On the Meaning of ConWIP Cards:
An Assessment by Simulation

Matthias Thürer (corresponding author), Nuno O. Fernandes, Nick Ziengs and Mark Stevenson

Name: Prof. Matthias Thürer
Institution: Jinan University
Address: Institute of Physcial Internet
School of Electrical and Information Engineering
Jinan University (Zhuhai Campus)
519070, Zhuhai, PR China
E-mail: matthiasthurer@workloadcontrol.com

Name: Prof. Nuno O. Fernandes
Institution: Instituto Politécnico de Castelo Branco
Address: Av. do Empresário, 6000-767
Castelo Branco - Portugal
E-mail: nogf@ipcb.pt

Name: Nick Ziengs
Institution: University of Groningen
Address: Department of Operations
Faculty of Economics and Business
University of Groningen
9742 EM Groningen
The Netherlands
E-mail: n.ziengs@rug.nl

Name: Prof. Mark Stevenson
Institution: Lancaster University
Address: Department of Management Science
Lancaster University Management School
Lancaster University
LA1 4YX - U.K.
E-mail: m.stevenson@lancaster.ac.uk

Keywords: *Order Release; Production Control; ConWIP (Constant Work-in-Process).*

On the Meaning of ConWIP Cards: An Assessment by Simulation

Abstract

The simplicity of ConWIP (Constant Work-In-Process) makes it one of the most widely adopted card-based production control solutions. Its simplicity however also limits the opportunities that are available to improve the concept. There are arguably only two major search directions: (i) to alter the meaning of cards away from controlling jobs; and (ii) to adopt alternative, more sophisticated backlog sequencing rules. In this study, we outline a simple, practical load-based ConWIP system that changes the meaning of cards. Rather than controlling the number of jobs, cards are associated with a certain amount of workload. Simulation results demonstrate the positive performance impact of limiting the total shop load. The Workload Control literature advocates the use of a corrected load measure as it better represents the direct load queuing at a station; but this worsens performance when compared to a shop load measure in the context of ConWIP.

Keywords: *Order Release; Production Control; ConWIP (Constant Work-in-Process).*

1. Introduction

Constant Work-in-Process (ConWIP; e.g. Spearman *et al*., 1990; Hopp & Spearman, 2001) is a simple card-based production control system. It is essentially a pull system (Hopp & Spearman, 2004) that uses a Work-In-Process (WIP) limit or cap (WIP-Cap) to realize input/output control (e.g. Plossl & Wight, 1971). In accordance with input/output control, the output of work from the shop floor determines the input of work to the shop floor. Jobs are only permitted to enter the shop floor if the WIP-Cap, which is pre-established by management, is not violated; otherwise, they have to wait in a so-called 'backlog' (Spearman *et al*., 1990) until a job on the shop floor has been completed. Cards circulate between the exit from the shop floor and the backlog or entry point. The return of a card signals that one job has been completed (output), and another can be released (input).

ConWIP is an effective means of exercising pull control providing that product variety is restricted – its applicability to high-variety make-to-order environments is therefore rather limited (Thürer *et al*., 2016). There are two key reasons for this: (i) ConWIP's simple loop structure, which contains all possible stations in the routing of jobs within one loop, requires short routings and complete routing homogeneity to ensure effective control (Hopp & Spearman 2001); and (ii) ConWIP's lack of load balancing capabilities requires low levels of processing time variability (Germs & Riezebos, 2010). These two weaknesses have been a key focus of the extant literature on ConWIP. For example, a backlog sequencing rule has been used to enhance ConWIP's ability to balance the workload across resources (Thürer *et al*., 2017a). In this study, we extend this literature by arguing that further improvement can be obtained by changing the meaning of cards such that they represent a certain contribution to the workload.

The original ConWIP cards were job anonymous, i.e. they signal that a job can be released but they did not indicate what kind of job (Spearman *et al*., 1990). The motivation behind this was that product specific cards, as used in *kanban* systems, require a large number of cards to be managed and maintained when product variety is high. Thus by making cards job anonymous, only a single card type was needed. This unique characteristic of ConWIP was questioned by Duenyas (1994) who introduced m-ConWIP. In m-ConWIP, cards are again product specific (like *kanban* cards). This overcomes the restrictions on routing variability for the original ConWIP system and even led to improvements in terms of load balancing in Germs & Riezebos (2010). However, it re-introduces the limitation on product variety. Moreover, it is argued here that this adaptation only addressed the lack of load balancing capability that is caused by routing variability and not in terms of processing time variability.

Load balancing is here defined as the balancing of the workload across resources and is thus also influenced by variety in the workload of jobs and not just the routing.

