Proceedings of the 2001 Winter Simulation Conference
B. A. Peters, J. S. Smith, D. J. Medeiros, and M. W. Rohrer, eds.

UNDERSTANDING THE FUNDAMENTALS OF KANBAN AND CONWIP PULL SYSTEMS USING SIMULATION

Richard P. Marek
Debra A. Elkins
Donald R. Smith

Department of Industrial Engineering
Texas A&M University
238 Zachry Engineering Center
College Station, TX 77843-3131, U.S.A.

ABSTRACT

This paper presents an introductory overview and tutorial in simulation modeling and control of serial Kanban and CONWIP (CONstant Work In Process) pull systems using ARENA/SIMAN 3.5/4.0. Card level estimation is discussed for both types of pull systems, and a heuristic method to adjust card levels controlling system WIP (Work In Process) is provided. The objective is to present a tutorial for students and practicing engineers familiar with the basics of simulation, but unfamiliar with pull system fundamentals.

1 INTRODUCTION

Global manufacturing enterprises continually strive to improve their respective manufacturing operations to regain a competitive advantage particularly in the automotive and computer industries. These industries are responding to the challenge of e-commerce and customer ordering via the Internet by shifting to re-configurable manufacturing equipment and a make-to-order environment. Traditional mass production manufacturing is not particularly responsive to changing customer demands, for it relies on forecasting future demand and scheduling the release of work into the system to meet expected demand. Mass production systems often have excess inventory, higher WIP levels, and longer quoted lead-times from order to delivery. In contrast, just-in-time production relies on actual demand triggering the release of work into the system, and "pulling" work through the system to fill the demand order. Just-in-time production is better able to respond to changing customer demands, for as a production philosophy, it advocates producing the right products at the right times and in the right amounts. Re-configurable systems allow rapid and low-cost changeovers to allocate production capacity as needed to the products that are desired. Manufacturers are also moving toward modular subassemblies built off-line and delivered by suppliers as needed. Thus, a fundamental understanding of pull manufacturing and assembly systems is required to implement the make-to-order paradigm.

Industrial engineering undergraduate curriculums generally include a course on production and operations analysis, in which just-in-time and lean manufacturing principles are conceptually presented. Many students also take a course on simulation that covers a simulation language, random number generation, input modeling, verification and validation strategies, and output analysis techniques. However, there is little or no textbook material available discussing modeling, control, and analysis of pull systems using simulation. This paper attempts to address this deficiency, and can serve as a supplement for simulation and production operations courses.

Simulation models are used in this paper to illustrate the mechanics of pulling within systems, and give the reader a "hands-on" approach toward studying Kanban and CONWIP pull systems. Spearman and Zazanis (1992) provide a more advanced discussion of push, pull, and CONWIP production systems and present theoretical motivations for the improved performance of pull systems over traditional push systems. They contribute analytical results for the types of pull systems considered in this paper, and offer several conjectures that the reader is encouraged to consider while studying the pull simulation models presented herein.

(1) There is less congestion in pull systems.
(2) Pull systems are inherently easier to control than push systems but can be conceptually more difficult to model.
(3) The benefits of a pull environment owe more to the fact that WIP is bounded than to the practice of "pulling" everywhere.

2 PULL SYSTEMS: KANBAN AND CONWIP

Kanban, meaning *card* or *marker* in Japanese, is the more widely known and recognized type of pull system. A Kanban pull system is sometimes referred to as the Toyota Production System (just-in-time manufacturing using a Kanban pull system) (Monden 1981a). A Kanban pull system uses card sets to tightly control work-in progress (WIP) between each pair of workstations. Total system WIP is limited to the summation of the number of cards in each card set. Production occurs at a workstation only if raw material is available and the material has a card authorizing production. Material is pulled through the system only when it receives card authorization to move. Figure 1 illustrates a serial Kanban system. Each Kanban card set between workstations authorizes material to be pulled into the upstream workstation for processing and delivery to the downstream workstation. For example, card set 2 (between Workstations 1 and 2) authorizes an order in the paperwork queue (before Workstation 1) and raw material to be released for processing at Workstation 1, and delivery to Workstation 2.

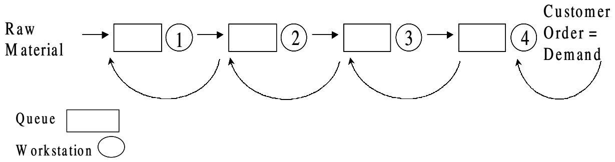

Figure 1: Kanban Pull System

In contrast, a CONWIP pull system uses a single global set of cards to control total WIP *anywhere* in the system. Material enters a CONWIP system only when demand occurs, and the raw material receives a card authorizing entrance; the same card authorizes the material to move through the system and complete production. When the final product leaves the system, the card is released, allowing new material to enter the system as new demand occurs. Notice that WIP is not controlled at the individual workstation level in the CONWIP system. Total WIP in the system is a constant (thus the name CONWIP), for the cards limit the total amount of work that can be *anywhere* in the system. The Kanban system in Figure 1 pulls work *everywhere* (between *every* pair of workstations), while the CONWIP system in Figure 2 only pulls work at the beginning of the line. Notice that in both diagrams orders are kept in a paperwork queue prior to Workstation 1 until the order and raw material receive a production and material movement authorization card.

