RC 20158 (08/14/95)
Computer Science/Mathematics

IBM Research Report

Determination of Optimal Kanban Allocation Patterns in Serial Production Lines

Markus Ettl

IBM Research Division
T.J. Watson Research Center
Yorktown Heights, New York

Hansjörg Fromm

IBM German Manufacturing Technology Center
Sindelfingen, Germany

IBM Research Division
Almaden · T.J. Watson · Tokyo · Zurich

Abstract

Kanban systems are well-known for their ability to operate production lines smoothly and according to demand by keeping the work-in-process down on some prespecified level. The problem of determining the right number of kanbans to be assigned to individual production stages is still not well understood. This paper attempts to attain more insight into the behavior of kanban systems and to derive general design rules for the allocation of kanbans which can be used by the practitioner. The production systems under consideration are serial production lines with identical and non-identical machines. Kanban allocation patterns are compared with respect to maximum throughput and minimal work-in-process. In the case of non-identical machines, capacity and variability bottlenecks are considered. The results will be compared with results available for the allocation of buffers in finite-buffer production lines.

Keywords: Just-in-Time, CONWIP, Kanban Allocation, Stochastic Petri Nets, Performance Evaluation.

1. Introduction

Kanban systems are well-known for their ability to operate production lines smoothly and according to demand by keeping the work-in-process (WIP) down to some prespecified level. It is a common misunderstanding, however, that work-in-process inventories have to be as small as possible, affected by such slogans as *zero inventories* or *stockless production* (Hall 1983; Schonberger 1982). Even if there are prominent cases which show that production lines can be operated with extremely small in-process inventory (Monden 1983), it is a mistake to believe that this is true for manufacturing systems in general. Many such attempts have been made though, but they were determined to fail and left behind distrust in the practicability of the kanban principle. Such pitfalls could be avoided if manufacturing people knew more about the principles of kanban systems and their possible extensions, and if rules and guidelines were available for the design and configuration of such systems. This paper is an attempt to make one step in this direction.

In industrial installations, kanban systems often cause a loss of productivity due to their tight coupling of machines. This is particularly the case if the processing times are subject to variability. Variability is often not a property of the production process itself, but it may be introduced by random interruptions such as machine failures, maintenance periods or setup times. To account for the problem of variability when introducing kanban control, two principles can be considered for implementation:

- Decoupling of the line through the proper allocation of kanbans, e.g., assigning more kanbans to bottleneck processes.
- Segmentation of the line into groups of several machines (production stages) and assigning kanbans to production stages rather than to single machines.

Both principles are an attempt to preserve the advantages of a kanban system (e.g., limitation of WIP inventory, reduction of manufacturing cycle times) without loosing too much productivity.

In this paper, kanban systems are studied experimentally with the attempt to attain more insight into their behavior and to find general design rules which can be used by the practitioner. From a production management point of view, the paper is important in that companies wishing to properly implement the pull concept need some general design guidelines to determine where additional kanbans can be useful and where they would result in waste. Such design guidelines are likely to facilitate the design and configuration of kanban-controlled lines and should also enhance our understanding of their behavior.

The remainder of this paper is organized as follows. Section 2 reviews the literature related to kanban sytems modeling and discusses previous work that addresses kanban allocation issues. Section 3 describes the operation of the kanban system as considered in this paper. A stochastic Petri net representation of the system is presented in Section 4. Section 5 studies the kanban allocation issues under different production settings and briefly touches on the problem of configuring the production stages. Section 6 summarizes our findings and concludes the paper.

2. Literature Review

Previous research in the area of pull production systems modeling often uses simulation to study various performance and implementation aspects, see, e.g., Chu and Shih (1992) and the references therein. For performing quantitative analysis of pull production systems, several authors proposed the use of stochastic Petri nets (e.g., Jothishankar and Wang 1992; Wang *et al.* 1992). Di Mascolo *et al.* (1991) showed that stochastic Petri nets are well-suited to provide a framework for the modeling of various kanban systems proposed in the literature, including the model considered in the present study. Analytical approximation methods have been developed, among others, by So and Pinault (1988), Mitra and Mitrani (1990; 1991), Berkley (1992), Di Mascolo *et al.* (1992) and Di Mascolo (1993).

The majority of literature related to the allocation of buffers in production lines deals primarily with finite-buffer serial lines. Perhaps the best known research paper was published by Conway *et al.* (1988). Using a simulation approach, the authors demonstrated that in a balanced system the proper buffer placement strategy should provide more storage space towards the center machines if it is intended to maximize system throughput. Similar results have been reported by Hillier *et al.* (1993) in a recent study. Their experiments indicated that the optimal allocation pattern follows a *bowl phenomenon* whereby the center buffers are given preferential treatment over the end buffers. As a consequence, one of the basic questions investigated in this study is, does the bowl phenomenon also exist in serial kanban lines?

