European Journal of Operational Research 000 (2018) 1–14

Contents lists available at ScienceDirect

European Journal of Operational Research

journal homepage: www.elsevier.com/locate/ejor

Production, Manufacturing and Logistics

An asynchronous parallel disassembly planning based on genetic algorithm

Yaping Ren[a,b], Chaoyong Zhang[a,b,*], Fu Zhao[c], Huajun Xiao[a,b], Guangdong Tian[a,b,d]

[a] *School of Mechanical Science and Engineering, Huazhong University of Science and Technology, Wuhan, 430074 Hubei, China*
[b] *The State Key Lab of Digital Manufacturing Equipment and Technology, Huazhong University of Science and Technology, Wuhan, 430074 Hubei, China*
[c] *School of Mechanical Engineering, Division of Environmental and Ecological Engineering, Purdue University, West Lafayette, IN 47907, USA*
[d] *Transportation College, Jilin University, Changchun 130022, China*

A R T I C L E I N F O

Article history:
Received 13 April 2017
Accepted 30 January 2018
Available online xxx

Keywords:
Combinatorial optimization
Disassembly sequence planning
Asynchronous parallel disassembly
Operation time-dependent
Metaheuristics

A B S T R A C T

Disassembly is one of the most crucial remanufacturing activities. Disassembly sequence planning (DSP) is a combinatorial optimization problem and has been studied by many researchers. Conventional DSP techniques focus on sequential disassembly planning (SDP) in which only one manipulator is used to remove a single part or subassembly at a time such that it is inefficient when disassembling large or complex products. Recently, parallel disassembly has attracted some interest as it employs several manipulators to remove multiple components simultaneously. However, most of the work to date focuses on parallel disassembly techniques which require synchronization between manipulators, i.e., they must start their tasks simultaneously. This simplifies the modeling and analysis efforts but fails to fully realize the benefits of parallel disassembly. In this work, we propose asynchronous parallel disassembly planning (aPDP) which eliminates the synchronization requirement. In addition to precedence constraints, aPDP becomes highly operation time-dependent. To deal with this, we design an efficient encoding and decoding strategy for the disassembly process. In this paper, a metaheuristic approach, based on a genetic algorithm, is developed to solve the aPDP problem. The proposed algorithm is applied to four products which require disassembly processes of varying complexity, and the results are compared with two methods reported in literature. It is suggested that the proposed approach can identify faster disassembly processes, especially when solving large-scale problems.

1. Introduction

Disassembly plays a key role in the reuse and remanufacturing of end-of-life (EOL) products. Retrieving usable or repairable parts through disassembly preserves the geometry and function of the part, as well as the resources consumed during manufacturing of these parts. It has been argued that in comparison with other EOL product management options such as recycling, incineration, and landfilling, disassembly carries significant economic and environmental advantages (Li, Huang, Zhao, Sutherland, & Liu, 2017; Lu, Zhou, Xiao, Chang, & Tian, 2018; Ondemir & Gupta, 2014; Tang, Zhou, & Caudill, 2002b).

Disassembly sequence planning (DSP) is used to determine the best order for disassembling subassemblies, components, and parts from a product, according to some predetermined performance metrics (Tian, Zhou, & Chu, 2013). The disassembly of a product can be classified into two categories: sequential disassembly, where parts are removed one by one, and parallel disassembly, when multiple parts are removed simultaneously (Zhang et al., 2010). Consequently, DSP is categorized into sequential disassembly planning (SDP) and parallel disassembly planning (PDP). SDP is a classical problem that has been studied for decades (Tang, Zhou, Zussman, & Caudill, 2000). In SDP, only one part or subassembly is removed at a time. As a result, sequential disassembly follows a linear process and is relatively simple to model and optimize. However, because it is a linear process, sequential disassembly may lead to a long processing time, especially for large or complex products.

To address this issue, PDP has been developed. In contrast to SDP, more than one manipulator is employed to perform disassembly tasks and multiple parts can be removed at the same time. To fully utilize the potential of parallel disassembly to shorten processing time, coordination is needed between manipulators and the timing of different operations is extremely important. In pre-

* Corresponding author at: School of Mechanical Science and Engineering, Huazhong University of Science and Technology, Wuhan, 430074 Hubei, China.
E-mail addresses: renyp1@163.com (Y. Ren), zcyhust@hust.edu.cn (C. Zhang), fzhao@purdue.edu (F. Zhao), 1690113257@qq.com (H. Xiao), tiangd2013@163.com (G. Tian).

https://doi.org/10.1016/j.ejor.2018.01.055

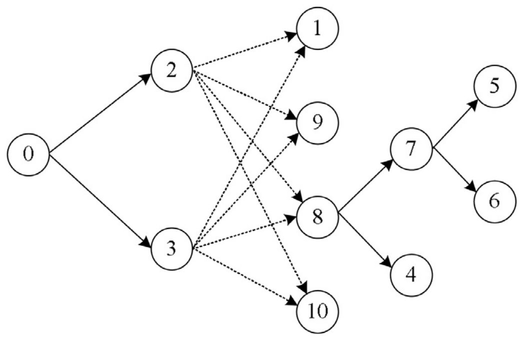

Fig. 1. The precedence diagram from McGovern and Gupta. Solid lines represent AND precedence, while dashed lines indicate OR precedence.

