J Intell Manuf (2009) 20:751–760
DOI 10.1007/s10845-008-0174-5

# A performance comparison between Kanban and CONWIP controlled assembly systems

**Yaghoub Khojasteh-Ghamari**

Received: 2 September 2007 / Accepted: 25 August 2008 / Published online: 9 September 2008


**Abstract** There exists controversy on the superiority of logistics control systems. Kanban and CONWIP systems are focused on and analyzed in this paper. CONWIP is a well-known production control system, and some papers have shown it has better performance than the Kanban system. Our research shows that the Kanban is more flexible for the assembly system under concern with respect to a given objective than the CONWIP. In some cases, if the number of kanbans at each manufacturing/assembling station is optimally set, Kanban system outperforms CONWIP with a lower average WIP and the same level of throughput. That is, the distribution of kanbans can be an important design parameter of the system. We also propose two different policies to release cards in a CONWIP controlled assembly system, followed by their comparison results.

**Keywords** Assembly systems · Production control systems · Kanban · CONWIP

## Introduction

In a "pull" production control system, the start of a job is triggered by the completion of an earlier job. Control of work-in-process (WIP, for short) becomes easier and hence can be significantly reduced in a pull system (Monden 1998). The most well-known pull mechanism is a Kanban system. In a Kanban control system, instead of directly controlling the throughput, kanbans (cards) are used to authorize production or transportation of materials such that the parts are pulled and WIP is visualized and controlled. The advantage of this system is that the number of parts in every stage is limited by the number of kanbans of that stage.

CONstant work In process (CONWIP) control system proposed by Spearman et al. (1990) uses a single card type to control the total amount of WIP permitted in the entire line. It is a generalization of the Kanban system and can be viewed as a single stage Kanban system. A CONWIP system behaves as follow: when a job order arrives to a CONWIP line, a card is attached to the job, provided cards are available at the beginning of the line; otherwise, the job must wait in a backlog. When a job is processed at the final station, the card is removed and sent back to the beginning of the line, where it might be attached to the next job waiting in the backlog. No order can enter the line without its corresponding card. The primary difference between CONWIP and Kanban systems is that CONWIP pulls a job into the beginning of the line and the job goes with a card between workstations, while Kanban pulls jobs between all stations (Hopp and Spearman 2001).

There are many studies on control policies for manufacturing systems. Spearman et al. (1990) proposed that the CONWIP concept could be applied to an assembly system fed by two fabrication lines. Hopp and Roof (1998) studied such assembly systems using statistical throughput control (STC) method, for setting WIP levels to meet target production rates in the CONWIP system. Duri et al. (2000) developed an approximation method to obtain some performance measures in three stage production lines under CONWIP control policy with random processing time and random inspection. Framinan et al. (2000) studied the input control and dispatching rules that might be used in a flow shop controlled by the CONWIP system within a make-to-stock environment. Cao and Chen (2005) developed a nonlinear mixed integer programming model for a CONWIP based production system where an assembly station is fed by two

Y. Khojasteh-Ghamari (✉)
Graduate School of Systems and Information Engineering, University of Tsukuba, Ibaraki 305-8573, Japan
e-mail: khojast@sk.tsukuba.ac.jp; khojastey@gmail.com

parallel fabrication lines. Optimal part assignment, production sequence and lot sizes are simultaneously determined by solving the model.

Hopp and Spearman (1991); Duenyas and Hopp (1992, 1993); Duenyas (1994) and Hazra and Seidmann (1996) addressed the application of CONWIP control to assembly operations. The analyses used in each of these references rely on queueing network approximations in computing the throughput. Hopp and Spearman (1991) approximated the throughput of a flow-shop under CONWIP control. They assumed that processing times are deterministic, but service can be interrupted by machine failures that are exponentially distributed in duration. Duenyas and Hopp (1992, 1993) approximated the throughput of an assembly system consisting of multiple station tandem production lines, feeding an assembly operation under the CONWIP control. Duenyas (1994) generalized this approximation to a cyclic assembly system with general processing time distributions. His approach is similar to that of Duenyas and Hopp (1992). Hazra and Seidmann (1996) considered closed tree structured assembly systems with exponential machine processing times, and developed an aggregation/disaggregation algorithm to approximate the system throughput and mean queue lengths at the workstations. Ip et al. (2007) compared single-loop and multi-loop CONWIP production control systems in a lamp assembly production line producing different kinds of products with discrete distribution processing time and demand.

There are also some studies on comparing Kanban and CONWIP systems to determine the superior one. Sato and Khojasteh-Ghamari (2008) developed an integrated framework for analyzing performance of card-based production control systems. They provided comparative results between Kanban and CONWIP systems. Their analytical results showed that in a serial production process, CONWIP outperforms Kanban, when the total number of cards in CONWIP is less than that in Kanban. Their analysis is based on the theory of token transaction systems. They also provided analytical results for comparing these two control systems in a tree-shaped production process. In a survey paper, Framinan et al. (2003) reviewed comparison of CONWIP with other production control systems. According to this survey paper, in comparison of Kanban and CONWIP, many authors have shown through both simulation and analytical models that CONWIP outperforms Kanban when processing times on component operations in production processes are variable. In a flow line that produces a single part type, Spearman and Zazanis (1992) showed that CONWIP produces a higher mean throughput than Kanban. In the same scenario, Muckstadt and Tayur (1995a,b) showed that CONWIP produces a less variable throughput and a lower maximal inventory than Kanban. In a case study, Huang et al. (1998) compared the CONWIP system and the original control system for four situations in a cold rolling plant. They showed that the CONWIP system has better performance than Kanban, with lower WIP and higher throughput rate. Takahashi et al. (2005) compared Kanban, CONWIP and synchronized CONWIP in a tree-stage production process with respect to two performance measures, WIP and backlog, in supply chains consist of assembly stages with different lead times. Their simulation results showed the superiority of both CONWIP and synchronized CONWIP over Kanban, when all inventory levels among the three stages are equally important.