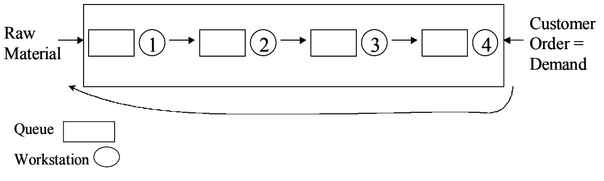

Figure 2: CONWIP Pull System

Once raw material is authorized to enter the CONWIP "black box", the material flows freely as if it were in a push system. Inside the "black box", WIP naturally accumulates in front of the bottleneck station. CONWIP systems handle a mix of parts having different bottlenecks with more ease than Kanban systems. If the bottleneck shifts as the mix of parts changes, there may be an opportunity to reduce WIP by reducing the total number of cards allocated for product flow. Conversely, cards may need to be added to increase WIP and ensure a desired throughput. CONWIP systems are easy to manage, for there is only one set of global cards that requires review and adjustment. Kanban systems are more difficult to manage but more tightly control WIP, for card control of WIP is implemented at the workstation level. If a product mix change shifts the bottleneck in a Kanban system, the number of cards allocated to each card set may require adjustment to ensure a desired throughput. In the simple four workstation example illustrated in Figure 1, if the bottleneck shifts, three different sets of Kanban cards (controlling WIP before Workstations 2, 3, and 4) must be inspected.

3 WHY CONTROL WIP?

Manufacturers have found several advantages in controlling WIP. A finite WIP capacity limits the amount of material released into the system, allowing orders to stay on paper instead of as physical material on the production floor. Production systems have a degree of flexibility that is lost when large volumes of WIP are in the physical system. Keeping orders on paper until actual production occurs facilitates execution of scheduling and design changes. Scrapping product due to a design or engineering change can be costly, especially to a company with large amounts of WIP in the system. By controlling WIP, the amount of material that needs to be scrapped or reworked is reduced, and financial losses from sales of a now inferior product are diminished.

A second advantage of WIP control is a reduction in cycle time variability. Referring to Little's Law (WIP = Cycle Time * Arrival Rate), if the arrival rate is held constant, as the level of WIP increases, the cycle time must also increase. Push systems allow the possibility of large WIP buildups, causing high variability in cycle time plus increased costs in terms of inventory buildup. Increased variability in cycle time forces companies to quote longer lead-times in order to achieve the same level of customer service. Limiting WIP reduces the variability in cycle time while allowing the pull system to still achieve the same throughput level with less WIP than a push system. To accurately quote a time from order to delivery in a pull system, the time should include both the time that the order spends on paper and the actual time in the physical production system.

4 SHOULD EVERY MANUFACTURING COMPANY USE PULL SYSTEMS?

The next question to address is should pull systems be implemented in most manufacturing facilities. Surprisingly, the answer is NO. The two types of pull systems respond slightly differently to changes in volume and product mix. The major disadvantage for both types of pull systems is that they require fairly steady product flow. Kanban is typically restricted to repetitive manufacturing where material flows at a steady rate in a fixed path. Large variations in volume or product mix destroy the flow and undermine the system's performance goals. If there is too much WIP, the goal of minimizing WIP in the system is not achieved, and financial flexibility in dealing with scheduling and engineering changes is lost. If there is too little WIP, throughput goals cannot be attained. CONWIP, while still requiring a relatively steady volume, is a little more resilient in handling changes in product mix. The difference between their capabilities of handling product mixes has to do with the individual products having different bottlenecks and how WIP is controlled within the system. Questions to consider when assessing whether a pull system should be adopted include:

- How often do design, engineering and schedule changes occur?
- What are the economic consequences of maintaining the current system compared to converting to a pull system?
- Can a pull system reduce overall lead-time compared to a push system?
- Are suppliers reliable enough to support just-in-time delivery of raw materials or sub-components?
- Is the production system reliable, or does it suffer frequent breakdowns that stop production?
- Are labor and management committed to making the changes needed?
- How often and how significantly does the product mix change?

In situations where a pull system is found to be acceptable for a facility, a decision of which type of pull system to implement must be made. As discussed previously, the choice depends on the level of WIP control desired (at the individual workstation level, or a "black box" system level).

5 SIMULATION MODELS OF KANBAN AND CONWIP PULL SYSTEMS

Simulation models have been developed in Arena 3.5 and tested in Arena 4.0 for the Kanban and CONWIP systems in Figures 1 and 2 respectively (Marek, 2000). The reader is assumed familiar with the basics of simulation programming and analysis. The code for these models is presented in the following sections for the reader to obtain a "hands-on" feel for the different pull mechanics in each system.