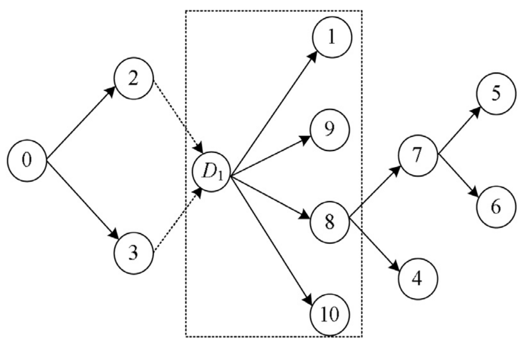

Fig. 2. A reduced precedence diagram of McGovern and Gupta example.

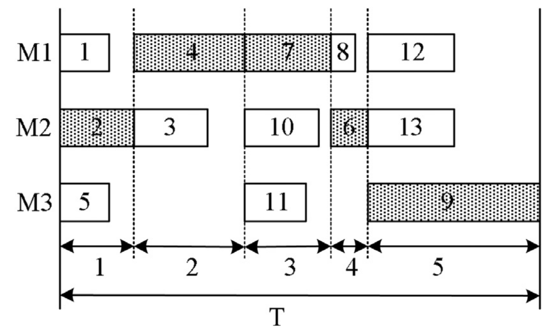

Fig. 3. A simple example of sPDP.

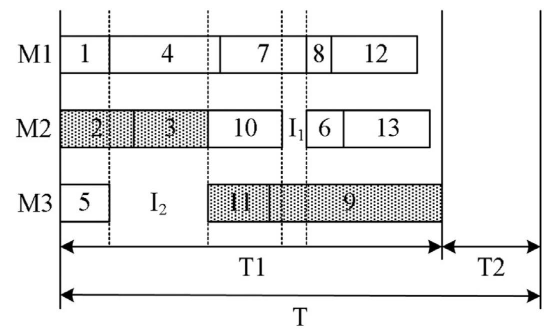

Fig. 4. The conversion from Fig. 3 showing aPDP.

vious studies, it is assumed that no manipulator can start a new task unless all other manipulators are ready for their next operation (i.e. the start time among manipulators is synchronized). This is called synchronous parallel disassembly planning (sPDP). This simplifies the problem but adds unnecessary idle time for some manipulators and may increase the total disassembly time. In this study, we propose asynchronous parallel disassembly planning (aPDP), which allows a manipulator to start its next task immediately, as long as precedence (and other) constraints are not violated.

The rest of this paper is organized as follows: Section 2 reviews the literature on disassembly planning. Section 3 describes precedence relationships in the disassembly process and presents sPDP and aPDP. Section 4 proposes an efficient metaheuristic based on a genetic algorithm (GA) to solve the aPDP problem. Section 5 shows the computational results of the proposed approach using four products with different complexities and compares our results with results obtained from methods reported in literature. Finally, Section 6 concludes our work and provides some future research suggestions.

2. Literature review

DSP can identify the best disassembly sequence/solution, among all feasible alternatives, to strip down assemblies into components without violating precedence relationships, while simultaneously maximizing economic and environmental benefits (Tang, Zhou, Zussman, & Caudill, 2002a). So far, many models and methods have been developed to solve DSP problems. Both mathematical programming and metaheuristic methods have been adopted. According to a simple disassembly graph, a linear programming model was developed to find the optimal disassembly sequence for maximum revenue (Lambert, 1999). Kang, Lee, Xirouchakis, and Persson (2001) used an integer programming model for disassembly sequencing, which was modified by applying precedence constraints to a shortest path problem. Lambert (2007) constructed an integer linear programming model for disassembly sequencing that was subject to sequence-dependent costs. In this work, an AND/OR graph was applied to graphically represent the complete set of possible disassembly operations, while an iterative method was proposed to identify the optimal solution based on a binary programming formulation.

The above papers used mathematical programming to solve DSP. Although the global optimum can be found, these methods often met challenges when the scale or the complexity of the problem increased, since DSP is a NP-hard problem (Adenso-Díaz, García-Carbajal, & Lozano, 2007; Lambert & Gupta, 2008). Some heuristic or metaheuristic methods have been developed to handle more complex problems, so that the near-optimal/optimal solution can be obtained within a reasonable time. Güngör and Gupta (2001) presented a branch-and-bound heuristic algorithm to produce disassembly sequence plans for the recycling and remanufacturing of products. A disassembly precedence matrix was defined to construct a hierarchical disassembly tree to generate feasible sequences. Another approach, which combined Petri net modeling with heuristic search procedures, was developed by Moore, Gungor, and Gupta (2001) and Rai, Rai, Tiwari, and Allada (2002). They generated disassembly sequence procedures by means of a disassembly Petri net. A heuristic was proposed for determining solutions which were ‘good enough’ for the disassembly sequencing problem; these solutions were then integrated with an exact algorithm based on an iterative method (Lambert & Gupta, 2008).

Among the metaheuristic methods, the most common approaches are intelligent algorithms, in particular, GAs. Seo, Park, and Jang (2001) adopted a disassembly AND/OR graph and developed a GA to find an optimal/near-optimal disassembly sequence. Kongar and Gupta (2006) considered three factors (i.e. disassembly demand, directions, and methods) to present a weighted multi-objective GA to solve a simple case. Hui, Dong, and Guanghong (2008) provided a disassembly feasibility information graph (DFIG) to illustrate the structure of the assembly. According to DFIG, A GA was proposed to quickly