We argue that changing the meaning of cards away from anonymous jobs to a workload contribution can address load balancing issues caused by processing time variability while at the same time maintaining the advantage of ConWIP, allowing for high product variety. In other words, cards represent a workload or quantum of work (rather than 'a job') as is the case in the literature on some other card based systems, including Control of Balance by Card Based Navigation (COBACABANA; e.g. Land, 2009; Thürer *et al*. 2014) and Paired-cell Overlapping Loops of Cards with Authorization (POLCA; e.g. Suri, 1998; Riezebos, 2010). The objective of this study is twofold. First, we outline a simple, practical load-based ConWIP system that changes the meaning of cards. Second, we use controlled simulation experiments to assess the potential of this refinement to improve the performance of ConWIP in a general flow shop that produces to-order; i.e. a type of shop environment characterized by high product variety, high routing variety, and high processing time variability.

The remainder of this paper is structured as follows. In Section 2, we review the mechanisms underlying ConWIP and discuss refinements proposed in the ConWIP literature. Section 3 then outlines our load-based ConWIP system. The simulation model used to evaluate the impact of our refinement is then described in Section 4 before the results are presented, discussed, and analyzed in Section 5. Finally, conclusions are summarized in Section 6, where managerial implications and future research directions are also outlined.

2. Background – The ConWIP Production Control System

ConWIP (Spearman *et al*., 1990; Hopp & Spearman, 2001) – as illustrated in Figure 1 – is arguably the simplest card-based control system available in the literature. Whenever the number of jobs in the system (or on the shop floor) is below a pre-established limit, a new job is released to the system. In order to control the number of jobs, each job in the system has to have a ConWIP card attached to it. Thus, by restricting the number of cards that can circulate in the system, the number of jobs is also restricted. Once a job leaves the system, its card is freed and can be used by a different job from the set of jobs waiting to enter the system. ConWIP cards are job (or product) anonymous. This means that the material control system only signals the need to release a new job to the shop floor irrespective of the actual product type or its requirements (Riezebos, 2010). The place where these jobs wait is referred to as the backlog in the ConWIP literature (Spearman *et al*., 1990). Clearly, for the workload to be controlled jobs have to be homogenous, which makes ConWIP less suitable for high-variety

contexts. This has triggered a broad literature that has attempted to enhance ConWIP's applicability by refining the original concept.

[Take in Figure 1]

This study focuses on ConWIP in a balanced shop, i.e. where capacity is evenly distributed across stations, since load balancing across resources is less important in shops with stationary bottleneck(s) (Thürer *et al*. 2017b). Given this focus, ConWIP extensions, such as ConWork or ConLoad (Rose, 1999), that presuppose a stationary bottleneck are not considered when discussing existing refinements. We also do focus on a single shop that provides a high variety of products rather than on an assembly shop where several different ConWIP loops need to be coordinated (see e.g. Huang *et al*., 2016). In a single shop, the loop structure itself cannot be changed without creating a different card-based control system altogether. Similar, while there has been research aimed at dynamically adapting the number of ConWIP cards in response to demand (e.g. Renna *et al*., 2013), this research increases the number of cards if demand increases. This not only leads to the well known 'leadtime syndrome' but also runs counter to the original idea that drove the development of ConWIP systems – a low and stable inventory buffer on the shop floor. As a result, there are arguably only two major search directions: (i) to alter the meaning of cards away from controlling jobs; and (ii) to adopt alternative backlog sequencing rules when considering jobs for release. These search directions will be discussed in Section 2.1 and Section 2.2, respectively.

Finally note that this section does not aim to present an exhaustive review of the ConWIP literature; rather, it focuses only on studies that are considered to be relevant to the specific focus of this paper. For a broader review of ConWIP, the reader is referred to Framinan *et al*. (2003) and Prakash & Chin (2015).

2.1 Altering the Meaning of cards

ConWIP uses a single loop to control the input of work to the shop floor. This has two important consequences for the routing characteristics that can be accommodated by ConWIP (Hopp & Spearman, 2001): (i) there should not be too many stations contained in the loop; and, (ii) the routing of jobs should not differ (in other words, lines should not be split). An alternative ConWIP system designed to overcome the latter shortcoming is the m-ConWIP system that makes ConWIP loops product specific (Duenyas, 1994; Framinan *et al*., 2000). Product specific means that the system signals the requirement for a specific component (Riezebos, 2010). In other words, if there are four different types of jobs, then a specific m-ConWIP card is associated with each job type and, as a consequence, four independent m-ConWIP loops exist.

