Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 52 (2016) 130 – 133

www.elsevier.com/locate/procedia

Changeable, Agile, Reconfigurable & Virtual Production

Recommender system for acyclic graph generation in multi-agent assembly line planning

Daniel Neumann

Professorship of Production and Industrial Management, Technische Universität Chemnitz, Chemnitz, 09111, Germany
Tel.: +49-371-531-39986; fax: +49-371-531-639986. E-mail address: daniel.neumann@wirtschaft.tu-chemnitz.de

Abstract

Any assembly line planning (ALP) problem requires the knowledge of multitude of planners and is therefore the interface between the product-oriented and process-oriented planning process. The assembly line planning process (i.e., assembly task definition, time analysis, product-oriented assembly task sequencing, assembly line balancing and assembly line sequencing and scheduling) can be associated as the gateway between the fuzzy front end and the fuzzy back end of the innovation process. As ALP can be solved using solution search approaches, the formalization of the required information is the key to be able find at least a feasible, preferably a satisfying at most optimal solution. The input to solve the ALP are optimization objective(s), a precedence graph and further restrictions to consider. As decision making problems are widely researched, the collaboration for the precedence graph and the restriction modeling is regarded for the ALP.
As the Collaborative Precedence and Constraint Modeling (CPCM) has also to meet the requirements of the solution method. In this case genetic algorithms (GA) is used for finding solution to the problem. The used algorithm for solution search is fast and state of the art, but also requires uniform input data. Hence, only an acyclic precedence and constraint digraph is computable in an adequate way.
As soon multiple planners work jointly together within a virtual environment, only locking or conflict resolution approaches are offered to generate a computable precedence and constraint model for the solving an ALP using GA. As locking is a practical way from the technical point of view, it does not meet the requirements of the planners to model relevant precedence and constraint relations of the regarded elements (tasks, stations, resources and line segments). As a result a recommender system for transforming a cyclic digraph to an acyclic digraph and promoting solution improvements are presented based on network structure metrics of precedence, constraint and solution graph.

Peer-review under responsibility of the scientific committee of the Changeable, Agile, Reconfigurable & Virtual Production Conference 2016
Keywords: Recommender System; Precedence Graph; Assembly Line Planning; Collaboration; Constraint Modelling; Genetic Algorithm

1. Introduction

A recent development in assembly line planning is caused by new technological possibilities that occur through the industry 4.0 movement, the “ICT-fication” of things and the virtualization of processes. In addition, a new culture of information and knowledge sharing leaves two important footprints. First, the formalization of knowledge supports the abilities for virtual collaboration. Second, the information exchange by an increasing number of individuals allows a knowledge gathering which has not been regarded before. Both aspects make the evaluation of individual and collective knowledge for a specific problem set as well as the extraction of relevant knowledge from an existing database for a specific situation, a complex problem.

2. Collaborative Assembly Line Planning

2.1. Assembly Line Planning

Assembly lines are flow-line production systems originally developed for mass production of standardized products. In recent times, assembly lines are also important in mass customization. Especially in automobile manufacturing an individualization of products have to be taken into consideration to respond different costumer needs. Multi-purpose machines with automated tool swaps allow varying models with respect to not considerable setup costs [1].

In general, the productive units performing the operations are aligned serial. Workpieces visit these stations successively as they are moved along the line by using a transportation

Peer-review under responsibility of the scientific committee of the Changeable, Agile, Reconfigurable & Virtual Production Conference 2016
doi:10.1016/j.procir.2016.07.065

system, e.g. a conveyor mechanism [1]. These assembly systems are associated with high investment costs. Because of that, the configuration of an assembly line is very important. In this planning process, all decisions regarding equipment, alignment and capacity (number of stations, stroke cycle time, number of workers in a stroke) of stations for a given product or product group with a high number of operations and precedence relations have to be made before the production starts. In scientific literature, this configuration problem of assembly systems is called Assembly Line Planning (ALP) or Assembly Line Balancing (ALB) [1], [2].

