into a goal state containing the finished product.

This assembly sequence is then handed to the Cognitive Control component which triggers the accordant robot commands, interacts with the human operator

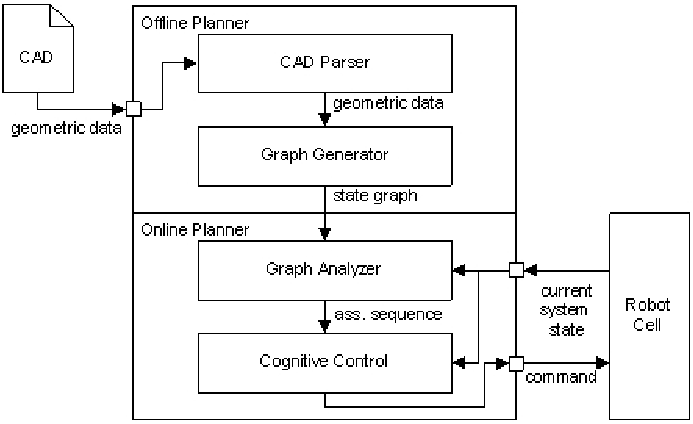

Fig. 4. System Overview of the CCU

and reacts to unforeseen changes in the state. Thereby, it is not bound to follow the given assembly, for example it can decide to invoke actions that move blocks from the conveyor belt to the buffer and vice versa instead of continuing an assembly. The details of the described process are discussed in the following sections in particular.

3.3 Offline Planning

Graph generation

During the offline planning phase, a solution space for the assembly sequence planning problem is derived from a product description. The current approach relies on a description of the assembled product's geometry and its constituting parts, possibly enriched with additional mating directions or mating operation specifications. Based on this information, an "assembly by disassembly" strategy is applied to ultimately generate a state graph that can be efficiently interpreted during online planning [11]. The main concept of this strategy is a recursive analysis of all possibilities of an assembly or subassembly, in which any assembly or subassembly is being separated into two further subassemblies until only nonseparable parts remain.

The geometric data is read by the CCU from a CAD file and stored in a tree data structure that is used to represent the product's assembly related properties. This assembly model tracks each atomic part that can not be disassembled further with a separate identity. These objects are reused in the assembly graph and reference all properties including the geometry of the represented part. To describe functional aspects like a thread or face of such a part, assembly feature instances can be used. This is relevant, if additional data beside the part geometries shall be taken into account by the assembly planner.

Evaluation of assembly separations

While the CCU logically disassembles the complete assembly, multiple calculations have to be performed on each separation:

- The feasibility of the separation has to be estimated.
- The assembly plan for the resulting assembly operation has to be calculated.
- A static cost for the assembly operation has to be assigned, which later will be utilized by the online planner or the graph pruning algorithms.

These operations are performed by an evaluation pipeline built from multiple *separation evaluators*. This enables later additions of evaluation operations and encapsulates all evaluation done on a single separation. In [10] an algorithm for pure geometrical inference on assembly parts is depicted. This algorithm is used for evaluation, if a separation can be performed at all. Furthermore, it delivers a set of assembly directions from which the assembly plan for an operation can be generated. Following Kaufman et al. [9], the results from the geometric inference are additionally enriched with possible userdefined overridings that have shown to be necessary for complicated geometries like screws or threads. They provide a way to drastically enhance the resulting assembly plans.

To enable this mixing of geometric inference and predefined mating directions, the geometric analysis of a mating operation is separated into two steps. In the first step, only the geometries of assembly parts which change contact state when the operation is performed are taken into account. Even if this processing yields no viable mating directions, but there is at least one that is predefined by the user, the processing is carried on without directly discarding the separation in graph building. The second, global geometric inference takes the complete subassemblies of the separation into account. If the geometric inference from the first step labeled user defined mating directions as blocked the responsible geometric features are masked from each other to suppress their lokal blocking behavior.