While the switch from job anonymous cards to product specific cards allows routing variability to be accommodated and improves load balancing capability (Germs & Riezebos, 2010), there are at least two weaknesses. First, m-ConWIP does not work in high-variety contexts as jobs cannot typically be grouped into a restricted number of specific job types; as a result, a large number of m-ConWIP cards and associated loops must be maintained, and this leads to the same criticisms as those leveled on the *kanban* system that triggered the development of ConWIP in the first place. Second, m-ConWIP does not address processing time variability. Both ConWIP and m-ConWIP neglect the actual workload contributions of jobs, which hinders effective load balancing if work content varies.

2.2 Backlog Sequencing Rules

One means of realizing load balancing in high-variety contexts is via the backlog sequencing decision (see, e.g. Leu, 2000; Framinan *et al*., 2001), which determines the sequence in which orders are released to the system. Previous studies on the 'backlog pool-sequencing problem' have often focused on complex optimization algorithms (e.g. Woodruff & Spearman, 1992; Herer & Masin, 1997; Golany *et al*. 1999; Framinan *et al*., 2001; Zhang & Chen, 2001; Cao & Chen, 2005). In this body of work, a fixed set of orders has been assumed and the sequence in which those orders should be released by a ConWIP system has been determined to optimize a certain set of performance parameters. However, in a make-to-order system, job arrivals follow a stochastic process and jobs may arrive at any moment in time. In response, Thürer *et al*. (2017a) assessed the impact of a simple greedy heuristic, i.e. a simple backlog sequencing rule. Thürer *et al*. (2017a) showed that a capacity slack-based sequencing rule, which averages the capacity slack (i.e. the difference between a target workload and the

actual workload released to a station) across stations in the routing of a job, has the potential to enhance ConWIP's load balancing capability.

Capacity slack-based sequencing is however rather complex and requires a significant amount of feedback from the shop floor. It remains to be established whether there are other simple means of improving load balancing in the context of ConWIP. While the loop structure itself cannot be changed without creating a different card-based control system altogether, the meaning of cards can effectively be changed – and this will be discussed next.

3. Load-based ConWIP: Changing the Meaning of Cards

In this study, we propose that a ConWIP card should be adapted such that it represents a measure of workload rather than a job. This refinement is contextualized in Table 1.

[Take in Table 1]

The potential of this refinement for performance improvement is highlighted by the COBACABANA and POLCA literature. In COBACABANA, a card represents a workload contribution rather than a job (Land, 2009; Thürer *et al*. 2014). Meanwhile, POLCA also recognizes the need to shift the meaning of cards from jobs (the original POLCA system) to a so-called quantum where cards represent a workload (see e.g. Suri, 1998; Riezebos, 2010). In this case, an order may have to acquire more than one POLCA card to be released. Both systems are however more complex than ConWIP (Thürer *et al*., 2016) and require feedback from each station or every routing step. This presents major implementation challenges especially for shops with limited resources that are looking for a simple straightforward solution to shop floor control. We therefore ask:

Can ConWIP performance be improved by associating ConWIP cards with a workload?

Figure 2 illustrates how our refinement can be operationalized in practice. Based on the refinement to COBACABANA proposed in Thürer *et al*. (2014), we invert the meaning of cards. While in the original ConWIP system having a card available at release signals that a job can be released, in our refined ConWIP system a card represents a workload. As a consequence, cards need to be duplicated. One card, the release card, is used to represent the released workload while the second card, the operations card, travels with the order and signals its completion. Once a job is completed, the release card is withdrawn. Release cards are cut to the size of a job's workload contribution. The stack of release cards then represents the workload on the shop floor. A new job can only be released if its workload contribution does not violate the WIP-Cap.

But the question remains – which type of workload measure should be applied? Workload Control is an alternative production control concept that focuses on the workload (see, e.g. Thürer *et al*. (2011) for a review). We therefore refer to Workload Control theory to identify suitable workload measures to embed within our refined version of ConWIP.

[Take in Figure 2]

3.1 Workload Measures to be Associated with ConWIP cards

With the objective of stabilizing the direct load queuing in front of a station, Workload Control typically controls the workload released but not yet completed at a station (the so-called aggregate load). This however requires feedback to the release function after the completion of each operation. In an attempt to reduce feedback requirements, Tatsiopoulos (1993) suggested only feeding back information after the completion of the whole job. This so-called extended aggregate load appears to be similar to the load controlled in a ConWIP system but where the workload rather than the number of jobs in the system is controlled. But Workload Control's extended aggregate load is the shop load of each station (Land & Gaalman, 1996) while ConWIP uses the (total) shop load. In other words, if there are six stations then there are six extended aggregate loads but there is only one shop load.