Boysen, Fliedner and Scholl introduced a classification schema of the General Assembly Line Balancing Problem (GALBP). It is divided into three elements, which are noted as tuple [α, β, γ]. In this context α describes the characteristics of the precedence relation graph, β the characteristics of assembly lines and stations and γ the different planning objectives, e.g. minimal number of stations or minimal costs of assembly line configuration. In this regard, SALB characteristics are chosen as common reference for classification. That means that only deviations from that basic problem description must be provided by using the tuple notation. The presented classification scheme has been adopted from the machine scheduling scheme of Graham et al., which was used and refined by Brucker et al. for project scheduling [1], [2], [3], [4], [5].

In this context exists a wide gap between theoretical discussion and practical application. Most of the standard optimization models are not able to consider all practical aspects together with task-station assignment restrictions (combined tasks, assembly high, ergonomic and qualification aspects) and multiple planning objectives. Furthermore, different roles and actors are important within planning process. Involved persons or groups can be divided into process and product planner, logistics experts and material supply experts as wells as planners of pre-assembly lines [1], [2], [3].

The depicted problem class can be assign to Combinatorial Optimization because of the finite solution space. The resulting high complexity of these comprehensive planning problems interconnected with the problem size necessitates the use of capable solution methods. For practical occurring problems with more than thousand operations, the use of exact solutions methods is impossible for the purpose of computation time. In this case, heuristic methods can be used to get satisfying solutions with an acceptable computation time. Such methods can be divided into problem specific heuristics (constructive procedures) and metaheuristics (meta-strategies) [2], [6], [7].

Especially in the last decades, a large variety of heuristic approaches for different specifications of GALBP can be found in scientific literature. Constructive procedures are for example priority rule based approaches or incomplete enumeration approaches. In contrast to that, Genetic Algorithms (GA), Ant Colony Optimization (ACO) or local search strategies (e.g. Tabu Search or Simulated Annealing) can be assigned to metaheuristics.

2.2. Distributed Problem Solving

Any assembly line planning (ALP) problem requires the knowledge of multitude of planners within the innovation process and is therefore the interface between the product-oriented and process-oriented planning process. The assembly line planning process (i.e., assembly task definition, time analysis, product-oriented assembly task sequencing, assembly line balancing and assembly line sequencing and scheduling can be associated as the gateway between the fuzzy front end and the fuzzy back end of the innovation process.

Technical feasibility of the distribution of problem solving processes has been shown within the ICT research. The general approach for distributed problem solving lies in the decomposition of tasks into solvable subtasks, the allocation of tasks to appropriate agents, the accomplishment of tasks as well as the synthesis of the results.[8]

The aspects of distributed problem solving are presented within a taxonomy dividing aspects of control, i.e. cooperation, organization and dynamics, as well as aspects of communication, i.e. protocols, content and paradigms. [9]

The distributed constraint satisfaction problem (DCSP) requires a global structure, which is based on the concept of viral engineering, where information fragments are made available and processed by multiple agents [9], in form of directed weighted graphs as shown in [10]. This way the collaborative formalization for multi-objective optimization problems is split into subtasks whereas every agent models its assembly tasks and line structure constraints.

2.3. Collaborative Precedence and Constraint Modeling

A Genetic Algorithm for ALP requires an acyclic graph to be able to search for feasible balanced assembly lines (solutions). As it seems advantageous to integrate multiple planners within the precedence graph modelling process as well as in the constraint definition process, multiple graphs are modelled by the involved agents. As the independent modelling of multiple agents will eventually result in conflicts when merging the subgraphs to a computable precedence graph, as presented in [12], a conflict resolution for generating an acyclic precedence graph is required.

As constraints of ALP can be modelled as a graph as well the conflict analyzation can be carried out similar to those of the precedence graph. Hereby, not only tasks are represented by nodes but also stations, assembly line sections, tools, personnel or competencies as well as other relevant resources for the assembly line planning.

3. Recommender System

The synthesis of the subgraphs created by multiple agents allows on the one hand an independent and individual knowledge representation of every involved agent. On the other hand the problems arise after the modelling, while cycles generated throughout subgraph synthesis to one complete graph is not precluded.