Relatively little attention has yet been devoted to the problem of kanban allocation in pull systems. Lambrecht and Segaert (1990) investigated the behavior of WIP inventory in serial and assembly-type of production lines, given that the objective is to maximize throughput. A key concept in their paper is the *long-pull* strategy which is similar to the kanban control policy considered here. Siha (1994) derived a continuous time Markov model to investigate the impact of different kanban allocation patterns for up to five-station kanban systems. The author found that a *funnel pattern* which allocates more kanbans towards the downstream stations improves system performance in a variety of cases. Chang and Yih (1994) proposed a simulated annealing algorithm to determine the number of kanbans and lot sizes in generic kanban systems. De Araujo *et al.* (1995) developed a computational procedure which determines (sub-)optimal kanban allocations based on an analytical approximation method for evaluating the performance of kanban systems.

Theoretical results on kanban systems, such as dominance of kanban allocations, were derived by Tayur (1992) using stochastic ordering. Based on these results, Tayur (1993) presented a heuristic for the allocation of kanbans to attain maximum system throughput. Glasserman and Yao (1995) studied the effect of varying buffer sizes in serial lines with general blocking which includes kanban blocking as a special case. Other approaches to kanban allocation that have been presented usually depend on specific modeling assumptions and thus are often of restricted practical value (e.g., Davis and Stubitz 1987; Bard and Golany 1991; Wang *et al.* 1992; Duenyas and Hopp 1993).

3. Description of the Model

The production system studied here is a tandem network of N machines which are distributed among P production stages. Each production stage is assumed to consist of a production process, and output buffer, and a kanban buffer (Figure 1). The production process consists of one or several machines in tandem, each equipped with an unlimited local buffer to store unfinished parts. The output buffer is a storage area which contains the finished parts of the production stage. The kanban buffer is a designated location in which unattached kanbans are collected. In order to enter production stage i, a part must acquire a kanban from the kanban buffer which is attached to the part as long as it is in that stage. When the processing of a part has been completed, the finished part is moved to an output buffer where it waits for admission to the downstream stage (stage $i + 1$). The kanban associated with a finished part is detached as soon as the part is withdrawn by the downstream stage. The unattached kanban is then returned without delay to the kanban buffer where it serves as a pull signal for the upstream stage (stage $i-1$).

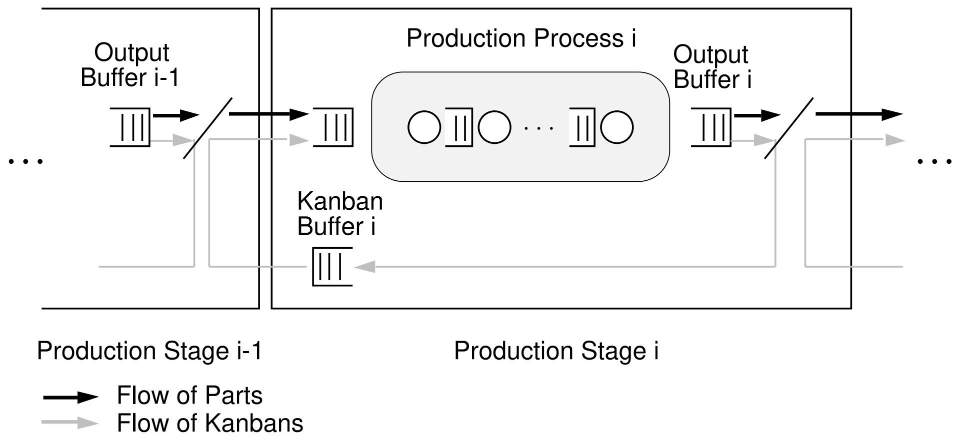

Figure 1: Illustration of Parts and Kanban Flow Between Different Production Stages.

The kanban system produces one type of parts. We assume that an unlimited supply of parts ready to be processed is available at the first stage. We further assume that an inexhaustible demand for finished products is present at the output of the kanban system, that is, we consider a saturated system. Notice that the flow of parts through the system is controlled through a combined push/pull control policy. A push control policy is used within a production stage, whereas parts are pulled between different production stages according to the rate at which parts are consumed by the downstream stages. To the authors' experience, such kanban systems are sufficiently general for practical purposes. Prototype implementations are under way at different manufacturing sites in Europe and in the U.S. (e.g., Tayur 1993).

It is important to see that two implementations of kanban systems discussed previously in the literature appear as special cases of the above model through certain arrangements of the production stages:

- *Conventional Kanban.* The conventional kanban system formally described by Mitra and Mitrani (1990) implements the basic pull principle. Every production stage consists of exactly one machine, one kanban buffer, and one output buffer.