The extended aggregate load measure was later refined by Oosterman *et al*. (2000), who introduced two corrections; one to the extended and one to the aggregate load. Both corrections recognize that a job's contribution to a station's direct load is limited to only the proportion of time that a job is at the station. First, the corrected extended aggregate load divides the workload contribution of a job by its routing length. In other words, it uses the average of the processing times of a job. Using this average led to improved performance in pure job shops and equivalent performance to the extended load in general flow shops in the context of Workload Control. Second, the corrected aggregate load divides each processing time contribution by its position in the routing of a job. The use of this measure outperformed the extended and the corrected extended load approaches in both the pure job shop and general flow shop. Consequently, four workload measures will be considered in this study as follows:

- *The number of jobs*: this is the original ConWIP system;
- *The shop load*: this is the total workload of all jobs on the shop floor;
- *The shop load corrected by the routing length*: this is the workload of all jobs on the shop floor where the load contribution of each job is divided by its routing length; and,

- *The shop load corrected by the routing position*: this is the workload of all jobs on the shop floor where the load contribution of each job's operation(s) is divided by its routing position.

Controlled simulation experiments will next be used to assess the performance impact of these different workload measures in the context of ConWIP. The following section outlines the simulation model used.

4. Simulation Model

The shop and job characteristics modeled in the simulations are first outlined in Section 4.1. Section 4.2 then details how ConWIP (and our refinement) were modeled, before the dispatching rule for prioritizing jobs on the shop floor is outlined in Section 4.3. Finally, the experimental design is outlined and the measures used to evaluate performance are presented in Section 4.4.

4.1 Overview of Modeled Shop and Job Characteristics

A simulation model of a general flow shop (Oosterman *et al*., 2000) has been implemented using ARENA simulation software. Our model is stochastic, whereby job routings, processing times, inter-arrival times and due dates are stochastic (random) variables. The shop contains six stations, where each station is a single constant capacity resource. The routing length varies uniformly from one to six operations. All stations have an equal probability of being visited and a particular station is required at most once in the routing of a job. The resulting routing vector (i.e. the sequence in which stations are visited) is sorted for the general flow shop so that the routing is directed and there are typical upstream and downstream stations.

Operation processing times follow a truncated 2-Erlang distribution with a maximum of 4 time units and a mean of 1 time unit before truncation. Set-up times are considered part of the operation processing time. Meanwhile, the inter-arrival time of orders follows an exponential distribution with a mean of 0.648, which – based on the number of stations in the routing of an order – deliberately results in a utilization level of 90%. Due dates are set exogenously by adding a random allowance factor, uniformly distributed between 30 and 50 time units, to the job entry time. The minimum value will be sufficient to cover a minimum shop floor throughput time corresponding to the maximum processing time (4 time units) for the maximum number of possible operations (6) plus an arbitrarily set allowance for the waiting or queuing times of 6 time units. These settings have been chosen to facilitate comparisons

with earlier studies on ConWIP (e.g. Thürer *et al*. 2017a). While any individual high-variety shop in practice will differ in many aspects from this stylized environment, it captures the typical shop characteristics of high routing variability, processing time variability, and arrival variability. Finally, Table 2 summarizes the simulated shop and job characteristics.

[Take in Table 2]

4.2 ConWIP

As in previous simulation studies on ConWIP (e.g. Hopp & Spearman, 1991; Bonvik *et al*., 1997; Herer & Masin, 1997; Jodlbauer & Huber, 2008; Muhammad *et al*., 2015), it is assumed that materials are available and all necessary information regarding shop floor routing, processing times, etc. is known upon the arrival of an order at the shop. On arrival, jobs directly enter the backlog and await release.

4.2.1 Backlog Sequencing

Backlog sequencing is a major factor influencing ConWIP performance. Consequently, we need to consider different backlog sequencing rules when assessing the impact of our refinement. In this study four backlog sequencing rules are applied. The choice of rules is based on recent results in Thürer *et al*. (2017a). The four rules can be summarized as follows.