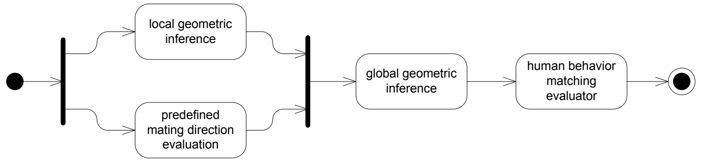

Fig. 5. The current computations performed to evaluate and rate a separation

The resulting evaluation pipeline is shown in figure 5. A separation is only excluded from the assembly graph if it is evaluated as being impossible to perform. This contrasts to other approaches like [10] and [12] where graph pruning heuristics are used to reduce computational demands and the utilized solution space. To enable the online planner to react upon unforeseen assembly states or events the complete solution space has to be processed. Though, high costs might be assigned to separations that should be avoided, as is discussed later in this section.

Where actual online planning relies on a state graph representing the solution space, the graph builder initially constructs an AND/OR Graph representing all viable assembly operations and states, that might appear [11]. A state graph, as shown in figure 6, is then derived from that graph in a second step. This conversion is necessary, because the planning algorithm which is used during the online phase is not capable to handle the AND/OR Graph, which is a directed hypergraph.

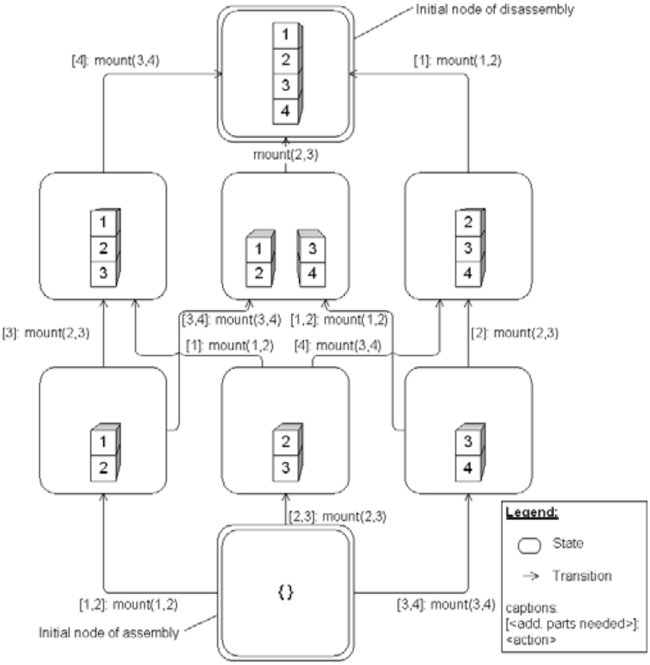

Fig. 6. Example of a state graph representation

All data, which is generated by the separation evaluators for a separation, is stored in the AND/OR Graph's respective hyperarc. Currently, this information contains static operation costs and mating operation descriptions for each assembly, that might be active (e. g. moved) during the assembly step represented by the edge. The state graph is created in such a way, that its transitions/edges are directed in the opposite direction of the AND/OR Graph's hyperarc they originate from. Each state only contains the passive assembly to which other parts may be added. The availability of the active parts is encoded as condition for a transition which contains the operations static cost and mating operation description.

Increasing machine transparency by mimicking human behaviour
Mayer et al. found that assemblies executed by human individuals follow - in most cases - simple rules [13]:

- Parts are placed next to each other. In other words, two separate subassemblies are not constructed in parallel.
- Structures are built layer by layer from bottom to top.
- Assembly begins at the boundary of the desired product.

During graph generation, each state is evaluated regarding its consistency with these (or similar) rules and receives an accordant rating. In the later phases of the online plan generation these ratings are taken into account and states with better ratings are preferred to other states. This strategy allows the CCU to mimic human behavior to a certain level.

3.4 Online planning

Responsibilities and Workflow of the Online Planner

The Online Planner is responsible for deriving plans during the assembly. Therefore it uses the state graph provided by the Offline Planner and current information about the robot cell's situation. This approach is similar to a system developed by [14], which reactively generates assembly task instructions for a human worker. The general process of the plan derivation from the state graph is depicted in figure 7.