- *CONWIP (Constant Work-in-Process).* A different pull-based production control scheme called CONWIP was described by Spearman *et al.* (1990). In a CONWIP system, all stations are combined to one production stage in which the work-in-process is held at a constant level. Pull-signals are sent from the last to the first machine in the line and parts are pushed between all others.

4. Stochastic Petri Net

To evaluate the throughput and average WIP inventories of the kanban system in an accurate way, we must find an appropriate representation of the underlying stochastic process. For that purpose, we shall use stochastic Petri nets. Figure 2 illustrates the basic Petri net model which depicts two successive production stages, each stage consisting of two machines in tandem. The transition located in between the two production stages has two input places and two output places. As soon as a token is present in the place representing the kanban buffer of stage i and the place representing the output buffer of stage $i-1$, the transition is instantaneously fired. The tokens are then being removed from the associated places and a token will be added to the place representing the input buffer of stage i and the place representing the kanban buffer of stage $i-1$, thus modeling the transportation of parts between stages. The number of tokens that circulate among the places associated with a single production stage is equal to the number of kanbans assigned to that stage. The construction of a multi-stage kanban system is entirely straightforward once the individual stages are combined in a linear fashion.

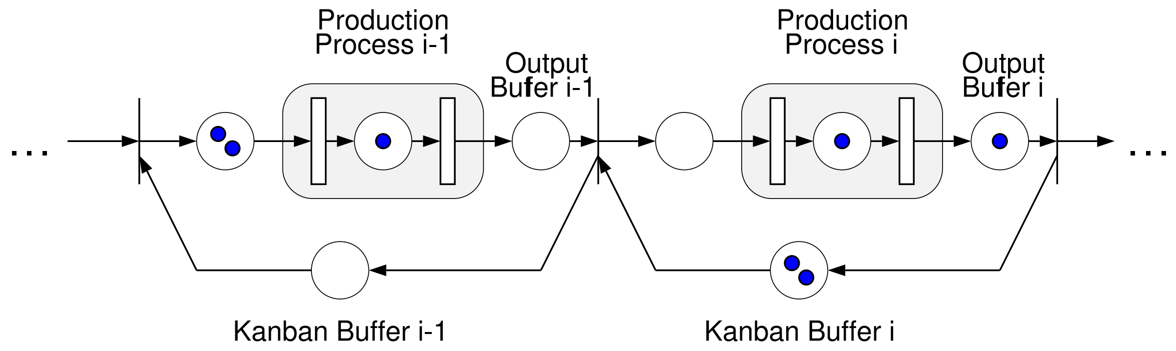

Figure 2: Illustration of the Stochastic Petri Net Model.

The processing times at the individual machines are represented by timed transitions with a probabilistic delay (e.g., deterministic or exponentially distributed). For the purpose of our investigations, we shall assume that the processing times are exponentially distributed. It has been shown elsewhere that the exponential distribution is often a reasonable approximation of real-world production, in particular when equipment failures are taken into account (e.g., Solberg and Nof 1980). We conducted a number of additional experiments with non-exponential processing time distributions and it turned out that the conclusions drawn in this paper are valid for other distributions as well. For efficient model construction and evaluation, we employed the programming package SPNP (Ciardo and Muppala 1991) which uses a combination of successive over-relaxation and the Gauss-Seidel method to solve the underlying continuous-time Markov chain.

5. Experimental Results

We now present the results of an experimental study which we have conducted to explore the kanban allocation problem. The performance of the kanban system is measured in terms of the achieved throughput, given that the number of kanbans for allocation is predetermined. The optimal kanban allocation was determined through a simple neighborhood search procedure where the starting allocation was chosen to be a reasonably uniform one. All performance measures of interest have been obtained from the steady-state distribution of reachable markings that is supplied by the SPNP analysis package. Unless stated otherwise, we assume that all processing times are independent and exponentially distributed with mean equal to one.

Subsections 5.1 to 5.5 discuss the kanban allocation problem in conventional kanban systems with identical and non-identical machines. The more general system in which several machines are assigned to individual production stages is considered in subsection 5.6.