The serial manufacturing systems being modeled contain four workstations, and must produce two types of products. The make-to-order production facility has re-configurable manufacturing equipment, allowing rapid and low cost changeovers to switch between product types. The setup times for changing between product types are considered to be zero on the assumption that the products are quite similar. This is a realistic assumption, for production line designers are now examining the value of agile tooling, fixtures, and material handling, so that any part in a general family may be produced on the line if the designed part fits within the line's production envelope. For this reason, product types are not batch processed on a forecasted basis, but are processed on a first-come first-serve (FCFS) basis as orders arrive. Product types are assigned from a discrete probability distribution for each arriving order with 70% type 1 and 30% type 2. Process times at each workstation may depend on product type. Machine breakdowns and supply chain failures are currently not considered.

The variance reduction technique of Common Random Numbers (CRN) (Pegden, et al., 1995) is employed to synchronize usage of random numbers in the Kanban and CONWIP systems so that the systems are compared under similar conditions. Each system observes the same sequence of arrivals of type 1 and type 2 jobs and uses the same processing times for jobs at each workstation. This approach is often justified for scenario analysis whereby the analyst seeks to compare two or more alternatives (systems) and control specified parameter sequences while permitting other system parameters to vary. By designing the various simulation runs, the analyst can better distinguish the impact(s) of specific changes in the scenarios.

Throughout the remainder of this paper, specific ARENA modeling constructs are used to define the modeling approach. The ARENA SEEDS element controls the six random number streams used (See Table 1). By using common random numbers, randomness in experimental conditions is reduced, and any measured differences in the two systems are due to the pull behavior and card control level used.

Table 1: Random Number Streams

Stream	Seed	Purpose
1	2323	Job Inter-Arrival Times
2	4545	Workstation 1 Processing Times
3	8080	Workstation 2 Processing Times
4	8181	Workstation 3 Processing Times
5	1717	Workstation 4 Processing Times
6	1974	Job Type

5.1 The Arrival Rate and the Shifting Bottleneck

The arrival rate of orders is taken arbitrarily to be 1/54 orders per minute. This arrival rate is of interest, for the paperwork queue (queue before Workstation 1) explodes if only part type 1 or part type 2 is processed. Considering product mix is important, for by construction, the bottleneck also shifts if only one type of part is processed. If only part type 1 is processed, Workstation 3 is the bottleneck; if part type 2 is processed, Workstation 4 is the bottleneck. For the product mix as stated and orders processed FCFS, the *system bottleneck* is Workstation 3, and the paperwork queue is relatively stable (does not explode).

5.2 Bottleneck Determination

Bottleneck determination is straightforward for both types of serial pull systems. Ignoring machine breakdowns, and assuming no scrap or rework occurs, the bottleneck is the workstation with the highest utilization. Suppose that 24 cards are allotted for each Kanban card set. After running the Kanban model for a replication length of 96000 minutes with a warm-up of 64000 minutes, Workstation 3 can be verified to be the bottleneck, with 99.213% utilization (compared to utilizations of 38.249%, 57.247%, and 81.421% at Workstations 1, 2, and 4 respectively). Similarly, if a total of 30 cards is allotted for the CONWIP system, and the CONWIP model is run for a replication length of 96000 minutes with a warm-up of 64000 minutes, Workstation 3 is again the bottleneck with 99.23% utilization (compared to utilizations of 38.16%, 57.28%, and 81.44% at Workstations 1, 2, and 4 respectively).

5.3 Measuring Workstation Utilization

In the Kanban model, a card and workstation are seized simultaneously. As soon as processing completes, the workstation is released. However, the current card is retained, until the part receives the next card authorizing movement to the next workstation. The ARENA SEIZE-RELEASE sequence allows a more accurate measure of workstation utilization for Kanban pull systems. Each workstation processes only when authorized to do so, and is busy only for the process time duration. In the CONWIP model, the workstation is seized when available and released as soon as the processing time is complete. The SEIZE-RELEASE pattern in the CONWIP system also yields an accurate measure of workstation utilization for the CONWIP system.

5.4 One Card or Two Cards?

The Kanban pull model demonstrates a 1-card Kanban system with 24 cards assigned to control WIP before each of Workstations 2, 3, and 4. The CONWIP pull model is also a 1-card model, with a total of 30 cards allotted to control WIP. 1-card systems are the easiest to understand and implement, and use the same card to authorize material movement and production. 2-card systems are similar to 1-card systems, but use 2 different types of cards to control production and material movement separately. The codes can be modified appropriately to implement a 2-card level of control.

5.5 Blocking After Service

Card control in a Kanban system can cause a workstation to become idle, even if it has raw material to process. This idleness is due to blocking after service. The blocked workstation is forced to stop production because there are no available cards to pull work from the current workstation. Card control at the individual workstation level introduces an additional level of dependence between the workstations. The simpler card control structure in CONWIP systems does not introduce the additional workstation dependency nor cause blocking after service. Since a CONWIP system behaves as a push system inside the black box, each workstation will continue to process work as long as there is work in the queue before it. WIP will tend to accumulate in front of the bottleneck workstation. However, queue explosion does not occur as in a push system, since card control limits total WIP.