The Graph Analyzer receives the generated state graph from the offline planning phase. It then has to pause until the Cognitive Control sends the actual world state describing, which contains the current situation of the assembly. The Graph Analyzer maps this world state onto the matching state contained in the state graph, now the "is-node". If the is-node is identical to the goal-node, the node containing the completed assembly, the assembly has been completed and the process ends. Otherwise, the current world state is compared to the previously received world state. If new objects have occurred since the last run, the graph is being updated. This process is detailed in the section "update phase". After the graph has been updated, the optimal path from the is- to the goal-node is calculated as described in section "sequence detection". The found path represents the optimal assembly plan for the given current situation. It is sent to the Cognitive Control component, which then executes the assembly.

Update phase

In this phase, dynamic costs are assigned to the state graphs edges. The cost of an edge depends on the static costs of the assigned action and costs depending on the availability of required parts. Here, robotic actions allocate lower costs than actions that must be executed by the human operator. Actions that rely on parts which are currently not available receive penalty costs. Due to this weighting of actions the Online Planner prefers currently realizable assemblies before assemblies that rely on parts, which are currently not present. However,

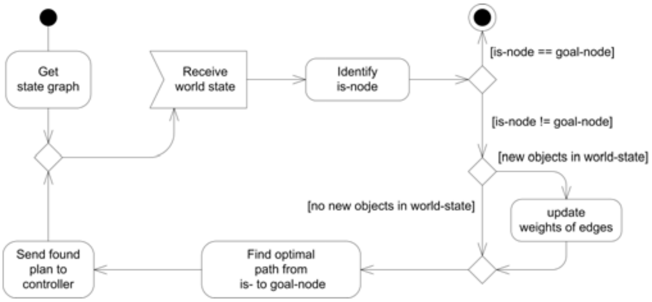

Fig. 7. Workflow of the Online Part

reducing the penalty costs so that it becomes possible for a sum of realizable actions to outweigh the cost of one unrealizable action introduces speculation in the planning process: The algorithm could now choose to prefer a short but yet not realizable sequence over a realizable that needs much more actions to fabricate the final product. This decision could turn out to be better, if the missing part is delivered to the assembly before the previously not realizable action is to be executed. This behaviour can be facilitated even more by assigning penalty costs depending on the distance of the action in question to the current state. Considering time: The further away the action to be executed, the lower are the penalty costs it retrieves.

Sequence detection

The algorithm identifies the node of the updated graph which is matching the current state of the assembly. This node becomes the initial node from which an optimal path to the final node has to be calculated. This is achieved using the A* search algorithm [15]. A* chooses nodes for expansion with a minimal value for $f(x) = g(x) + h(x)$, where $h(x)$ is a heuristic for the distance from node x to the goal node, and $g(x)$ represents the present path costs. Here, $h(x)$ takes into account the number of not correctly mounted parts as well as the "performance" of the node during its evaluation as described in 3.3. The path costs g are calculated as the sum of the costs of the traversed edges plus the costs for every necessary tool change along that path. The alteration between actions executed by the human operator and actions done by a robot also counts as a tool change, resulting in a preference for assembly sequences where human involvement is concentrated in single time slots.

3.5 Realisation of the assembly

The actual assembly is realised by the Cognitive Control component. It is based on Soar, a cognitive framework for decision finding that aims on modelling the human decision process [16]. It is responsible to invoke accordant actions based on the current world state received from the robot cell and the assembly sequence sent by the Graph Analyzer. The default behavior of the Cognitive Control is to realize the given assembly sequence by invoking the accordant robot command or instructing the operator. In case of unforeseen changes to the world state, due to a delivery of an object, it decides whether to discard or store the new object, depending on if the object can be used in later assembly steps. In such a situation the Cognitive Control can request a new plan from the Graph Analyzer which regards the new object. The component is also responsible for dealing with failure, e.g. failed robot actions or similar.

4 Outlook

Empiric studies need to show the capability of the described approach to accomplish feasible assembly sequences for different products of varying complexity. It is to be determined, which weightings of realizable and not realizable actions lead to optimal results. Furthermore, research on how much graph pruning influences the average computation time is required, and if the gain in time is worth the risk of missing an optimal assembly. Additionally, the scalability and applicability of the approach to industrial use cases has to be examined.