5.1 How to Optimally Allocate One Additional Kanban

In the first experiment, suppose that each machine is equipped with one kanban and one extra kanban is available for allocation. Where should it be placed in order to achieve the maximum throughput? Table 1 reports the results for up to seven identical machines. It is apparent that the highest throughput is achieved if the extra kanban is allocated towards the center of

4 Machine Line			*5 Machine Line*		
Kanban Allocation	Throughput	WIP	Kanban Allocation	Throughput	WIP
2-1-1-1	.5446	4.24	2-1-1-1-1	.5054	4.97
1-2-1-1	.5758	4.08	1-2-1-1-1	.5277	4.91
1-1-2-1	.5758	3.58	1-1-2-1-1	.5380	4.54
1-1-1-2	.5446	3.23	1-1-1-2-1	.5277	4.09
			1-1-1-1-2	.5054	3.86
6 Machine Line			*7 Machine Line*		
Kanban Allocation	Throughput	WIP	Kanban Allocation	Throughput	WIP
2-1-1-1-1-1	.4805	5.70	2-1-1-1-1-1-1	.4633	6.43
1-2-1-1-1-1	.4965	5.70	1-2-1-1-1-1-1	.4752	6.46
1-1-2-1-1-1	.5078	5.45	1-1-2-1-1-1-1	.4851	6.30
1-1-1-2-1-1	.5078	5.04	1-1-1-2-1-1-1	.4889	5.97
1-1-1-1-2-1	.4965	4.67	1-1-1-1-2-1-1	.4851	5.59
1-1-1-1-1-2	.4805	4.53	1-1-1-1-1-2-1	.4752	5.31
			1-1-1-1-1-1-2	.4633	5.22

Table 1: Allocation of One Additional Kanban in Conventional Kanban Systems.

the line. An intuitive explanation is that an extra center kanban breaks the production line into two nearly identical subsystems such that both subsystems can operate more independent of each other. Moreover, we observe that mirror image kanban allocations produce the same throughput whereby the amount of WIP inventory is smaller when the extra kanban is allocated downstream. For example, in a four-machine line, the two candidate allocations 2-1-1-1 and 1-1-1-2 yield the same throughput but the second allocation results in 24 % less WIP. The same effect has been observed by Conway *et al.* (1988) in production lines with finite buffers.

5.2 How to Characterize the Optimal Allocation of Kanbans

In this experiment, we consider conventional kanban systems of identical machines when the number of kanbans is increased. The results are given in Table 2 for lines consisting of 5 and 6 machines. In cases where two mirror-image optimal allocations exist, we always selected the one with less WIP inventory, that is, the one with the larger number of kanbans allocated to the downstream half of the line.

	5 Machine Line			6 Machine Line		
Kanbans	Optimal Allocation	Throughput	WIP	Optimal Allocation	Throughput	WIP
10	1-3-2-3-1	.6788	7.68	1-2-2-2-2-1	.6158	7.68
11	1-3-3-3-1	.7002	8.46	1-2-2-3-2-1	.6354	8.23
12	1-4-2-4-1	.7181	9.12	1-3-2-2-3-1	.6600	9.16
13	1-4-3-4-1	.7366	9.96	1-3-2-3-3-1	.6794	9.77
14	1-4-4-4-1	.7501	10.71	1-3-3-3-3-1	.6965	10.76
15	1-5-3-5-1	.7641	11.41	1-3-3-3-4-1	.7100	10.90
16	1-5-4-5-1	.7761	12.20	1-4-3-3-4-1	.7264	12.22
18	1-6-4-6-1	.7967	13.66	1-5-3-3-5-1	.7487	13.68
20	1-7-4-7-1	.8133	15.09	1-5-4-4-5-1	.7701	15.29
30	1-11-6-11-1	.8681	22.39	1-9-5-5-9-1	.8311	22.65

Table 2: Optimal Allocation of Kanbans in Conventional Kanban Systems.

A key observation is that the optimal allocation pattern gives some preference to machines located directly upstream and downstream of the center, rather than the center machines themselves. The imbalance of the kanban allocation can thus become quite severe, in particular for the higher ranges of kanban capacity. Furthermore, we observe that it is always sufficient to allocate one kanban to the end stations, irrespective of the total number of kanbans available. The dominance theorem by Tayur (1992) gives mathematical evidence that in conventional kanban systems this holds true in general. The results of Table 2 can be summarized as follows:

- It is most effective to allocate additional kanbans near the center, where the center machine(s) should be given somewhat lower priority. The first and last machines should be operated with only one kanban.
- The optimal allocation pattern tends to be symmetrical.
- Mirror image kanban allocations result in the same throughput. To reduce the average work-in-process inventories, it is advisable to choose the allocation pattern with a larger amount of kanbans allocated to the downstream half of the line.

As a consequence of the first observation, we conclude that the bowl phenomenon does not hold in conventional kanban system.

5.3 Impact of the Kanban Allocation Pattern on System Throughput

In this experiment, we study the impact of different kanban allocation patterns on the throughput of a conventional kanban system. In all examples considered, we assume identical machines. Suppose the total kanban capacity is three times the number of machines. Which kanban allocation pattern is closest to optimal with respect to system throughput?