6 KANBAN SIMULATION MODEL

The Kanban pull model is presented first to illustrate the amount of coding required to implement the tight control of WIP at each workstation. Each set of Kanban cards controlling WIP between a pair of workstations is modeled as a RESOURCE. DSTATS are collected on average queue lengths, and average workstation utilizations. Since raw material always has a Kanban card attached when it is in the physical production system, the average system WIP level can be observed by measuring the sum of the utilizations of the three card RESOURCES. The reader should carefully study the card SEIZE-RELEASE sequence at each workstation to better understand the pull mechanics and card control implemented for each workstation. The SIMAN code for this model is presented in sections 6.2 and 6.3.

6.1 Estimating the Number of Cards Needed to Control a Kanban Pull System

The number of cards used to control WIP in a Kanban system can be estimated using a modification of a formula from Monden (1981b). Monden's modified formula for the number of cards is given by $\#cards = \frac{\overline{D} \times L + w}{2a}$, where $\overline{D}$ is the expected demand per unit time (approximated by the arrival rate = 1/54 orders per minute), L is the quoted lead-time to the customer from time of order to delivery, w is a buffer stock variable assumed to be 10% of $\overline{D} \times L$, and $a = 1$ is the container capacity (number of orders controlled per Kanban card). Monden's original formula does not have a factor of 2 in the denominator of the fraction. The formula is modified for this application, because the same card is used to authorize both production and material movement. Monden uses 2 different cards to control production and movement in a Kanban system, so the number of cards required is doubled. The lead-time L can be estimated from the simulation model to be at least the sum of the average time in the paperwork queue and the average time in physical production. For example, if the quoted lead-time is 2300 minutes (time in paperwork queue + time in physical system is on average 2245.3 minutes), then the total number of cards needed is approximately 24 cards. The drawback to this estimation method is that it does not specify how to allocate the 24 cards total across the 3 Kanban card sets needed to manage WIP at the workstation level. One can use the estimated card level as the initial number of cards needed per Kanban card set at each workstation.

6.2 Kanban Pull System: Experiment Frame

The ARENA modeling environment consists of two specific interrelated systems (frames): The modeling frame and the experimental frame. The modeling frame is where the actual logic of entity flow takes place and the experiment frame defines the operational parameters and the collection of specific statistical values. For the presented example, the source code for the experiment frame follows.

```
PROJECT, Kanban24cards, M-E-S,,Yes;
ATTRIBUTES:   1,TimeIn:
              2,Type:
              3,Tout:
              4,PaperTime;
VARIABLES:    1,Mean1(2,1),60,30:
              3,Mean2(2,1),40,50;
SEEDS:        1,2323,No:
              2,4545,No:
              3,8080,No:
              4,8181,No:
              5,1717,No:
              6,1974,No;
QUEUES:
4,Workstation1Q,FirstInFirstOut:
5,Workstation2Q,FirstInFirstOut:
6,Workstation3Q,FirstInFirstOut:
7,Workstation4Q,FirstInFirstOut;
RESOURCES:
1,Card2,Capacity(24):
2,Card3,Capacity(24):
3,Card4,Capacity(24):
4,Workstation1,Capacity(1):
5,Workstation2,Capacity(1):
6,Workstation3,Capacity(1):
7,Workstation4,Capacity(1);
COUNTERS:
1,Type1Entering,,Replicate:
2,Type2Entering,,Replicate:
3,Type1Completed,,Replicate:
4,Type2Completed,,Replicate:
5,TotalEntering,,Replicate:
6,TotalCompleted,,Replicate;
TALLIES:
1,TimeInSys,"TimeInSysCom.dat":
2,TimeInPaperQ;
DSTATS:
1,NR(Workstation1), WS1 Utilization:
2,NR(Workstation2), WS2 Utilization:
3,NR(Workstation3), WS3 Utilization:
4,NR(Workstation4), WS4 Utilization:
5,(NR(Card2)+NR(Card3)+NR(Card4)),
TotalWIP,"TotalWIPcom.dat":
6,NQ(Workstation1Q),Queue
WS1,"Queue1.dat":
7,NQ(Workstation2Q),Queue WS2:
8,NQ(Workstation3Q),Queue WS3:
9,NQ(Workstation4Q),Queue WS4;
REPLICATE,    1,0.0,96000,No,Yes,64000;
```