Gstettner and Kuhn (1996), however, arrived at the opposite conclusion. According to their results, Kanban achieves a given throughput level with less WIP than CONWIP. They showed that by choosing an appropriate number of cards at each station, Kanban can outperform the CONWIP system. They considered a *serial production line* with exponential service time distributions and unlimited demand at the final buffer. In this paper, we verify whether their result holds for *assembly production processes* under the same assumptions. In other words, we verify the role of card distribution in assembly systems whether Kanban outperforms CONWIP by choosing a proper distribution of cards within the Kanban system. Moreover, we propose two different policies to release cards into a CONWIP controlled assembly system followed by their comparison results, in order to find out the superior policy which provides the system with a less WIP given the same rate of throughput.

The remainder of this paper is organized as follows. Section "Problem description" describes the considered model and assumptions. Section "Kanban and CONWIP concepts in assembly systems" details concepts of Kanban and CONWIP in assembly systems. Two proposed CONWIP control policies are also presented in this section. Simulation results for comparison between two proposed CONWIP control policies are given in section "Analysis of SCB and DCB CONWIP policies". Section "Comparison of Kanban and CONWIP" presents the simulation results of comparing the performance of Kanban and CONWIP systems with respect to the average WIP. The role of card distribution in a Kanban system is also verified in this section. Section "Conclusions" discusses the conclusions drawn from the experiments as well as some suggestions for future research.

## Problem description

In this research, we assume an assembly production process in which the assembly of parts proceeds in three stages. The third stage is an assembly line with several workstations, such that the first one is fed by sub-assembly lines in stage 2. The first station of sub-assembly lines are fed by fabrication lines in the first stage. Each workstation is a production (or assembly)/inventory system made up of a manufacturing (or

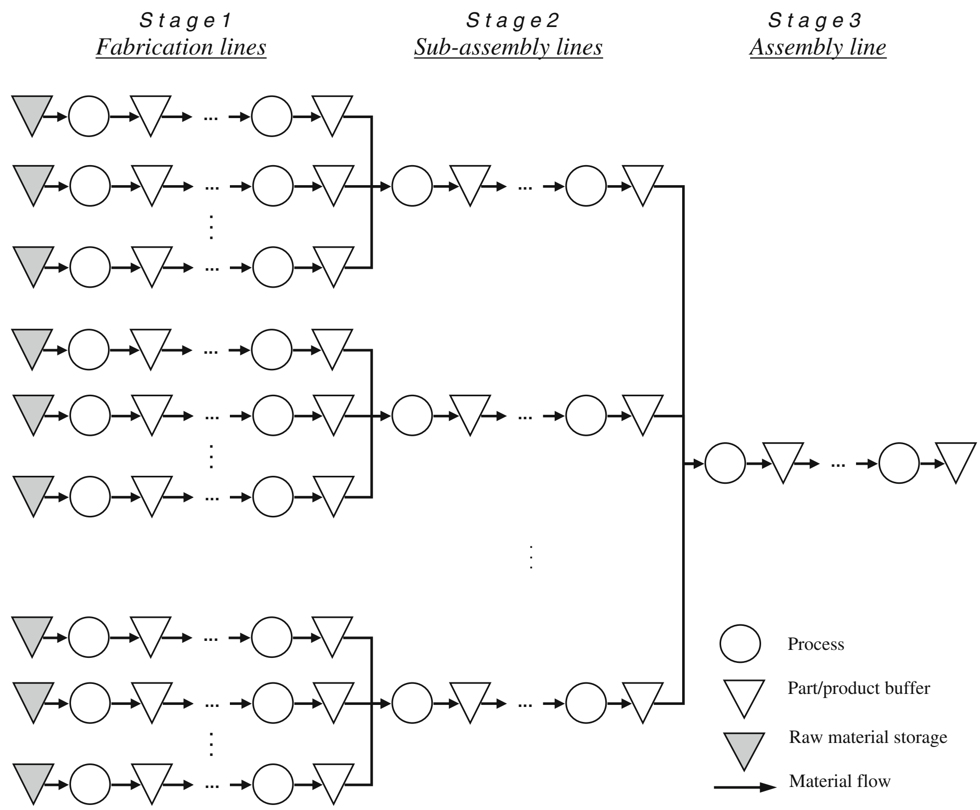



**Fig. 1** An assembly production system with three stages

assembly) process and an output buffer. The manufacturing process may consist of a single machine or a sub-network of several machines. Figure 1 shows the schematic model. The manufacturing/assembling processes at each stage are drawn as circles, the intermediate and output buffers as triangles, and raw material buffers are drawn as shaded triangles. Solid lines represent material flows.

We make some common assumptions across the two policies:

- The system makes a single part type.
- Material is transported in units of one without delay.
- Information, such as kanbans, is transmitted instantly.
- Machines operate asynchronously, so parts can be loaded whenever a part is present and the proper authorization has been received.
- Jobs authorized for loading follow a first-come, first-serve (FCFS) dispatching policy at all stations.

There are several reasons for making these assumptions: one is to simplify our study by reducing the number of variables considered. Another reason is that several of the variables we left out, such as kanban transmittal time and batch size, can be seen as implementation details rather than essential aspects of the different control policies. By creating an ideal implementation, we will illuminate the inherent behaviour and limitations of the control policies. Finally, these assumptions are not essential to our control policies, and they do not influence the conclusions drawn.

We also make the same assumptions considered by Gstettner and Kuhn (1996). That is, the first stations of all fabrication lines are never starved, and there is an unlimited demand at the end of the assembly line in the final stage (i.e., finished products leave the final buffer immediately).

We verify the role of card distribution in assembly production systems whether Kanban outperforms CONWIP by choosing a proper distribution of cards within the Kanban system. The objective is to compare performance of the Kanban and CONWIP systems with respect to the average WIP to determine the superior system.

