Mechatronics xxx (2014) xxx–xxx

Contents lists available at ScienceDirect

Mechatronics

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

Real-time administration of tool sharing and best matching to enhance assembly lines balanceability and flexibility

Mohsen Moghaddam *, Shimon Y. Nof

PRISM Center & School of IE, Purdue University, West Lafayette, IN 47907, USA

ARTICLE INFO

Article history:
Received 31 January 2014
Revised 15 September 2014
Accepted 3 October 2014
Available online xxxx

Keywords:
Assembly line design
Collaborative assembly (CA)
Collaborative Control Theory (CCT)
Multi-agent system
Best matching

ABSTRACT

Collaboration has been found in previous studies on the design of assembly lines to be a useful mechanism. In this study, the focus is on a collaborative assembly (CA) framework, inspired by the design principles of CCT, the Collaborative Control Theory, to improve *balanceability* and *flexibility* of assembly lines through tool sharing (TS) among idle and bottleneck workstations. TS is widely practiced in advanced assembly facilities to reduce cost and improve consistency and standardization in assembly and in assembly-and-test utilities, relying often on real time control. The framework developed here addresses the systems design aspect of Mechatronics, covering the planning, execution, and control mechanisms. Planning includes assembly line balancing (ALB) and initial TS decisions, made continually by solving a bi-objective mixed-integer program (BOMIP). A collaborative multi-agent system (CMAS) enhanced with a TS-best matching (BM) protocol is developed to execute the plan, control the process, and modify the TS decisions, considering dynamic changes in the system's operations. Experiments show that the new CA framework significantly outperforms classic approaches (*i.e.*, ALB without TS-BM) in terms of cycle time, utilization of tools, and balanceability. In addition, the control mechanism is proven to augment the line's flexibility against the inherent uncertainties of assembly processes, compared to the previously developed *static* CA frameworks.

Published by Elsevier Ltd.

1. Introduction

An assembly line is an arrangement of *tools* placed in distinct workstations through which semi-finished assembly parts move to undergo a set of tasks. In this work, a tool refers to any human operator, robot or machine used to process the tasks, depending on the level of automation and type of assembly line. In an assembly line, each task is processed according to its precedence relations with other tasks, takes certain amount of workload to be processed, and cannot be subdivided between workstations [23]. In order to improve the efficiency of an assembly line, the overall workload must be balanced with respect to the demand and required production throughput [20]. The performance of an assembly line is typically evaluated according to the number of workstations or cycle time, as two primary criteria for assembly line balancing (ALB). Other criteria have also been considered in the literature, *e.g.*, workforce costs [29,24,12], efficiency of line and tools [28], and task duplication costs [7].

Various types of assembly lines exist [6] and several approaches have been developed in literature for solving ALB problems (see [13,4–6]). In a typical assembly line, the performance of the entire system is constrained by one or more overloaded workstations know as *bottleneck*. Despite the efforts made to optimize the efficiency, bottleneck is an inevitable phenomenon. Indivisibility of tasks among workstations is the primary cause of this phenomenon, which may be intensified by dynamic and unforeseen changes in the system's characteristics (*e.g.*, demand, processing times) over time. A bottleneck workstation defines the cycle time of the entire line, and diminishes the line's *flexibility* and *balanceability*. In this context, flexibility refers to the ability of the system to increase its throughput (despite dynamic changes such as disruptions in demand or supply) without utilizing any additional resources. Moreover, an assembly line is called balanceable, if the overall workload is (or can be) distributed equally between all workstations. In practice, however, fully balanceable assembly lines may be difficult or even impossible to achieve [1], since the task here is an atomic concept, it cannot be subdivided between workstations, and its processing time may vary dynamically. Hence, in spite of having undesirable bottleneck workstations, there exists a set of workstations with idle and underutilized tools.

* Corresponding author. Tel.: +1 (765) 494 5427.
E-mail addresses: mmoghadd@purdue.edu (M. Moghaddam), nof@purdue.edu (S.Y. Nof).

http://dx.doi.org/10.1016/j.mechatronics.2014.10.001
0957-4158/Published by Elsevier Ltd.