6.3 Kanban Pull System: Model Frame

```
0$ CREATE, 1:Expo(54,1):
MARK(PaperTime);Create entities
21$ COUNT: TotalEntering,1;
Count entities entering
20$ ASSIGN:
Type=Disc(0.7,1,1.0,2,6):
Tout=(Type+2);    Assigns type
S COUNT:Type,1; Count by type entering
1$ QUEUE, Workstation1Q; Queue for WS1
2$ SEIZE, 1:
Workstation1,1:
Card2,1;       Seize WS1 and Card2
```

```
23$ TALLY:
TimeInPaperQ,INT(PaperTime),1;
Time in Paper Queue
3$ DELAY:
Norm(20,2,2):MARK(TimeIn);
Delay by process time
4$ RELEASE: Workstation1,1;
5$ QUEUE, Workstation2Q; Queue for WS2
6$ SEIZE, 1:
Workstation2,1:
Card3,1; Seize WS2 and Card3
24$ RELEASE: Card2,1; Release Card2
7$ DELAY: Tria(20,30,40,3);
Delay by process time
8$ RELEASE: Workstation2,1;
9$ QUEUE, Workstation3Q; Queue for WS3
10$ SEIZE, 1:
Workstation3,1:
Card4,1; Seize WS3 and Card4
25$ RELEASE: Card3,1; Release Card3
11$ DELAY: Expo(Mean1(Type,1),4);
Delay by process time
12$ RELEASE: Workstation3,1;
13$ QUEUE, Workstation4Q;Queue for WS4
14$ SEIZE, 1:
Workstation4,1;   Seize Workstation4
15$ DELAY: Expo(Mean2(Type,1),5);
Delay by process time
16$ RELEASE: Workstation4,1:
Card4,1; Release WS4 and Card4
17$ TALLY: TimeInSys,INT(TimeIn),1;Time
in System
19$ COUNT: Tout,1; Count by type
completed
22$ COUNT: TotalCompleted,1; Total
completed
18$ DISPOSE; Dispose of part
```

7 CONWIP SIMULATION MODEL

Only minor code modifications (presented in sections 7.2 and 7.3) must be made to change the Kanban pull system to a CONWIP pull system. The RESOURCES and DSTATS elements in section 7.2 replace the RESOURCES and DSTATS elements defined for the Kanban system previously. The model code in section 7.3 is presented in its entirety to highlight that only one global set of cards must be used to control system WIP. Further, the reader should notice that there is no card control for material moving between workstations in the CONWIP model. Card authorization for production and material movement is obtained prior to leaving the paperwork queue (before Workstation 1). The part keeps the card for the entire production route, and releases the card upon completing production.

7.1 Estimating the Number of Cards Needed to Control a CONWIP Pull System

Hopp and Spearman (1996) present a formula for system throughput that can be used to estimate the number of cards w needed to control a CONWIP system. They define throughput TH as a function of the number of cards w by $TH(w) = \frac{wr_b}{w + W_0 - 1}$, where r_b represents the rate of the bottleneck workstation in jobs per minute, and W_0 is the WIP level attained for a line with maximum throughput operating at the rate of the bottleneck. Using the type dependent mean processing times for each workstation, and weighting the means by the percentage of type 1 or type 2 jobs, it can be shown that Workstation 3 has the longest average processing time of 51 minutes (compared to Workstation 1 at 20 minutes, Workstation 2 at 30 minutes, and Workstation 4 at 43 minutes). Thus the bottleneck rate is r_b = 1/51 jobs per minute. The critical WIP level $W_0 = r_bT_0$, where T_0 is the sum of the average processing times of the workstations, ignoring any processing time variability, blocking, machine breakdowns, and supply chain failures. Here T_0 = 144 minutes, so that W_0 = 144/51 = 2.8224 jobs. The CONWIP system acts inside the black box as a push system, so the throughput rate may be taken to be equal to the arrival rate of 1/54 jobs per minute (i.e. $TH(w)$ = 1/54). Note that the maximum output (throughput) rate of the "black box" is always less than or equal to the input (arrival) rate. Using Hopp and Spearman's formula, and solving for the remaining unknown w, the number of cards needed to control the CONWIP system can now be estimated. For this particular example, w = 30.68 cards, a close estimate for the 30 cards specified to control WIP in the simulation model. In this case the number of cards is rounded to the nearest integer, and experience with the model allows the number of cards to be set at 30 rather than 31.

7.2 CONWIP Pull System: Experiment Frame Code Modifications

```
RESOURCES: 1,Card,Capacity(30):
4,Workstation1,Capacity(1):
5,Workstation2,Capacity(1):
6,Workstation3,Capacity(1):
7,Workstation4,Capacity(1);
DSTATS:
1,NR(Workstation1), WS1 Utilization:
2,NR(Workstation2), WS2 Utilization:
3,NR(Workstation3), Ws3 Utilization:
4,NR(Workstation4), WS4 Utilization:
5,NR(Card),Total WIP,"TotalWIPcom.dat":
6,NQ(Workstation1Q),Queue
WS1,"Queue1.dat":
7,NQ(Workstation2Q),Queue WS2:
```

```
8,NQ(Workstation3Q),Queue WS3:
9,NQ(Workstation4Q),Queue WS4;
```