## Kanban and CONWIP concepts in assembly systems

In this section, we describe the operation and control characteristics of Kanban and CONWIP control systems in assembly production processes. For both systems, we first give a brief description as the general concept. Then, we propose two different policies to release cards in a CONWIP controlled assembly system. For the sake of simplicity, the activity interaction diagram (AID, for short) of each control policy is shown in an assembly system having two subassembly lines each of which is fed by two fabrication lines.

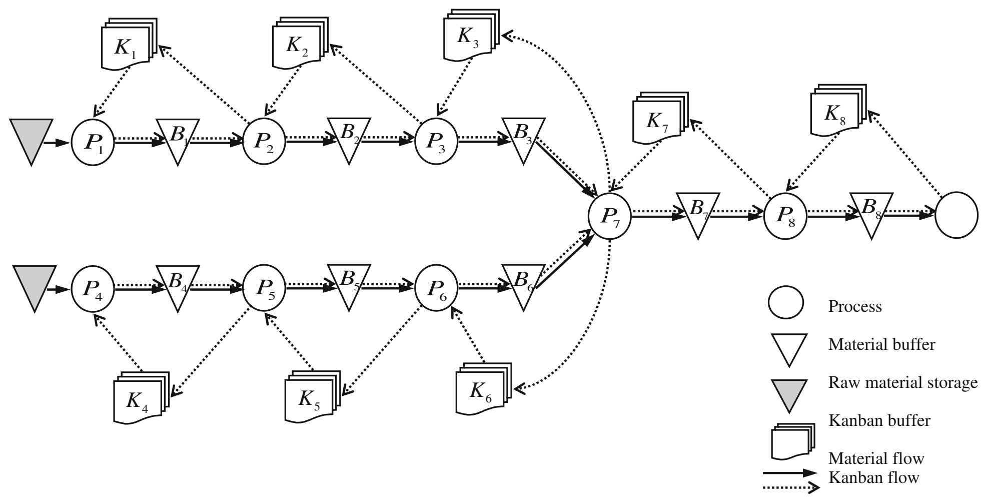



**Fig. 2** Kanban controlled assembly system

A multi-line system can be modeled based on these simple models.

Kanban controlled assembly system

Let us consider an assembly line fed by two fabrication lines each of which consists of three workstations. The AID of this model under the Kanban control system is depicted in Fig. 2. The manufacturing/assembling processes at each stage are drawn as circles, the intermediate and output buffers as triangles, and raw material buffers as shaded triangles. Queue $K_i$ contains station $i$ kanbans. Queue $B_i$ is the output buffer of station $i$ containing both finished parts and station $i$ kanbans. Solid lines represent material flows and the kanban movement is shown by the dotted lines. Once a finished product comes out of the assembly line ($B_8$), a signal kanban is sent back to the upstream stations in series to pull the parts along the assembly line as well as to pull the parts required for the assembly station ($P_7$) through the two fabrication lines.

The Kanban control is a simple control mechanism that depends only on one parameter per workstation: the number of kanbans at workstation $i$, $k_i$, $i = 1, \ldots, N$, where $N$ is the total number of workstations in the system. These parameters influence both the transfer of finished parts downstream through the system and the transfer of demands upstream through the system. Let $Q(q)$be the number of items in queue $q$. The invariant of Kanban mechanism of each workstation can be expressed as follow.

$$Q(K_i) + Q(P_i) + Q(B_i) = k_i, \quad i = 1, \ldots, N$$

where, $K_i$ is the station $i$ kanban queue containing station $i$ free kanbans, $P_i$ the manufacturing/assembly process of station $i$, $B_i$ the output buffer of station $i$, and $k_i$ the total number of kanbans in station $i$.

This also implies that both the WIP and the number of finished parts in each workstation $i$ are bounded by $k_i$.

CONWIP controlled assembly system

In the simplest implementation of a CONWIP system, a new job is not started in a line until an existing job exits the line and jobs are pushed along the line in first-come, first-serve sequence (FCFS). CONWIP system maintains a WIP level upper bound for the entire system. When the preset WIP level is reached, no new jobs are authorized to release to the system before some job leaves. A CONWIP line can be seen as controlled by a single Kanban cell encompassing all stations. CONWIP control is indeed considered as a single-station control.

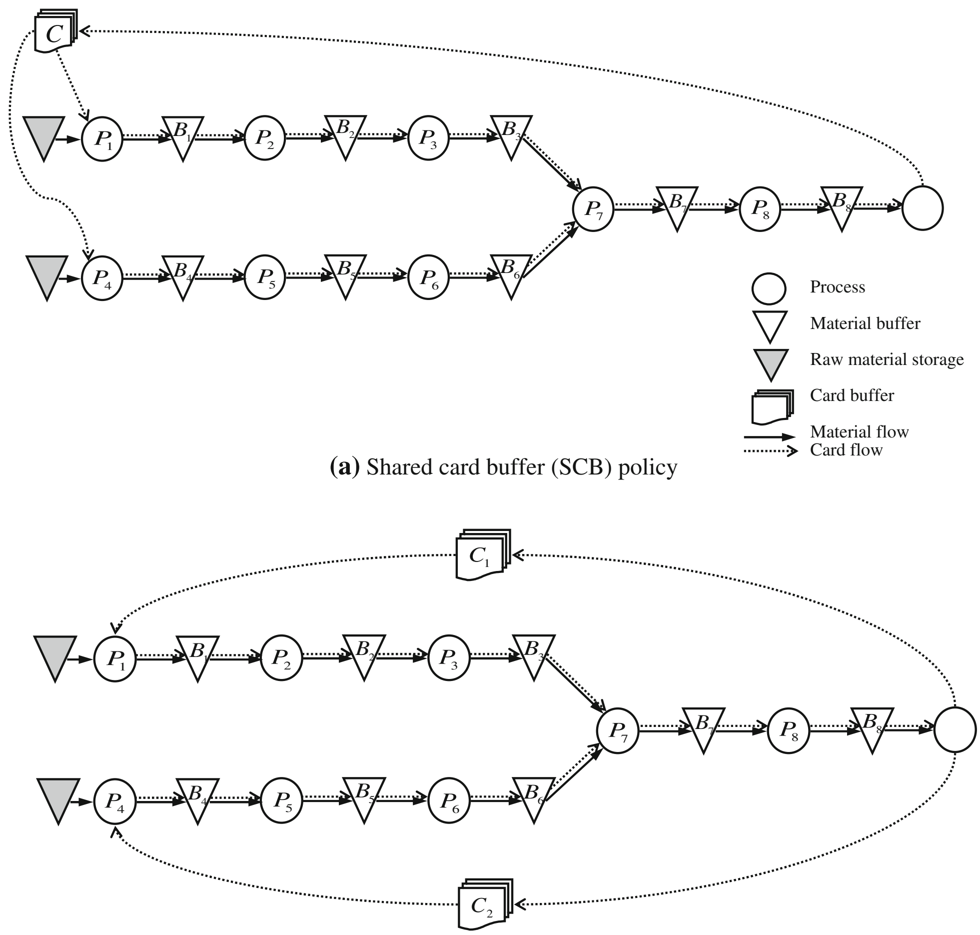
The CONWIP control is a very simple control mechanism that depends only on one parameter for the entire system: the total number of circulating cards, $c$. It influences both the transfer of finished parts downstream and the transfer of demands upstream through the system. There is no demand transfer between each workstation except the last and the first workstations. The production capacity or the maximum production rate of the system is affected only by the amount of $c$. The total amount of parts in the system is bound by $c$ and can be expressed as follows (see Fig. 3a).

