European Journal of Operational Research 236 (2014) 774–786

Contents lists available at ScienceDirect

European Journal of Operational Research

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

Innovative Applications of O.R.

Mathematical analysis and solutions for multi-objective line-cell conversion problem

Yang Yu [a], Jiafu Tang [a], Jun Gong [a], Yong Yin [b], Ikou Kaku [a,c,*]

[a] *Institute of Systems Engineering, State Key Laboratory of Synthetic Automation for Process Industries, Northeastern University, Shenyang, China*
[b] *Department of Economics and Business Management, Yamagata University, Yamagata, Japan*
[c] *Department of Enviromental and Information Studies, Tokyo City University, Yokohama, Japan*

ARTICLE INFO

Article history:
Received 13 February 2012
Accepted 7 January 2014
Available online 20 January 2014

Keywords:
Assembly cell
Assembly line
Line-cell conversion
Multi-objective optimization
Genetic algorithm
Seru

ABSTRACT

The line-cell (or line-*seru*) conversion is an innovation of assembly system applied widely in the electronics industry. Its essence is tearing out an assembly line and adopting a mini-assembly unit, called *seru* (or Japanese style assembly cell). In this paper, we develop a multi-objective optimization model to investigate two line-cell conversion performances: the total throughput time (TTPT) and the total labor hours (TLH). We analyze the bi-objective model to find out its mathematical characteristics such as solution space, combinatorial complexity and non-convex properties, and others. Owing to the difficulties of the model, a non-dominated sorting genetic algorithm that can solve large size problems in a reasonable time is developed. To verify the reliability of the algorithm, solutions are compared with those obtained from the enumeration method. We find that the proposed genetic algorithm is useful and can get reliable solutions in most cases.

1. Introduction

Today's business can be described by a single word: turbulence. Turbulent markets have the following characteristics: short product life cycles, uncertain product types, and fluctuating production volumes. To compete, sometimes only to survive, in such a volatile business environment, a number of approaches have been developed to aid companies for their management decisions and engineering designs (Yin, Kaku, Tang, & Zhu, 2011).

The line-cell (or line-*seru*) conversion, conceived at Sony, is an innovation of assembly system used widely in the Japanese electronics industry. To compete in a turbulent market, in 1992, several mini-assembly units were created in one of Sony's video-camera factories for an 8-millimeter CCD-TR55 video-camera, after dismantling a long assembly conveyor line. As did the original conveyor line, each mini-assembly unit produced the entire product. In 1994, Tatsuyoshi Kon, a former Sony staff, called this mini-assembly organization *seru*, a Japanese word for cellular organism. A detailed introduction of *seru* system and its managerial mechanism can be found in Yin, Stecke, and Kaku (2008) and Stecke, Yin, Kaku, and Murase (2012). *Seru* is similar to assembly cells, a widely adopted assembly system in western industries. Equipment, however, is less important for *seru*. As a human-centered assembly system, *seru* is an old-fashioned workshop where craftsperson, including jack-of-all-trades workers, assembles an entire product from-start-to-finish by her- or himself. This mini-assembly organization is regarded as an ideal combination of lean and agile production paradigms (Yin, Stecke, Swink, & Kaku, 2012).

There are three *seru* types: divisional *seru*, rotating *seru*, and *yatai*. A divisional *seru* is a short line staffed with several partially cross-trained workers. Tasks within a divisional *seru* are divided into different sections. Each section is in the charge of one or more workers. On the other hand, workers staffed within rotating *serus* or *yatais* are completely cross-trained. A rotating *seru* is often organized in a U-shaped layout with several workers. Each worker assembles an entire product from-start-to-finish without disruption. The assembly tasks are performed on fixed stations, so workers walk from station to station. A *yatai* is a single worker *seru*, the smallest production organization. So a *yatai* owner does all operational and managerial tasks by her- or himself (Liu, Lian, Yin, & Li, 2010; Liu, Stecke, Lian, & Yin, 2014; Stecke et al., 2012; Yu, Gong, Tang, Yin, & Kaku, 2012; Yu, Tang, Sun, Yin, & Kaku, 2013a, 2013b). In this paper, we only analyze rotating *serus* and *yatais*, and leave the analysis of divisional *serus* as a future research topic. To be consistent with previous research (Kaku, Gong, Tang, & Yin, 2008, 2009), from Section 2, we use "assembly cell" in this paper to represent *seru* (i.e., Japanese style assembly cell) and call the conversion from assembly lines to *serus* "line-cell conversion".

---

[*] Corresponding author at: Department of Enviromental and Information Studies, Tokyo City University, Yokohama, Japan. Tel.: +81 45 910 2904; fax: +81 45 910 2605.

E-mail address: kakuikou@tcu.ac.jp (I. Kaku).

http://dx.doi.org/10.1016/j.ejor.2014.01.029
0377-2217/© 2014 Elsevier B.V. All rights reserved.

A *seru* system, which consists of one or more *serus*, is more flexible and agiler than the assembly line. To improve the flexible of assembly lines, Guo, Wong, Leung, and Fan (2009) proposed an intelligent production control decision support system to solve the flexible assembly line (FAL) problem with flexible operation assignment. In addition, the *seru* system has a better balance than assembly line, because in *seru* system the balanced capacity can be improved by the workers assignment (i.e., cell formation) and batches assignment (i.e., cell loading) (Yu et al., 2012). To improve the balance of assembly line, Guo, Wong, Leung, Fan, and Chan (2008) used a genetic algorithm based optimization model to solve the assembly line balancing problem.