7.3 CONWIP Pull System: Model Frame

```
Stream1  CREATE,
1:Expo(54,1):MARK(PaperTime); Create
entity
20$ COUNT: TotalEntering,1; Count
entities entering
19$ ASSIGN: Type=Disc(0.7,1,1.0,2,6):
Tout=(Type+2); Assigns type
S COUNT: Type,1; Count by type entering
0$ QUEUE, Workstation1Q; Queue for WS1
1$ SEIZE, 1:
Workstation1,1:
Card,1;  Seize Workstation1 and a Card
22$ TALLY:
TimeInPaperQ,INT(PaperTime),1; Time in
Paper Queue
2$ DELAY:
Norm(20,2,2):MARK(TimeIn);Delay by
process time
3$ RELEASE: Workstation1,1;
4$ QUEUE, Workstation2Q; Queue for WS2
5$ SEIZE, 1:
Workstation2,1; Seize Workstation2
6$ DELAY: Tria(20,30,40,3); Delay by
process time
7$ RELEASE: Workstation2,1;
8$ QUEUE, Workstation3Q; Queue for WS3
9$ SEIZE, 1:
Workstation3,1; Seize Workstation3
10$ DELAY: Expo(Mean1(Type,1),4); Delay
by process time
11$ RELEASE: Workstation3,1;
12$ QUEUE,Workstation4Q; Queue for WS4
13$ SEIZE, 1:
Workstation4,1; Seize Workstation4
14$ DELAY: Expo(Mean2(Type,1),5); Delay
by process time
15$ RELEASE:
Workstation4,1:
Card,1; Release Workstation and a Card
16$ TALLY: TimeInSys,INT(TimeIn),1;
Time in System
18$ COUNT: Tout,1; Count by type
completed
21$ COUNT: TotalCompleted,1; Total
orders completed
17$ DISPOSE; Dispose of part
```

8 HEURISTIC TO REDUCE CARD LEVELS

In this section a heuristic strategy to reduce card levels in the Kanban or CONWIP pull simulation models is discussed in detail. Numerical results for card reductions in each of the Kanban and CONWIP systems are presented to allow the reader to try the card reduction heuristics. The reader should observe that the CONWIP heuristic is a simplification of the Kanban heuristic. However, the two versions are presented separately to help the reader in studying the card reduction process.

The key to the card reduction heuristic is to reduce system WIP by reducing the number of cards while still meeting or exceeding a desired throughput goal. Cards must be available for material to be pulled from upstream into the bottleneck, so that the bottleneck does not “starve”. In addition, downstream workstations must have enough cards to pull processed material from the bottleneck, so the bottleneck is not blocked. Utilization levels and processing rates at the workstations may change slightly as the card levels are reduced. While these parameters measure system performance as card levels change, the main performance measure is still the total throughput of the system. The heuristics provided next guide the reader through the card reduction process. The heuristics may not be optimal, but they do provide an algorithmic strategy to reduce system WIP that terminates in finitely many steps.

8.1 Kanban Card Reduction Heuristic

1) Estimate the number of cards needed per card set in the Kanban pull system using the analytical formula from section 6.1.
2) Using the estimated number of cards per card set, find the current workstation utilizations and system output levels (number of parts completed over the simulation runtime).
3) Begin the card elimination process at the workstation with the highest utilization (i.e. the bottleneck workstation). Drop the card level incrementally until the card reduction lowers the system throughput below the desired goal. Then add one card back to restore throughput to a level that meets or exceeds the desired goal. Larger card decreases may be used initially to speed up the card reduction process.
4) Repeat the card elimination process with the workstation with the next highest utilization.
5) Continue the elimination process until all workstations (including the bottleneck station) have been considered for card reduction.

8.2 Card Reduction Example for the Kanban System

Based on Kanban card estimation in section 6.1, the Kanban system starts with 24 cards allocated to each card set. Since card sets are defined as RESOURCES in the experiment frame of the model, it is easy to change the card levels for the Kanban system. The card reduction process and some key measures are presented in Tables 2, 3, and 4 to allow the reader to try the card reduction process and verify results. The Kanban system must attain a throughput of at least 1767 orders completed in 96000 minutes. Note that this throughput goal is chosen somewhat arbitrarily, but is also the throughput rate that is obtained if 10 cards are allocated to each card set.

The card allocation of 24 cards to each card set is denoted 24-24-24. Card reduction begins at Workstation 3, the workstation with the highest utilization. Table 2 shows the card reduction process at Workstation 3. Cards are reduced in card set 3 in the sequence 24, 14, 10, 6, 3. Using 3 cards yields a throughput of at least 1767 orders in 96000 minutes (actually 1777 orders in 96000 minutes). Results for 2 cards in card set 3 are not presented as the paperwork queue explodes, exceeding the Arena student version limits.