	4 Machine Line			*5 Machine Line*		
	Kanban Allocation	Throughput	WIP	Kanban Allocation	Throughput	WIP
optimal	1-5-5-1	.7847	9.13	1-5-3-5-1	.7641	11.41
uniform	3-3-3-3	.7696	9.56	3-3-3-3-3	.7503	11.80
bowl	2-4-4-2	.7805	9.53	2-3-5-3-2	.7564	11.72
funnel	2-3-3-4	.7641	8.41	1-2-3-4-5	.6998	7.67
reversed funnel	4-3-3-2	.7641	10.46	5-4-3-2-1	.6998	14.20
	6 Machine Line			*7 Machine Line*		
	Kanban Allocation	Throughput	WIP	Kanban Allocation	Throughput	WIP
optimal	1-5-3-3-5-1	.7487	13.68	1-4-3-5-3-4-1	.7366	16.02
uniform	3-3-3-3-3-3	.7370	14.05	3-3-3-3-3-3-3	.7272	16.31
bowl	2-3-4-4-3-2	.7446	14.07	2-3-3-5-3-3-2	.7325	16.30
funnel	2-2-3-3-4-4	.7178	11.36	2-2-3-3-3-4-4	.7098	13.33
reversed funnel	4-4-3-3-2-2	.7178	15.99	4-4-3-3-3-2-2	.7098	18.52

Table 3: Kanban Allocation Patterns in Conventional Kanban Systems.

The results of Table 3 demonstrate that the bowl pattern performs rather well, even though it differs from the optimal allocation. It is also apparent that the uniform strategy which allocates kanbans equally over the line behaves relatively poorly with respect to throughput, and it tends to accumulate a higher level of WIP in comparison to the bowl and optimal pattern.

An allocation pattern of high interest is the *funnel* pattern in which more kanbans are allocated towards the downstream machines. The impact of the funnel allocation was recently investigated by Siha (1994) in a conventional kanban system where production is triggered by demands for finished products arriving randomly in time (i.e., a non-saturated kanban system). The author showed empirically that the funnel pattern may improve system performance, especially if the line is operated in moderate to high throughput ranges. The questions is, does the funnel strategy perform equally well in the saturated model where demands for finished products are always available at the output of the final production stage? For that purpose, we present Figure 3 which displays the throughput versus WIP tradeoff curves for the uniform, bowl and the funnel allocation patterns in a five-machine conventional kanban system. It is readily observed that with the funnel pattern, a given system throughput can be achieved at the lowest level of WIP inventory. The difference in the amount of WIP inventory is well above 20 % in the present example and it is expected to increase further when the throughput steps up.

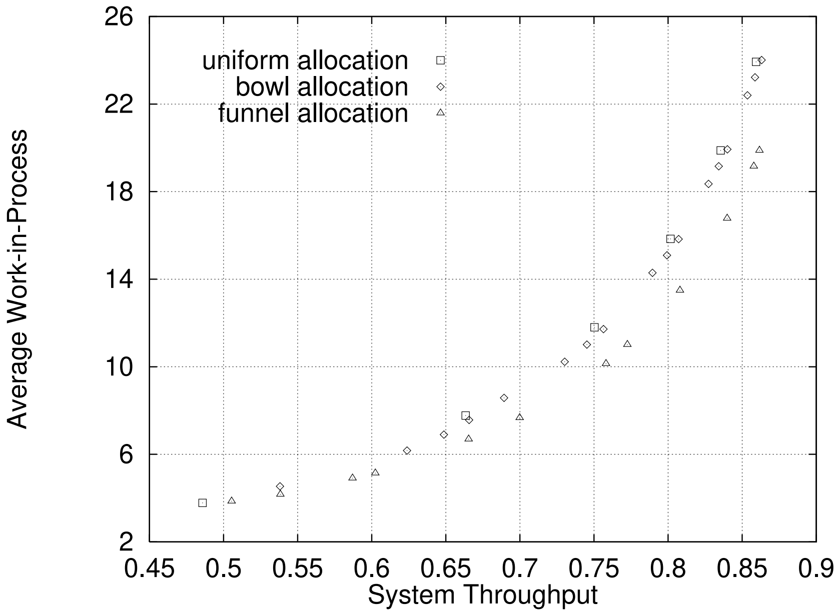

Figure 3: Throughput Versus WIP Tradeoff in a Five-Machine Conventional Kanban System.

This observation can be explained when we trace the distribution of WIP over the individual machines. Table 4 gives the average WIP content per machine, w_i, $1 \leq i \leq 5$, for different uniform and funnel allocation patterns. It can be observed that the funnel pattern results in a balanced distribution of WIP over the entire line, with a slight increase towards the downstream machines. As a result, the release of finished products can take place rather smoothly. With the uniform allocation, in contrast, WIP accumulates in the upstream buffers of the line which is a relatively ineffective location if it is intended to improve system throughput.