$$Q(C) + \sum Q(P_i) + \sum Q(B_i) = c, \quad i = 1, \ldots, N$$

where $C$ indicates the CONWIP queue containing free cards, $P_i$ the manufacturing/assembly process of station $i$, $B_i$ the output buffer of station $i$, and $c$ the total number of cards in the system. If a workstation fails in a CONWIP line, the amount of material downstream of it will be gradually flushed out of the system by the demand process. These demand events will trigger the release of new raw parts into the system. When all CONWIP cards accumulate in front of the failed machine, the release of new jobs to the system will then stop.

In an assembly production system governed by the CONWIP mechanism, the fabrication lines begin a new job

**Fig. 3** CONWIP controlled assembly system




**(a)** Shared card buffer (SCB) policy

**(b)** Discrete card buffer (DCB) policy

whenever a part is completed at the end of the assembly line. The way in which cards from the last station are released in the first station of the fabrication lines depends on the specific rule adapted by the system. In the following section, we define two types of card releasing rules for an assembly production system governed by CONWIP.

### *Two policies for releasing cards*

In this section, we propose two policies to release cards in a CONWIP controlled assembly system. Figure 3 illustrates the two policies for the CONWIP system in an assembly line fed by two fabrication lines. In the first policy, called shared-card-buffer (SCB), there is only one card buffer shared for all fabrication lines, so that available cards might be attached to new jobs in either one of the fabrication lines, according to the first-come, first-serve discipline (Fig. 3a). The second policy, called discrete-card-buffer (DCB) is to set a separated card buffer for each fabrication line (Fig. 3b). According to this policy, a job is released in a fabrication line when there is an available card in the corresponding card buffer. When a finished product leaves the final buffer $B_8$, the cards are detached and then with the DCB policy, one card is added into each of $C_1$ and $C_2$, while with the SCB policy, two cards are added into the card buffer $C$. To the best of our knowledge, this is the first study that proposes these two policies to release cards in a CONWIP controlled assembly system, and provides a performance comparison between them.

The main difference between these two schemes is that they can have a different level of WIP with the same level of throughput. In other words, the average WIP can be less with DCB than with SCB CONWIP, providing the same total number of cards. The simulation results of comparison between these two policies are given in the next section.

## Analysis of SCB and DCB CONWIP policies

In this section, we provide performance comparisons between the two proposed CONWIP policies discussed in the previous section. Performance measures used in the comparisons are presented before simulation results.

### Performance measures

The following three performance measures are used in the simulation.

1. Throughput rate, $TH$. This is measured by the average number of products produced per time unit during the time length of the simulation.

$$TH = \frac{1}{T}\sum_{i=1}^{T} P_i,$$

where $P_i$ is the number of products produced at time $i$, and $T$ is the time length of the simulation.
2. Average total WIP, $WIP_{AT}$. This is measured by the average number of parts in the whole system during the time length of the simulation. This includes the products being processed on the machines and stored in the buffers.

$$WIP_{AT} = \frac{1}{T}\sum_{i=1}^{T} WIP_i,$$

where $WIP_i$ is the total WIP at the end of time $i$, and $T$ is the time length of the simulation.
3. Average WIP, $WIP_A$. This is measured by the average total WIP at each workstation.

$$WIP_A = \frac{WIP_{AT}}{N},$$

where, $N$ is the total number of workstations in the system.

## Numerical experiments

We consider an assembly system with three stages as depicted in Fig. 4d. The third stage is an assembly line which consists of two workstations fed by two sub-assembly lines in stage 2, each of which is fed by two fabrication lines in stage 1. Each line in stages 1 and 2 has three workstations. There are in total 20 workstations in the system. We ran each simulation program of the SCB and DCB for 30 cases. Processing times of operations are i.i.d. random variables generated from an exponential distribution with a mean of 10 min. No inventory was initially set at each output buffer, and every process except the first stations is assumed to be idle. We also assume that there are always enough raw materials at the first stations, and finished products leave the final buffer immediately, i.e. there is an unlimited demand for finished products. Each simulation was run for 40,000 min including a warm-up phase of 1,000 min.

Two performance measures, $WIP_A$ and $TH$, under each policy were calculated. The results are given in Table 1. In this table, the second column represents the minimum number of cards in each case assigned into the system with the DCB policy to attain the maximum possible throughput. In fact, assigning more cards into the system increases the average WIP, while the throughput of the system remains constant. In each individual case, the number of cards was found by a trial and error approach. This number was then assigned to the system with SCB policy. In case 1, for example, with DCB policy at least 24 cards were necessary to achieve the maximum possible throughput 2.73 [parts/hour]. While this number of cards with SCB policy attains only 1.56 [parts/hour] as the system throughput, and 0.85 [parts] as the average WIP.