Table 2: Kanban Card Reduction Results at Workstation 3 (Card 3)

	24-24-24	24-14-24	24-10-24	24-6-24	24-3-24
Avg. Time in PaperQ	6.4961	26.101	52.693	122.73	743.55
Avg. Time in System	1536.7	1516.8	1489.8	1495.1	1560.6
Avg. WS1 Utilization	0.38249	0.38249	0.38249	0.38126	0.37536
Avg. WS2 Utilization	0.57247	0.56930	0.56930	0.56516	0.55625
Avg. WS3 Utilization	0.99213	0.99213	0.99213	0.99005	0.97272
Avg. WS4 Utilization	0.81421	0.81421	0.81421	0.81237	0.80187
Avg. Total WIP	29.178	28.803	28.296	28.366	29.163
Total Entering	1833	1833	1833	1833	1833
Type 1 Entering	1284	1284	1284	1284	1284
Type 2 Entering	549	549	549	549	549
Total Completed	1805	1805	1805	1801	1777
Type 1 Completed	1269	1269	1269	1265	1247
Type 2 Completed	536	536	536	536	530

Next, card reductions are attempted at Workstation 4, which has the second highest utilization. Initially, the card reduction down to 14 cards is too great, and the throughput goal is not attained. Adding one card back to the card resource at Workstation 4 yields exactly the desired throughput of 1767 orders completed per 96000 minutes. Table 3 summarizes the card reduction process at Workstation 4. The card level controlling Workstation 4 has been reduced to 15 cards.

Table 3: Kanban Card Reduction Results at Workstation 4 (Card 4)

	24-3-14	24-3-15
Avg. Time in PaperQ	952.28	884.96
Avg. Time in System	1561.2	1558.6
Avg. WS1 Utilization	0.37283	0.37342
Avg. WS2 Utilization	0.55230	0.55316
Avg. WS3 Utilization	0.96427	0.96658
Avg. WS4 Utilization	0.79607	0.79761
Avg. Total WIP	28.937	28.959
Total Entering	1833	1833
Type 1 Entering	1284	1284
Type 2 Entering	549	549
Total Completed	1762	1767
Type 1 Completed	1233	1238
Type 2 Completed	529	529

Finally, card reductions are considered at Workstation 2, with results summarized in Table 4. The card set before Workstation 2 can be reduced down to one card, and the system still achieves the throughput goal. Note that a minimum of one card is needed in each card set to authorize production and movement at the workstation. The final card set levels are 1-3-15, or 1 card in card set 2, 3 cards in card set 3, and 15 cards in card set 4. For this particular system, the large number of cards in card set 4 indicates it is important not to block the bottleneck workstation. The system WIP controlled by the card sets has been reduced from 72 orders down to 19 orders. The results for card allocation 1-3-15 show that the card reduction process significantly decreases the time spent in actual processing, while increasing the amount of time the order spends on paper. As noted previously, the economic benefits of the order remaining as paperwork until actual production outweigh the fact that the order has to wait longer before processing.

Table 4: Kanban Card Reduction Results at Workstation 2 (Card 2)

	14-3-15	6-3-15	2-3-15	1-3-15
Avg. Time in PaperQ	1323.3	1720.1	1931.0	1985.4
Avg. Time in System	1115.6	715.56	502.63	448.56
Avg. WS1 Utilization	0.37139	0.36975	0.36888	0.36870
Avg. WS2 Utilization	0.55316	0.55316	0.55316	0.55316
Avg. WS3 Utilization	0.96658	0.96658	0.96658	0.96654
Avg. WS4 Utilization	0.79761	0.79761	0.79761	0.79761
Avg. Total WIP	20.661	13.208	9.2582	8.2593
Total Entering	1833	1833	1833	1833
Type 1 Entering	1284	1284	1284	1284
Type 2 Entering	549	549	549	549
Total Completed	1767	1767	1767	1767
Type 1 Completed	1238	1238	1238	1238
Type 2 Completed	529	529	529	529

8.3 CONWIP Card Reduction Heuristic

In this section, the simplified card reduction heuristic for the CONWIP system is presented.

1) Estimate the total number of cards needed for the CONWIP pull system using the analytical formula from section 7.1.
2) Using the estimated global card level, find the current workstation utilizations and system output levels (number of parts completed over the simulation runtime).
3) Begin the global card reduction process. Drop the global card level incrementally until the card reduction lowers the throughput level below the desired goal. Then add one card back to meet or exceed the throughput goal. Larger card decreases may be used initially to speed up the card reduction process.

8.4 Card Reduction Example for the CONWIP System

The initial CONWIP system starts with 30 cards allocated to control system WIP. Again, the card level is adjusted by changing the card RESOURCE in the experiment frame of the CONWIP model. The card reduction process and some key measures are presented in Table 5 to allow the reader to try the card reduction process and verify results. For comparison purposes, the CONWIP system must attain at least the same throughput as the Kanban system (1767 orders completed in 96000 minutes). From the results presented in Table 5, the CONWIP system needs 12 cards to obtain 1770 orders completed in 96000 minutes. The reader should also note that as the number of cards decreases, the time in paperwork queue increases, and the time in the physical system decreases. These results indicate that as the system WIP is reduced, the orders spend more time on paper, and less time as raw material on the shop floor.