5 Conclusion

The polylemma of production can be alleviated by incorporating automatic planning systems. Depending on the assembly setup, these systems have to face nondeterministic behaviour during the assembly, which increases the computational complexity by orders of magnitude. The presented approach follows a hybrid approach of offine pre-planning and online re-planning to deal with this problem under realtime constraints. The actual planning problem is solved prior to the assembly by generating a state graph that describes all possible assemblies of the intended product. This graph is updated during the assembly and at this time optimal assembly sequence is extracted. This process is by far less time consuming and allows a fluent assembly. By taking into account the human behaviour during assembly it is possible to speed up the graph generation. Situations during an assembly that would not occur during an assembly executed by humans are pruned from the graph. This results in reduced computation times and increased machine transparency by meeting human expectancy.

Acknowledgement

The authors would like to thank the German Research Foundation DFG for supporting the research on human-robot cooperation within the Cluster of Excellence "Integrative Production Technology for High-Wage Countries".

References

1. Brecher, C., et al.: Excellence in Production. Apprimus Verlag (2007)
2. Klocke, F.: Production technology in high-wage countries – From ideas of today to products of tomorrow. Industrial Engineering and Ergonomics, 13–30 (2009)
3. Hauck, E., Gramatke, A., Isenhardt, I.: Einsatz Kognitiver Technischer Systeme im Produktionsumfeld. In: AUTOMATION 2008, Lösungen für die Zukunft. VDI Berichte, Band 2032, 305–308. VDI Verlag GmbH (2008)
4. Kempf, T., Herfs, W., Brecher, C.: Soar-based sequence control for a flexible assembly cell. In: ETFA 2009: Proceedings of the 14th IEEE international conference on Emerging technologies & factory automation, Piscataway, NJ, USA, pp. 488–496. IEEE Press, Los Alamitos (2009)
5. Hauck, E., Gramatke, A., Henning, K.: A Software Architecture for Cognitive Technical Systems Suitable for an Assembly Task in a Production Environment. In: AUTOMATION CONTROL – Theory and Practic, pp. 13–28. INTECH (2009)
6. Hoffmann, J.: FF: the fast-forward planning system. The AI Magazine 22 (2001)
7. Hoffmann, J., Brafman, R.: Contingent planning via heuristic forward search with implicit belief states. In: Proceedings of ICAPS 2005, pp. 71–80 (2005)
8. Castellini, C., Giunchiglia, E., Tacchella, A., Tacchella, O.: Improvements to sat-based conformant planning. In: Proc. of 6th European Conference on Planning, ECP 2001 (2001)
9. Kaufman, S.G., Wilson, R.H., Calton, R.E.J.T.L., Ames, A.L.: LDRD final report: Automated planning and programming of assembly of fully 3d mechanisms. Technical Report SAND96-0433, Sandia National Laboratorie (1996)
10. Thomas, U.: Automatisierte Programmierung von Robotern für Montageaufgaben. Fortschritte in der Robotik, vol. 13. Shaker Verlag, Aachen (2008)
11. Homem de Mello, L.S., Sanderson, A.C.: Representations of mechanical assembly sequences. IEEE Transactions on Robotics and Automation 7(2), 211–227 (1991)
12. Deshmukh, A., Yung, J., Wang, H.P.: Automated generation of assembly sequence based on geometric andfunctional reasoning. Journal of Intelligent Manufacturing 4(4), 269–284 (1993)
13. Mayer, M., Odenthal, B., Faber, M., Kabuš, W., Kausch, B., Schlick, C.: Simulation of human cognition in self-optimizing assembly systems. In: Proceedings of the IEA 2009 – 17th World Congress on Ergonomics (2009)
14. Zaeh, M.F., Wiesbeck, M.: A Model for Adaptively Generating Assembly Instructions Using State-based Graph. In: Manufacturing Systems and Technologies for the New Frontier, pp. 195–198. Springer, Heidelberg (2008)
15. Hart, P.E., Nilsson, N.J., Raphael, B.: A formal basis for the heuristic determination of minimum cost paths. Systems Science and Cybernetics. IEEE Transactions on Systems Science and Cybernetics 4(2), 100–107 (2007)
16. Laird, J.E., Newell, A., Rosenbloom, P.S.: Soar: An architecture for general intelligence. Artif. Intell. 33(1), 1–64 (1987)