To respond to the uncertain demands from a turbulent market, a new management principle – Just-In-Time Organization System (JIT-OS) is used to manage a *seru* system. JIT-OS is different from the traditional, well-known Just-In-Time Material System (JIT-MS) of lean manufacturing. Yin et al. (2008) and Stecke et al. (2012) defined JIT-MS and JIT-OS as follows.

- "*JIT-MS is a system that can provide the correct materials, in the right place, at the appropriate time, in the exact amount.*"
- "*JIT-OS is a system that can provide the correct serus, in the right place, at the appropriate time, in the exact amount.*"

So JIT-OS is an extension or upgrade of JIT-MS from materials to organizations (i.e., *serus*). Most *serus* within a *seru* system can be constructed, modified, dismantled, and reconstructed frequently in a short time. This is a prerequisite for implementing a JIT-OS. For a detailed introduction of JIT-OS, see Stecke et al. (2012); for application cases of JIT-OS in Sony and Canon, see Yin et al. (2008) and Yin et al. (2012).

To fit the specific layout of a *seru* factory, appropriate case-by-case approaches are usually used to adjust the floor space of *seru* factories. Fortunately, since *serus* can be modified, dismantled, and constructed easily and quickly, and most *serus* occupy small spaces (e.g., mini units), managers can often get huge benefits from the adjustment of floor spaces. For example, by adopting *seru* system, Canon and Sony reduced 720,000 and 710,000 square meters of floor space, respectively (Stecke et al., 2012; Yin et al., 2012).

Cost can also be reduced largely by using *seru* systems. After adopting *seru* systems, Canon's costs were reduced significantly, by 55 billion yen in 2003, and by a total of 230 billion yen from 1998 to 2003. As a result, Canon emerged as a leading electronics maker. Its average productivity is higher than that of Toyota (Yin et al., 2008; Yin et al., 2012).

Other benefits from *seru* systems include (Stecke et al., 2012; Takeuchi, 2006) the reductions of throughput time, setup time, required labor hours, WIP inventories, and finished-product inventories. This paper analyzes two of *seru* performances: reductions in throughput time and required labor hours. Some amazing cases related to these two *seru* performances are, the throughput time was reduced by 53% at Sony Kohda and 35,976 required workers, equal to 25% of Canon's previous total workforce, have been saved (Yin et al., 2012). To investigate these two *seru* performances simultaneously, we adopt a bi-objective optimization model to study interactive relationships between them.

This paper, originally motivated by line-cell applications of Sony and Canon, has two purposes. First, we clarify several mathematical characteristics of the line-cell conversion problem. For simplicity and without loss of generality, we modify the model of Kaku et al. (2008) into a simple case in which a conveyer assembly line is converted to a pure *seru* (or cell) system. Second, we solve the developed multi-objective optimization problem. Two objectives of the total throughput time and the total labor hours are minimized simultaneously by using a NSGA-II based algorithm. We also use several numerical examples to illustrate the usefulness of our approach.

The remainder of this paper is organized as follows. The modified model is presented in Section 2. Section 3 clarifies several mathematical characteristics of the line-cell conversion problem from the modified model. Then a NSGA-II based algorithm is used to solve the multi-objective line-cell conversion problem in Section 4. Several numerical simulation experiments are performed to show the usefulness and performance of our approach in Section 5. Finally conclusions are given in Section 6.

2. A modified model of the line-cell conversion problem

2.1. Problem description

Kaku et al. (2008) compared three types of assembly systems: a pure cell system, a pure assembly line, and a hybrid assembly system that consists of an assembly line and several cells. For simplicity and without loss of generality, this paper studies a line-cell conversion problem shown in Fig. 1, i.e., an assembly conveyor line is converted to a pure cell system. All workers who formerly worked within the assembly line are assigned to cells (we name it as "assembly cell formation").

A robust JIT-OS is the key for implementing a successful cell system. One important problem for designing a JIT-OS is to schedule or assign customer orders to different cells. We call this problem "assembly cell loading (ACL)". Unfortunately, Yin, Stecke, Li, and Kaku (2011) have proved that even a simple ACL problem is NP-hard. In this paper, we adopt a First Come First Serve (FCFS) principle that applied in many companies. An arriving product batch is assigned to the empty cell with the smallest cell number. If all cells are occupied, the product batch is assigned to the cell with the earliest finish time. Fig. 2 shows a FCFS cell loading example with six batches and two cells. The length of rectangle charts in Fig. 2 is the flow time of a product batch.

We evaluate two line-cell conversion performances: throughput time and required labor hours, which have been reduced dramatically by *seru* users (e.g., 53% throughput time at Sony, 25% required workforce at Canon, respectively). Therefore, our problem is to decide how many cells should be formed, how to assign workers and product batches to appropriate cells to minimize two objectives, i.e., the total throughput time (TTPT) and the total labor hours (TLH).

2.2. Assumptions

Following assumptions are considered in this paper to construct the model of a pure cell system:

1. The types and batches of products are known in advance. There are N product types that are divided into M product batches. Each batch contains a single product type.
2. In the line-cell conversion process, the cost of duplicating equipment is ignored. Since most assembly tasks within a *seru* are manual so need only simple and cheap equipment. Duplicating them is usually not costly (Stecke et al., 2012; Yin et al., 2012).
3. A product batch needs to be assembled entirely within a single cell. In other words, a batch cannot be shared by cells.
4. Every assembly task is performed within its exclusive station. If a product type does not need some task, the product skips the task's station.
5. The assembly tasks within each cell are the same as the ones within the assembly line. In this paper, the number of tasks equals to W.