Kanban Allocation		Throughput	WIP	w_1	w_2	w_3	w_4	w_5
funnel	1-2-3-4-5	.6998	7.67	1.00	1.39	1.65	1.81	1.83
funnel	1-3-5-7-9	.7725	11.01	1.00	1.94	2.45	2.76	2.86
funnel	1-6-10-15-20	.8617	19.88	1.00	3.56	4.68	5.24	5.51
uniform	2-2-2-2-2	.6634	7.77	2.00	1.75	1.57	1.37	1.07
uniform	4-4-4-4-4	.8016	15.84	4.00	3.53	3.19	2.83	2.28
uniform	6-6-6-6-6	.8594	23.93	6.00	5.31	4.82	4.30	3.51

Table 4: Distribution of WIP in a Five-Machine Conventional Kanban System.

5.4 Impact of Processing Time Imbalance

To explore the effect of imbalance in the line, we consider the case that one of the machines is characterized by a larger mean processing time, that is, a *capacity bottleneck*. As the system throughput is determined by the bottleneck processing rate, we would expect the optimal kanban allocation to protect the bottleneck from starvation and blocking. What is the best way to achieve this goal? For example, is it better to provide more kanbans to machines located directly upstream and downstream of the bottleneck, or should the bottleneck itself be preferred? Table 5 gives evidence that the latter alternative prevails. The more severe the bottleneck becomes, the more unbalanced is the associated allocation pattern.

	Bottleneck Processing Time 1.2			*Bottleneck Processing Time 2.0*		
Bottleneck	Optimal Allocation	Throughput	WIP	Optimal Allocation	Throughput	WIP
1st	1-6-3-4-1	.7258	10.50	1-8-3-2-1	.4984	5.16
2nd	1-6-3-4-1	.7253	11.68	1-8-3-2-1	.4983	11.39
3rd	1-4-5-4-1	.7245	11.66	1-3-7-3-1	.4981	12.17
4th	1-4-3-6-1	.7253	11.65	1-2-3-8-1	.4983	13.58
5th	1-4-3-6-1	.7258	11.72	1-2-3-8-1	.4984	13.99

Table 5: Optimal Kanban Allocation in Conventional Kanban System with Capacity Bottleneck.

It can further be observed that if the bottleneck is situated at one of the end positions of the line, it does *not* pull kanban capacity towards itself. In such a case, it is better to operate the machine located immediately downstream (resp. upstream) with an increased number of kanbans. An intuitive explanation is that, due to the kanban principle, a kanban allocated to the first station, say, can buffer between the first and the second station whereas a kanban at the second station can buffer between the first and second or between the second and third station. In other words, an optimal allocation strategy will always give one kanban to the first and last machine, irrespective of the distribution of processing time.

5.5 Impact of Processing Time Variability

In this experiment, we study the impact of a *variability bottleneck* on the performance of conventional kanban systems. Suppose that all processing time distributions have a common mean, but there is one process step with a larger variability than the others, e.g., due to equipment failures, unscheduled maintenance periods, or other random events. To investigate the issue of kanban allocation in the presence of variability, a line of five machines is considered. For simplicity, we assume that the processing times of the non-bottleneck machines are exponentially distributed, that is, the coefficients of variation are equal to one. The coefficient of variation of the bottleneck, CV, is varied between 2 and 10. The top three kanban allocations were determined for different locations of the bottleneck. Table 6 illustrates the results.

	Position of Variability Bottleneck					
	2nd Position		3rd Position		4th Position	
CV	Allocation	Throughput	Allocation	Throughput	Allocation	Throughput
2.0	1-5-3-5-1	.7282	1-5-3-5-1	.7278	1-5-3-5-1	.7282
	1-4-4-5-1	.7274	1-4-4-5-1	.7258	1-5-4-4-1	.7274
	1-5-4-4-1	.7267	1-5-4-4-1	.7258	1-4-4-5-1	.7267
5.0	1-4-4-5-1	.6430	1-5-3-5-1	.6418	1-5-4-4-1	.6430
	1-4-3-6-1	.6416	1-5-2-6-1	.6412	1-6-3-4-1	.6416
	1-5-3-5-1	.6415	1-6-2-5-1	.6412	1-5-3-5-1	.6415
10.0	1-3-5-5-1	.5298	1-6-1-6-1	.5291	1-5-5-3-1	.5298
	1-3-4-6-1	.5297	1-5-2-6-1	.5284	1-6-4-3-1	.5297
	1-4-4-5-1	.5295	1-6-2-5-1	.5284	1-5-4-4-1	.5295

Table 6: Top 3 Kanban Allocations in Conventional Kanban System with Variability Bottleneck.