The results show that the DCB policy provides less average WIP than SCB in most cases (in only one case, case 10, they had the same performance). It shows that for the examined 30 cases, the DCB CONWIP system gives on average 15.2% less WIP than the SCB CONWIP system, with a higher level of throughput. In addition, the throughput rate obtained

**Table 1** Performance of the DCB and SCB CONWIP policies

| Case | Number of cards | DCB | | SCB | |
|---|---|---|---|---|---|
| | | TH | $WIP_A$ | TH | $WIP_A$ |
| 1 | 24 | 2.73 | 0.74 | 1.56 | 0.85 |
| 2 | 20 | 2.50 | 0.69 | 1.36 | 0.79 |
| 3 | 24 | 2.73 | 0.74 | 1.54 | 0.86 |
| 4 | 24 | 2.31 | 0.71 | 1.16 | 0.91 |
| 5 | 20 | 2.07 | 0.63 | 0.89 | 0.84 |
| 6 | 16 | 2.14 | 0.61 | 0.97 | 0.74 |
| 7 | 20 | 2.61 | 0.66 | 1.45 | 0.76 |
| 8 | 18 | 2.22 | 0.67 | 1.00 | 0.73 |
| 9 | 20 | 2.07 | 0.65 | 0.84 | 0.85 |
| 10 | 19 | 2.07 | 0.65 | 2.07 | 0.65 |
| 11 | 18 | 2.86 | 0.55 | 2.01 | 0.57 |
| 12 | 20 | 2.40 | 0.73 | 1.23 | 0.93 |
| 13 | 14 | 2.61 | 0.43 | 1.39 | 0.51 |
| 14 | 18 | 2.86 | 0.61 | 1.95 | 0.67 |
| 15 | 14 | 2.40 | 0.44 | 1.19 | 0.49 |
| 16 | 12 | 2.00 | 0.47 | 0.79 | 0.49 |
| 17 | 16 | 2.31 | 0.56 | 1.11 | 0.69 |
| 18 | 20 | 2.14 | 0.75 | 0.94 | 0.88 |
| 19 | 20 | 2.50 | 0.64 | 1.33 | 0.93 |
| 20 | 16 | 2.00 | 0.55 | 0.83 | 0.74 |
| 21 | 19 | 2.22 | 0.62 | 1.03 | 0.75 |
| 22 | 21 | 2.50 | 0.67 | 1.33 | 0.80 |
| 23 | 16 | 2.40 | 0.53 | 1.19 | 0.64 |
| 24 | 18 | 2.31 | 0.60 | 1.09 | 0.64 |
| 25 | 20 | 2.61 | 0.70 | 1.40 | 0.82 |
| 26 | 14 | 2.14 | 0.44 | 0.98 | 0.51 |
| 27 | 18 | 2.22 | 0.58 | 1.02 | 0.71 |
| 28 | 22 | 2.73 | 0.71 | 1.46 | 0.87 |
| 29 | 17 | 2.07 | 0.62 | 0.87 | 0.69 |
| 30 | 23 | 2.86 | 0.73 | 1.57 | 0.92 |