Nomenclature

Acronyms

ALB	assembly line balancing
BOMIP	bi-objective mixed-integer programming
CCT	Collaborative Control Theory
CMAS	collaborative multi-agent system
SN	scenario
TS	tool sharing
BM	best matching
CA	collaborative assembly
CE	collaboration efficiency
GAMS	general algebraic modeling system
TA	tool agent
WA	workstation agent

Indices

i, k	*Workstation*
n	*Task*

Parameters

D	demand rate
$\overline{C}$	cycle time upper bound
p_n	processing time of task n
$e_{ii'}$	CE for TS from i to i'
T	running time of the BOMIP model
$f_i(t)$	fraction of i's process finished up to t
$O_k(t)$	set of workstations that are not k's targets for TS and have more workload than k's targets at time t
$w_i(t)$	the overall workload of workstation i at time t
A	available production time
$\underline{W}$	number of workstations lower bound
IP_n	instant predecessor of task n
ξ	a very large positive number
$\alpha_i(t)$	progress rate of i at time t
X_k	k's target workstations for TS

Variables

W	number of workstations
M_{ni}	1, if task n and workstation i are matched; 0, otherwise
$S_{ii'}$	the time interval during which workstation i shares its tools with workstation i'
C	cycle time

In our previous works [17] – a brief, preliminary version from which this paper is expanded [18] – The general CA framework), we modeled a collaborative assembly (CA) framework, inspired by the design principles of Collaborative Control Theory (CCT) [21], for smoothing the workload of workstations and increasing the flexibility and balanceability of assembly lines. The notion of CA is based on sharing the tools of idle workstations with bottleneck workstations (see Fig. 1) such that the cycle time is minimized and the line balanceability is improved. Tool sharing (TS) is common in assembly, particularly in *Flexible Assembly* [20,22], *e.g.*, sharing of specialized insertion and bending tools; measurement instruments and positioning devices; and assembly robot precision and smart grippers. In assembly and test utilities, sharing of computer integrated testers and inspection tool is highly advantageous, especially when their actual use is only during a small fraction of the assembly process at each assembly station (*e.g.*, [11].

TS, as a collaborative mechanism that takes place among assembly workstations, leads to shorter manufacturing leadtime and lower work in process [10]. Work sharing is an alternative collaborative assembly approach, which is also related to the CA and discussed in literature [2,1,8]. In this approach, workstations are designed to share their tasks, rather than sharing tools, which may lead to some limitations in practice: (1) tasks cannot always be subdivided between workstations due to precedence relation constraints; (2) tools that are not designed to be shared (as specifically considered in this article) need to be duplicated, which imposes additional costs, often prohibitive when specialized tools, *e.g.*, assembly testers, must be duplicated (see, for example [11]; (3) tool redundancy increases, which reduces their utilization. The CA framework developed in this work, alternatively, improves the balanceability of assembly lines by using the *existing, effectively sharable* tools, while resolving the aforementioned limitations.

Previous studies by the authors on CA provide off-line planning models of collaborative TS and best matching (BM) decisions [17,18]. Such models solve the problems based on approximations and incomplete knowledge about the system's operations in real-time. These models have also been applied to analyze experimentally and prove the significant positive impact of assembly TS. Hence, the solutions provided by the off-line optimization models may be far from optimal under certain conditions in practice, for instance, in highly complex and chaotic systems. Thus, following the work by Moghaddam and Nof [17,18], the focus of this paper is on the design of mechanisms for monitoring and control of the TS and BM decisions in real-time. A collaborative multi-agent system (CMAS) is developed to enhance the automation of the CA framework. Intelligent and autonomous agents, distributed among the assembly system, operate in accordance with a TS-BM protocol. The real-time, collaborative control mechanism has been adapted and extended to provide feedback to the off-line plan generated and updated continuously using a bi-objective mixed-integer programming (BOMIP) model.

After this introduction and background, Section 2 presents the CA framework, including the BOMIP model for off-line planning

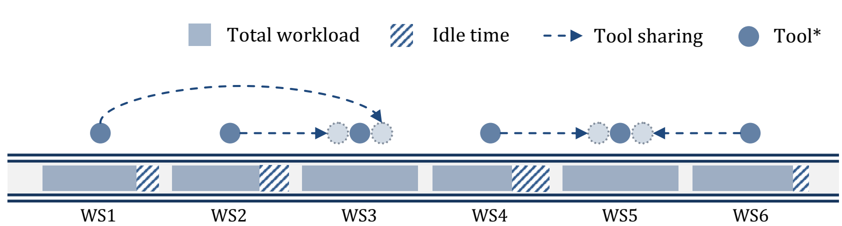

Fig. 1. TS in an assembly line with six workstations (WS). (* From the set of effectively sharable tools; the "Tool" symbols only represent the presence of the tools in the respective workstation(s), and not their exact locations.)

Please cite this article in press as: Moghaddam M, Nof SY. Real-time administration of tool sharing and best matching to enhance assembly lines balanceability and flexibility. Mechatronics (2014), http://dx.doi.org/10.1016/j.mechatronics.2014.10.001