Conway *et al.* (1988) found that in a finite-buffer production line, machines with highly variable processing times should have large upstream and downstream buffers. A similar effect can be perceived in kanban systems (as an example, allocation 1-6-1-6-1 is optimal if the bottleneck is located in the middle and the CV is equal to 10). Another general observation is that in a symmetrical kanban system, the optimal allocation is also a symmetrical one. This result is also valid in finite-buffer systems. What is remarkable, however, is that the allocation patterns observed here are fundamentally different from those established in the previous section: in a capacity bottleneck line, the system throughput is maximized when kanbans are allocated towards the bottleneck itself, whereas in a variability bottleneck line, it is better to provide more kanbans to machines located directly upstream and downstream of the bottleneck.

5.6 Impact of the Network Partition

We now proceed with an example in which the production stages contain more than one machine. In addition to the kanban allocation problem, we thus have to deal with the problem of determining an appropriate partition of the kanban system. In favor of a systematic design, we identify different network partitions and explore their impact on the system performance. We consider an eight-machine line with identical machines. The processing times again are exponentially distributed with mean equal to one. We study the impact of five different partitioning strategies. Suppose that the system is to be partitioned into $P = 4$ production stages, one common strategy would be to assign an equal number of machines to each production stage. This would result in the balanced partition 2/2/2/2. Alternative partitions can be classified roughly as funnel, reversed funnel, bowl and inverted bowl. Is there a benefit in selecting a certain partition, given that the objective is to maximize throughput, and what is the impact of the network partition on the associated kanban allocation? Table 7 shows the optimal kanban allocation for each of the above partitions. For the purpose of comparison, we also display the throughput and average WIP inventory values.

The first general observation is that the shape of the network partition is being reproduced somewhat by the optimal kanban allocation. In a balanced network, for example, it appears that a uniform kanban allocation will maximize system throughput. Another important observation is that out of the five different partitioning strategies, the bowl pattern performs best with respect to throughput without resulting in extensive WIP inventories. Furthermore, it should be pointed out that the reversibility theorem established by Yamazaki and Sakasegawa (1975) for finite-buffer production lines seems to hold in its full generality in kanban systems, as it is apparent from the results related to the funnel and the reversed funnel partition in Table 7.

Network Partition		12 Kanbans for Allocation Optimal	Throughput	WIP	16 Kanbans for Allocation Optimal	Throughput	WIP
balanced	2/2/2/2	3-3-3-3	.5812	10.02	4-4-4-4	.6522	13.42
funnel	1/2/2/3	1-3-4-4	.5888	9.85	1-5-4-6	.6617	13.14
reversed funnel	3/2/2/1	4-4-3-1	.5888	10.31	6-4-5-1	.6617	13.88
bowl	1/3/3/1	1-5-5-1	.6023	10.19	1-7-7-1	.6737	13.62
inverted bowl	3/1/1/3	4-2-2-4	.5864	10.16	6-2-2-6	.6566	13.55

Table 7: Optimal Kanban Allocations in an Eight-Machine Kanban System.

6. Summary and Outlook

We have shown in this paper that a number of design rules for kanban systems can be formulated. In some cases, these design rules correspond to design rules for finite-buffer production lines which have been studied extensively in the literature (Conway *et al.* 1988; Hillier *et al.* 1993). Coincidence is seen in the optimality of symmetric allocations of buffers respectively kanbans, and in the throughput equivalence of mirror image patterns. On the other hand, the bowl phenomenon observed in finite-buffer system does not hold in kanban systems. Optimal allocations of kanbans in five- and six-machine lines exhibit an alternating pattern.

It was verified by empirical examples that in a conventional kanban system, the throughput versus WIP tradeoff is severely affected by the kanban allocation. In particular, it was demonstrated that the funnel allocation which allocates more kanbans towards the downstream machines outperforms other strategies. It appears that with the funnel allocation, one can obtain a given target throughput at the minimum level of WIP inventory. This result confirms the findings of Siha (1994) related to conventional kanban systems with random demand and is another manifestation of the non-optimality of the bowl pattern in kanban systems.

The results reported in this paper are preliminary. Further research will be necessary to confirm the proposed design rules for production lines with a higher number of machines. Furthermore, a rigorous mathematical proof is needed to verify whether the reversibility and symmetry properties that have been observed in this study will hold in general.

References

DE ARAUJO S. L., FREIN Y. AND DI MASCOLO M. (1995). Efficient Procedures for the Design of Kanban Systems. In *Proceedings of the International Conference on Industrial Engineering and Production Management. Vol II*. Marrakech, April 4-7. 511–520.

BARD J. F AND GOLANY B. (1991). Determining the Number of Kanbans in a Multiproduct, Multistage Production Systems. *International Journal of Production Research*, **29**, 881-895.

BERKLEY B. J. (1992). A Decomposition Approximation for Periodic Kanban Controlled Flow Shops. *Decision Sciences*, **23**, 291-311.