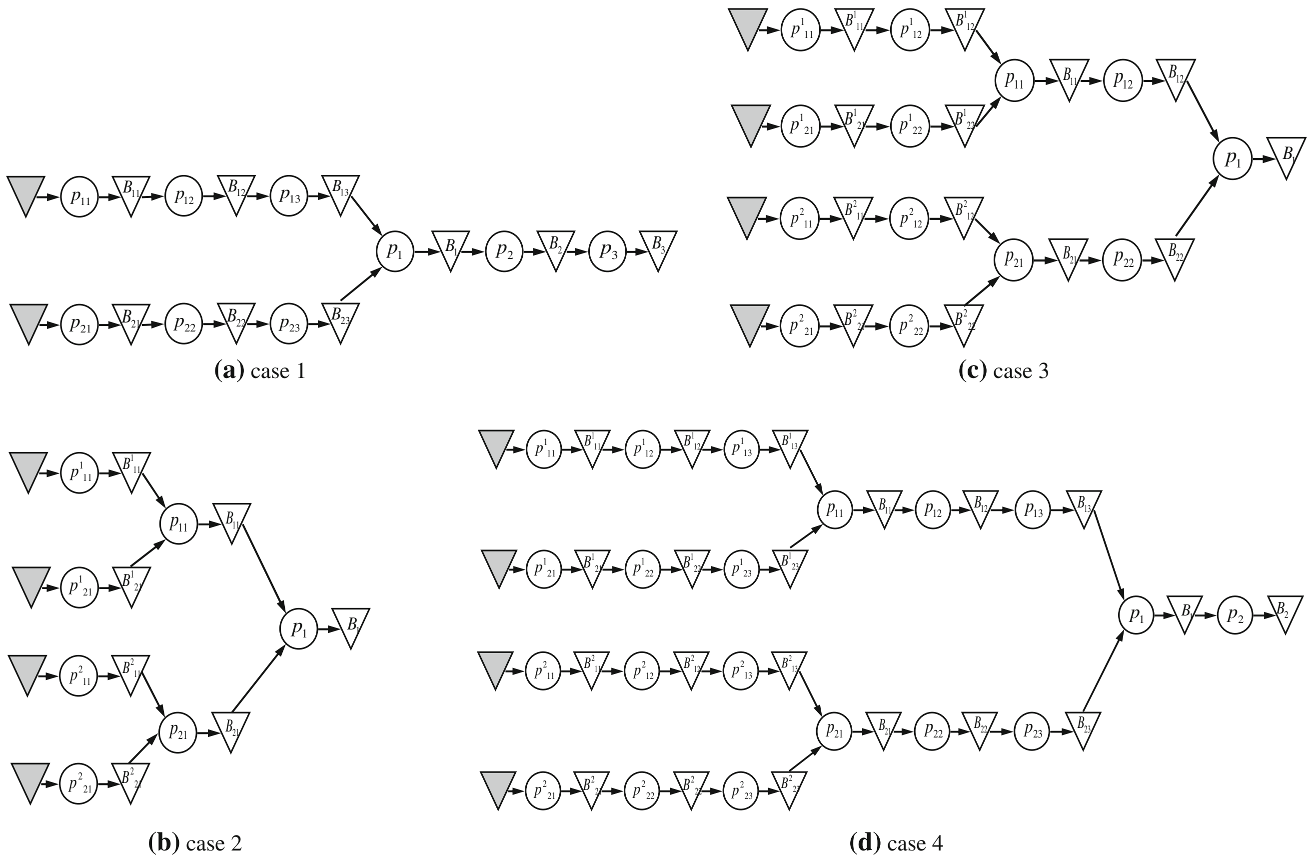


**Fig. 4** Four cases of simulation models

by the DCB policy is higher than that of SCB, given the same number of cards. As a consequence, DCB is superior to SCB CONWIP with a lower amount of average WIP and a higher rate of throughput, providing the same number of circulating cards within the system.

## Comparison of Kanban and CONWIP

In this section, we provide performance comparisons between Kanban and CONWIP controlled assembly systems with respect to the average WIP. We employ the DCB policy for the CONWIP system. Also, the effect of card distribution in Kanban systems is discussed.

### Simulation model

We constructed four cases as simulation models. Specific parameters are given in Table 2. The only difference among them is the number of lines at each stage, and hence, the total number of workstations in each case, which are 9, 7, 13 and 20 for cases 1, 2, 3 and 4, respectively. Case 1 is an assembly line fed by two fabrication lines, with three workstations in each line. For the other cases, we considered that an assembly system consists of three stages, where there are two sub-assemblies in the second stage and four fabrication lines in the first stage. However, the number of workstations at each stage is as follows.

Case 2: only one station in each line of each stage.
Case 3: two stations in each line of stages 1 and 2, with only one assembly station at the final stage.
Case 4: three stations in each line of stages 1 and 2, with two stations at the final stage.

Processing times of operations are generated from an exponential distribution with a mean of 10 min. As initial condition, every process except the first stations is assumed to be idle, and no inventory was set at each output buffer. We assume that there are always enough raw materials at the first stations, and finished products leave the final buffer immediately, i.e. there is an unlimited demand for finished products. Each simulation was run for 40,000 min with a 1,000 min warm-up time. The results are shown in Tables 3–6.

### Simulation results

Tables 3–6 show the simulation results for cases 1, 2, 3 and 4, respectively. Each table gives the *optimal* performance

**Table 2** Structure of cases 2, 3 and 4

| | Case 2 (Fig. 4b) | Case 3 (Fig. 4c) | Case 4 (Fig. 4d) |
|---|---|---|---|
| Number of sub-assembly lines, $M$ | 2 | 2 | 2 |
| Total number of fabrication lines, $\sum n_i \ (i = 1, \ldots, M)$ | 4 | 4 | 4 |
| Number of stations at the assembly line, $n$ | 1 | 1 | 2 |
| Number of stations at each sub-assembly line, $m_i \ (i = 1, \ldots, M)$ | 1 | 2 | 3 |
| Number of stations at each fabrication line, $n_{ij} \begin{pmatrix} i = 1, \ldots, M \\ j = 1, \ldots, n_i \end{pmatrix}$ | 1 | 2 | 3 |
| Total number of workstations, $\sum_{i=1}^{M} \sum_{j=1}^{n_i} n_{ij} + \sum_{i=1}^{M} m_i + n$ | 7 | 13 | 20 |

The four cases are illustrated in Fig. 4

**Table 3** Performance of the Kanban and CONWIP in case 1

| Sub-case | TH | $WIP_A$ | |
|---|---|---|---|
| | | Kanban | CONWIP |
| 1 | 3.54 | 0.92 | 1.18 |
| 2 | 3.16 | 1.05 | 1.35 |
| 3 | 3.16 | 1.05 | 1.42 |
| 4 | 3.34 | 0.94 | 1.15 |
| 5 | 3.00 | 1.17 | 1.29 |
| 6 | 3.34 | 0.96 | 1.22 |
| 7 | 3.01 | 0.82 | 1.29 |
| 8 | 3.53 | 0.93 | 1.24 |
| 9 | 3.01 | 0.99 | 1.29 |
| 10 | 3.16 | 1.22 | 1.19 |

**Table 4** Performance of the Kanban and CONWIP in case 2

| Sub-case | TH | $WIP_A$ | |
|---|---|---|---|
| | | Kanban | CONWIP |
| 1 | 4.00 | 0.95 | 0.95 |
| 2 | 4.00 | 0.96 | 0.97 |
| 3 | 3.33 | 0.88 | 1.22 |
| 4 | 4.00 | 1.00 | 0.97 |
| 5 | 5.46 | 0.96 | 1.08 |
| 6 | 4.29 | 0.87 | 0.88 |
| 7 | 4.62 | 0.93 | 0.95 |
| 8 | 3.53 | 0.85 | 1.15 |
| 9 | 3.00 | 0.80 | 1.16 |
| 10 | 3.00 | 0.94 | 1.04 |

**Table 5** Performance of the Kanban and CONWIP in case 3

| Sub-case | TH | $WIP_A$ | |
|---|---|---|---|
| | | Kanban | CONWIP |
| 1 | 3.16 | 0.80 | 0.90 |
| 2 | 3.53 | 0.98 | 1.01 |
| 3 | 3.34 | 0.87 | 0.93 |
| 4 | 3.34 | 0.84 | 1.12 |
| 5 | 3.34 | 0.89 | 0.96 |
| 6 | 3.75 | 0.93 | 1.01 |
| 7 | 3.16 | 0.77 | 0.72 |
| 8 | 3.63 | 0.91 | 1.07 |
| 9 | 3.00 | 0.87 | 0.95 |
| 10 | 3.75 | 0.95 | 0.98 |