Table 5: CONWIP Card Reduction Results

	30 cards	25 cards	20 cards	15 cards	12 cards	11 cards
Avg. Time in PaperQ	180.99	344.67	575.01	1061.80	1598.50	1897.50
Avg. Time in System	1346.00	1204.00	1010.40	785.43	641.39	592.14
Avg. WS1 Utilization	0.38157	0.38004	0.37877	0.37439	0.37069	0.36779
Avg. WS2 Utilization	0.57278	0.57109	0.56892	0.56242	0.55612	0.55172
Avg. WS3 Utilization	0.99230	0.99189	0.99021	0.98023	0.96805	0.96202
Avg. WS4 Utilization	0.81438	0.81398	0.81246	0.80635	0.799	0.79493
Avg. Total WIP	25.564	22.813	19.076	14.685	11.849	10.868
Total Entering	1833	1833	1833	1833	1833	1833
Type 1 Entering	1284	1284	1284	1284	1284	1284
Type 2 Entering	549	549	549	549	549	549
Total Completed	1804	1804	1802	1785	1770	1759
Type 1 Completed	1268	1268	1266	1253	1241	1231
Type 2 Completed	536	536	536	532	529	528

9 CONCLUDING REMARKS

At this point, the reader should feel comfortable with the basic concepts, modeling, and card reduction techniques for Kanban and CONWIP pull systems. The major advantages of implementing a pull system include reduced cycle time variability, and economic flexibility to make engineering and design changes. While Kanban systems maintain tighter control of system WIP through the individual card resources at each workstation, CONWIP systems are easier to implement and adjust, since only one set of system cards is used to manage system WIP. The card reduction strategy discussed also demonstrates how simulation can be used as an effective decision support tool for production operations.

Additionally, modeling pull systems with virtually any simulation language can present challenges to the analyst in that one must be somewhat innovative in the construction of the model and fully understand how to apply the given modeling constructs to effect a valid model. ARENA was chosen as the underlying simulation language because of its wide applicability in industry, and its ease-of-use as a teaching language. The authors' experience is that it is straightforward to learn additional simulation languages after learning concepts of process flow and modeling techniques using a first simulation language.

The Kanban and CONWIP pull systems logic should be relatively easy to implement in other simulation languages (such as AutoMod, Witness, ProModel, Simul8, etc.) that specialize in modeling manufacturing process flows. Thus, by studying the example problem contained herein, a greater insight and appreciation for the logic and application of the modeling constructs (especially in the ARENA frame) are obtained.

REFERENCES

Hopp, Wallace J. and Mark L. Spearman, *Factory Physics*, Irwin, Chicago, Illinois, 1996.

Marek, Richard P. "Understanding Pull Systems," unpublished Master's Thesis, Dept. of Ind. Eng., Texas A&M Univ., College Station, TX, Dec., 2000.

Monden, Yasuhiro, "What Makes the Toyota Production System Really Tick?" *Industrial Eng.*, Vol. 13(1), pp. 36-46, Jan. 1981a.

Monden, Yasuhiro, "Adaptable Kanban System Helps Toyota Maintain Just-In-Time Production," *Industrial Eng.*, Vol. 13(5), pp. 28-46, May 1981b.

Pegden, C. Dennis, Robert E. Shannon, and Randall P. Sadowski, *Introduction to Simulation using SIMAN*, 2nd Ed., McGraw-Hill, Inc. Ohio, 1995.

Spearman, Mark L. and Michael A. Zazanis, "Push and Pull Production Systems: Issues and Comparisons," *Op. Res.*, Vol. 40(3), pp. 521-532, May-June 1992.

AUTHOR BIOGRAPHIES

RICHARD P. MAREK is a Manufacturing Engineer at Ford Motor Company in Dearborn, MI. He received his B.S. and M.S. in Ind. Eng. from Texas A&M. Mr. Marek's research interests include management and control of manufacturing systems, facility planning and layout analysis. His email is `<rmarek2@ford.com>`.

DEBRA A. ELKINS is a Sr. Research Engineer at General Motors R&D Center in Warren, MI. She received her Ph.D. in Ind. Eng from Texas A&M. Dr. Elkins' research interests include the decision sciences, computational probability, and simulation modeling and analysis of production systems. Dr. Elkins is the contact author for this paper, and may be reached at General Motors R&D Center, Mail Code 480-106-359, 30500 Mound Road, Warren, MI 48090-9055. Her email is `<debra.elkins@gm.com>`.

DONALD R. SMITH is an Assoc. Professor of Ind. Eng. at Texas A&M. Dr. Smith has a B.S., M.S. and Ph.D. degree in Ind. Eng. from the Univ. of Arkansas-Fayetteville. He is a registered professional engineer in the State of Arkansas and a senior member of IIE. His research and teaching interests include simulation modeling and analysis, engineering management, and engineering economics. He is currently the director of graduate distance learning for the Ind. Eng. Dept. at Texas A&M. His email is `<dr-smith@tamu.edu>`.