CHANG T. M. AND YIH Y. (1994). Determining the Number of Kanbans and Lotsizes in a Generic Kanban System: a Simulated Annealing Approach. *International Journal of Production Research*, **32**, 1991–2004.

CHU C. H. AND SHIH W. L. (1992). Simulation Studies in JIT Production. *International Journal of Production Research*, **30**, 2573-2586.

CIARDO G. AND MUPPALA J.K. (1991). Reference Manual for the SPNP Package Version 3.1. Duke University, Durham, NC.

CONWAY R., MAXWELL W., MCCLAIN J. O. AND THOMAS L. J. (1988). The Role of Work-in-Process Inventory in Serial Production Lines. *Operations Research*, **36**, 229-241.

DAVIS W. J. AND STUBITZ S. J. (1987). Configuring a Kanban System Using a Discrete Optimization of Multiple Stochastic Responses. *International Journal of Production Research*, **25**, 721-740.

DI MASCOLO M., FREIN Y., DALLERY Y. AND DAVID R. (1991). A Unified Modeling of Kanban Systems Using Petri Nets. *International Journal of Flexible Manufacturing Systems*, **3**, 275-307.

DI MASCOLO M., FREIN Y. AND DALLERY Y. (1992). Queueing Network Modeling and Analysis of Kanban Systems. Proceedings of 3rd International Conference on Computer Integrated Manufacturing. IEEE Society Press.

DI MASCOLO M. (1993). Analysis of a Synchronization Station for the Performance Evaluation of a Kanban System with a General Arrival Process of Demands. In *INRIA/ORSA/TIMS/SMAI Conference on Applied Probability in Engineering, Computer and Communication Sciences*. Paris, France.

DUENYAS I. AND HOPP W. J. (1993). Estimating the Throughput of an Exponential CONWIP Assembly System. *Queueing Systems*, **14**, 135–157.

GLASSERMAN P. AND YAO D. D. (1995). Structured Buffer Allocation Problems. To appear in *Discrete Event Dynamic Systems*.

HALL R. W. (1983). *Zero Inventories*. Dow-Jones Irvin, Homewood, Illinois.

HILLIER F. S., SO K. C. AND BOLING R. W. (1993). Towards Characterizing the Optimal Allocation of Storage Space in Production Line Systems With Variable Processing Times. *Management Science*, **39**, 126-133.

JOTHISHANKAR M. C. AND WANG H. P. (1992). Determination of Optimal Number of Kanbans Using Stochastic Petri Nets. *Journal of Manufacturing Systems*, **11**, 449-461.

LAMBRECHT M. AND SEGAERT A. (1990). Buffer Stock Allocation in Serial and Assembly Type of Production Lines. *International Journal of Production Management*, **10**, 47-61.

MITRA D. AND MITRANI I. (1990). Analysis of a Kanban Discipline for Cell Coordination in Production Lines I. *Management Science*, **36**, 1548-1566.

MITRA D. AND MITRANI I. (1991). Analysis of a Kanban Discipline for Cell Coordination in Production Lines II: Stochastic Demands. *Operations Research*, **39**, 807–823.

MONDEN Y. (1983). *Toyota Production System*. Industrial Engineering and Management Press, Norcross.

SCHONBERGER R. J. (1982). *Japanese Manufacturing Techniques*. The Free Press, New York.

SIHA S. (1994). The Pull Production System: Modelling and Characteristics. *International Journal of Production Research*, **32**, 933-949.

SO K. C. AND PINAULT S. C. (1988). Allocating Buffer Storages in a Pull System. *International Journal of Production Research*, **26**, 1959–1980.

SOLBERG J. J. AND NOF S. Y. (1980). Analysis of Flow Control in Alternative Manufacturing Configurations. *Journal of Dynamic Systems in Management and Control*, **102**, 141-147.

SPEARMAN M. L., WOODRUFF D. L. AND HOPP W. J. (1990). CONWIP: A Pull Alternative to Kanban. *International Journal of Production Research*, **28**, 879-894.

TAYUR S. R. (1992). Properties of Serial Kanban Systems. *Queueing Systems*, **12**, 297-318.

TAYUR S. R. (1993). Structural Properties and a Heuristic for Kanban-Controlled Serial Lines. *Management Science*, **39**, 1347-1368.

WANG H.P., ZENG L. AND JIN S. (1992). Determination of the Number of Kanbans in JIT Systems: A Petri-Net Approach. *International Journal of Advances in Manufacturing Technology*, **7**, 51-57.

YAMAZAKI G. AND SAKASEGAWA H. (1975). Properties of Duality in Tandem Queueing Systems. *Ann. Inst. Stat. Math.*, **27**, 201-212.