**Table 6** Performance of the Kanban and CONWIP in case 4

| Sub-case | TH | $WIP_A$ | |
|---|---|---|---|
| | | Kanban | CONWIP |
| 1 | 2.07 | 0.97 | 0.99 |
| 2 | 2.07 | 0.85 | 1.09 |
| 3 | 2.07 | 0.95 | 0.94 |
| 4 | 3.20 | 0.97 | 1.00 |
| 5 | 2.40 | 0.84 | 0.91 |
| 6 | 3.53 | 0.85 | 1.11 |
| 7 | 2.40 | 1.00 | 1.00 |
| 8 | 3.34 | 0.85 | 1.00 |
| 9 | 3.53 | 0.86 | 1.08 |
| 10 | 3.00 | 0.94 | 1.06 |

measures, $TH$ [parts/hour] and $WIP_A$ [parts] of both Kanban and CONWIP at ten different sub-cases. Optimality here refers to the fact that the maximum possible throughput is achieved by assigning the minimum number of cards into the system. Thus, $WIP_A$ values given in the tables are the minimum number of WIP required to obtain the maximum possible throughput. The throughput of both Kanban and CONWIP for each sub-case is identical, because both systems have the same distribution of processing times and the same initial conditions.

From the four tables it can be inferred that the Kanban system provides better performance than CONWIP with a lower

$WIP_A$. In a precise description, the average WIP levels in the Kanban system are on average 20.14%, 10.69%, 8.02% and 10.38% less than those in the CONWIP system in cases 1, 2, 3 and 4, respectively. It shows that for an assembly system, Kanban can have a better performance than the CONWIP control system. However, only in rare cases, the CONWIP system has less value of $WIP_A$. For instance, in case 1, $WIP_A$ in the last sub-case is less in CONWIP than in Kanban. However, the difference between those two values is about 2.46%, which is too slight in analogy with the other sub-cases. Also, this rare result happened in three other cases, only once for each case. For those cases the differences are also slight, equivalent to 3% (sub-case 4), 6.49% (sub-case 7), and 1.05% (sub-case 3), in cases 2, 3 and 4, respectively. Therefore, the Kanban system provides a lower WIP level on average than the CONWIP system given the same level of throughput. As a consequence, the Kanban control system outperforms CONWIP on average, with a lower level of the average WIP and the same rate of throughput.

Meanwhile, in a Kanban system, we cannot neglect the effect of card distribution on the system performance. On the other hand, card distribution in a Kanban system as well as the number of circulating cards in a CONWIP system certainly influences the system performance. The effect of card distribution in a Kanban system and circumstance of setting a proper distribution of kanbans and a suitable number of circulating cards in a CONWIP system are detailed in the next section.

### The effect of card distribution

The last four tables compare the average WIP of the Kanban and CONWIP systems. For each case and each sub-case, we first found the optimum card distribution for both systems, which achieves a maximum possible throughput for the system with the minimum value of WIP, and then calculated the performance measures. Since there is an unlimited demand at the final buffer, the maximum possible throughput, denoted by $TH_{\max}$, is equal to the bottleneck rate of the system.

To find the optimal number of circulating cards in the CONWIP system, we started from the initial state in which there is only one card assigned into the system, and no initial inventory at each workstation. If system throughput is still less than $TH_{\max}$, we add one card and run the simulation program again. Once $TH_{\max}$ is achieved, we calculate $WIP_A$ which is the minimum value to attain the maximum possible throughput. When the system has $TH_{\max}$, assigning additional cards causes an increase in WIP, while throughput remains constant.

In the Kanban system, we observed that the best performance (i.e. maximum throughput with minimum WIP) in some cases can be achieved by assigning only one card at each station. Therefore, for those cases, the minimum WIP was obtained with a vector of card distribution in which all elements are equal to one. That is $K = \{k_1, k_2, \ldots, k_N\} = \{1,1,\ldots,1\}$, where $k_i$ indicates the number of cards at station $i$, and $N$ indicates the total number of stations in the system. However, in other cases, such as sub-cases 2, 5, 7 and 10 in Table 3, we first found the optimal card distribution. Because by assigning only one card at each workstation, the expected values of both WIP and throughput were not achieved. Therefore, in the Kanban system, the performance measures were obtained with adapted card distributions as well. As a result, card distribution in a Kanban system can affect the system performance. i.e., a different WIP level can be achieved by a different card distribution. In addition, in order to achieve the best performance, a proper number of cards should be assigned to the system.

## Conclusions

In this paper, we analyzed assembly production processes governed by Kanban and CONWIP control systems. For CONWIP, we introduced two methods for releasing cards into the system. These two were defined as SCB and DCB. We showed that the policy used has significant effect on the performance data of CONWIP controlled processes. The comparison results showed that DCB is superior to SCB CONWIP with a lower amount of average WIP and a higher rate of throughput, given the same number of circulating cards. Therefore, for an assembly production process controlled by the CONWIP system, using DCB policy can result in a better performance than using SCB.

Card distribution in a Kanban system, and the number of circulating cards in a CONWIP system can affect the system performance such that, WIP might rise by increasing the number of cards. For a CONWIP system, the best performance can be achieved by assigning the optimal number of circulating cards within the system. For a Kanban system, in some cases by assigning only one card at each station, the minimum WIP can be achieved, while for most cases an adaptation of card distribution is necessary in order to get the best performance.

We also compared Kanban and CONWIP systems in assembly production processes. Comparison between Kanban and CONWIP showed that Kanban provides a less WIP level on average than CONWIP given the same level of throughput. In most cases, an optimized Kanban system (a system with adapted card distribution) outperforms the CONWIP system with a lower WIP and the same level of throughput. However, only in rare cases, CONWIP provides a lower WIP level, whereas the difference is too slight and insignificant.

The comparison between Kanban and CONWIP revealed that the Kanban system is more flexible with respect to a certain objective than CONWIP, because in addition to the total number of cards, the card distribution is another parameter that influences performance. By selecting a favorable card distribution in the Kanban system, a given throughput rate can be reached with less WIP than in the CONWIP system. Thus, the CONWIP system does not always outperform Kanban in the assembly production processes. However, in order to achieve the best performance with Kanban, a proper distribution of cards should be sought and assigned to the system.

The following issues shall be considered in future research.

1. All our results are based on unlimited demand at the end of the assembly line in the last stage. It has to be seen if they also hold for limited demand. The important performance parameter will then be the ratio between immediately satisfied demand and total demand (service level).
2. We assumed that the system makes a single part type. In future research, more complex production systems (for example, multi-product manufacturing or assembly systems) can be considered. The analysis will be more complex if different part types and limited demand are considered. In a Kanban system, cards (and containers) for each part type are held at each station. In a CONWIP system, the cards are not assigned to a defined part type, they rather represent a certain amount of work. It has to be examined how much WIP is held in a Kanban line in comparison to a CONWIP line if more than one part type is considered.
3. Kanban and CONWIP can be compared with other control systems, such as Hybrid (a Kanban-CONWIP control system), the Base-stock, and Extended-Kanban which is a combination of the Kanban and Base-stock control systems.
4. Some popular meta-heuristics such as genetic algorithm and Tabu search can be developed to find the optimal card distribution in the Kanban system.

**Acknowledgements** The author thanks his academic advisor, Professor Ryo Sato, for his advice and valuable comments. He is also grateful to anonymous referees for helpful suggestions that improved earlier version of the paper.

## References

Cao, D., & Chen, M. (2005). A mixed integer programming model for a two line CONWIP-based production and assembly system. *International Journal of Production Economics*, *95*, 317–326.

Duenyas, I. (1994). Estimating the throughput of acyclic assembly system. *International Journal of Production Research*, *32*, 1403–1419. doi:10.1080/00207549408957007.

Duenyas, I., & Hopp, W. J. (1992). CONWIP assembly with deterministic processing and random outages. *IIE Transactions*, *24*, 97–109. doi:10.1080/07408179208964237.

Duenyas, I., & Hopp, W. J. (1993). Estimating the throughput of an exponential CONWIP assembly system. *Queueing Systems*, *14*, 135–157. doi:10.1007/BF01153531.

Duri, C., Frein, Y., & Lee, H. S. (2000). Performance evaluation and design of a CONWIP system with inspections. *International Journal of Production Economics*, *64*, 219–229. doi:10.1016/S0925-5273(99)00060-2.

Framinan, J. M., Gonzalez, P. L., & Ruiz-Usano, R. (2003). The CONWIP production control system: Review and research issues. *Production Planning and Control*, *14*, 255–265. doi:10.1080/0953728031000102595.

Framinan, J. M., Ruiz-Usano, R., & Leisten, R. (2000). Input control and dispatching rules in a dynamic CONWIP flow-shop. *International Journal of Production Research*, *38*, 4589–4598. doi:10.1080/00207540050205523.

Gstettner, S., & Kuhn, H. (1996). Analysis of production control systems kanban and CONWIP. *International Journal of Production Research*, *34*(11), 3253–3274. doi:10.1080/00207549608905087.

Hazra, J., & Seidmann, A. (1996). Performance evaluation of closed tree-structured assembly systems. *IIE Transactions*, *28*, 591–599.

Hopp, W. J., & Roof, M. L. (1998). Setting WIP levels with statistical throughput control (STC) in CONWIP production lines. *International Journal of Production Research*, *36*, 867–882. doi:10.1080/002075498193435.

Hopp, W. J., & Spearman, M. L. (1991). Throughput of a constant work in process manufacturing line subject to failures. *International Journal of Production Research*, *29*, 635–655. doi:10.1080/00207549108930093.

Hopp, W. J., & Spearman, M. L. (2001). *Factory physics: The foundations of manufacturing management*. McGraw-Hill, New York.

Huang, M., Wang, D., & IP, W. H. (1998). Simulation study of CONWIP for a cold rolling plant. *International Journal of Production Economics*, *54*, 257–266. doi:10.1016/S0925-5273(97)00152-7.

Ip, W. H., Huang, M., Yung, K. L., Wang, D., & Wang, X. (2007). CONWIP based control of a lamp assembly production line. *Journal of Intelligent Manufacturing, 18*(2), 261–271. doi:10.1007/s10845-007-0021-0.

Monden, Y. (1998). *Toyota production system: An integrated approach to just-in-time*. Industrial Engineering & Management Press.

Muckstadt, J. A., & Tayur, S. R. (1995a). A comparison of alternative kanban control mechanisms: I, background and structural results. *IIE Transactions*, *27*(1), 140–150. doi:10.1080/07408179508936726.

Muckstadt, J. A., & Tayur, S. R. (1995b). A comparison of alternative kanban control mechanisms: II, experimental results. *IIE Transactions*, *27*(1) 151–161. doi:10.1080/07408179508936727.

Sato, R., & Khojasteh-Ghamari, Y. (2008). An integrated framework for card-based production control systems. *Journal of Intelligent Manufacturing*, submitted.

Spearman, M. L., & Zazanis, M. A. (1992). Push and pull production systems: Issues and comparisons. *Operations Research, 40*, 521–532.

Spearman, M. L., Woodruff, D. L., & Hopp, W. J. (1990). CONWIP: A pull alternative to Kanban. *International Journal of Production Research*, *23*, 879–894. doi:10.1080/00207549008942761.

Takahashi, K., Myreshka, & Hirotani, D. (2005). Comparing CONWIP, synchronized CONWIP, and Kanban in complex supply chains. *International Journal of Production Economics*, 93–94, 25–40. doi:10.1016/j.ijpe.2004.06.003.