- *First-Come-First-Served (FCFS)*: a simple time-oriented rule that sequences jobs according to their time of arrival in the pool. This rule was used, e.g. by Leu (2000) and Ryan & Vorasayan (2005).
- *Shortest Total Work Content (STWK):* a simple load-oriented rule that sequences jobs according to the sum of all processing times in the routing of an order. This rule was applied, e.g. by Leu (2000).
- *Capacity Slack (CS)*: a capacity slack-based sequencing rule that sequences jobs according to a capacity slack ratio given by Equation (1) below – the lower the capacity slack ratio of job *j* (S_j), the higher the priority of job *j*. The rule integrates three elements into one priority measure: the *workload contribution* of a job (i.e. the processing time of job *j* at operation *i*: p_{ij}); the *load gap,* (i.e. the difference between a pre-established load norm N_s^A and the current aggregate workload W_s^A released to station *s* corresponding to operation *i*: N_s^A - W_s^A); and, the *routing length* (i.e. the number of operations in the routing of job *j*: n_j), which is used to average the ratio between the load contribution and load gap

elements over all operations in the routing of a job. This rule was introduced by Philipoom *et al*. (1993).

$$S_j = \frac{\sum_{i \in R_j} \frac{p_{ij}}{N_s^A - W_s^A}}{n_j} \quad (1)$$

- *Capacity Slack number of jobs direct load (CSjobdir)*: a capacity slack-based sequencing rule that uses the direct load measured in terms of the number of jobs (i.e. the load queuing in front of a station) instead of an aggregate load (which measures the load from release to completion at a station, i.e. direct and indirect load) to calculate the capacity slack S_j^d (Equation 2).

$$S_j^d = \frac{\sum_{i \in R_j} \left(\frac{1}{N_s^d - W_s^d} \right)}{n_j} \quad (2)$$

Meanwhile, ConWIP does not limit the workload measure W_s (or W_s^d) at each station; the workload may exceed the limit N_s (or N_s^d) resulting in a negative priority value. This means that a capacity slack-based rule may prioritize an already overloaded station. Therefore, if the workload of a station is equal to or exceeds the workload norm, that is $N_s - W_s \leq 0$ (or $N_s^d - W_s^d \leq 0$), then the job is positioned at the back of the queue by replacing the components $\left(\frac{p_{ij}}{N_s^A - W_s^A} \right)$ or $\left(\frac{1}{N_s^d - W_s^d} \right)$ related to this station in the priority value S_j (or S_j^d) with *M*, where *M* is a sufficiently large number.

Finally, the pre-established norm limit N_s (or N_s^d) that is used when calculating the priority measure for capacity slack-based pool-sequencing rules is given by the pre-established limit (WIP-Cap) divided by the number of stations on the shop floor (six). The approach adopted to set and measure the WIP-Cap will be outlined next.

4.2.2 Refinement: Introducing a Workload Limit

In this study we change the meaning of cards such that they represent an amount of workload. Four different measures of the workload that is to be controlled or limited by the ConWIP system are considered in our study (see Section 3.1 above): the number of jobs, the shop load, the shop load corrected by the routing length, and the shop load corrected by the routing

position. Six limits are applied if the WIP-Cap is the number of jobs: 30, 35, 40, 45, 50 and an infinite number of cards or jobs allowed. The same WIP-Cap, but in terms of work content, can also be applied for the shop load corrected by the routing length. However, for the shop load and the shop load corrected by the routing position, the limit has to be multiplied by the average work content of jobs.

4.3 Priority Dispatching Rule for the Shop Floor

ConWIP controls the work released to the shop floor; it does not control the flow of work on the shop floor. Instead, the job that should be selected for processing next from the queue in front of a particular station is determined by a shop floor dispatching rule. In this study, the Modified Operation Due Date (MODD) rule (see, e.g. Baker & Kanet, 1983) is used since it was arguably the best performing rule in Thürer *et al*. (2017a). The MODD rule prioritizes jobs according to the lowest priority number, which is given by the maximum of the operation due date δ_{ij} and earliest finish time. In other words, $max(\delta_{ij}, t+p_{ij})$ for an operation with processing time p_{ij}, where t refers to the time when the dispatching decision is taken. The MODD rule shifts between a focus on ODDs to complete jobs on time and a focus on speeding up jobs – through SPT (Shortest Processing Time) effects – during periods of high load, i.e. when multiple jobs exceed their ODD (Land *et al*., 2015).

The calculation of the operation due date δ_{ij} for the i^{th} operation of a job j follows Equation (3) below. The operation due date for the last operation in the routing of a job is equal to the due date δ_j, while the operation due date of each preceding operation is determined by successively subtracting an allowance c from the operation due date of the next operation. This allowance is given by the cumulative moving average of the actually realized operation throughput times at each station (i.e. the average of all occurrences until the current simulation time).

$$\delta_{ij} = \delta_j - (n_j - i) \cdot c \quad i{:}1 \ldots n_j \tag{3}$$

4.4 Experimental Design and Performance Measures

The experimental factors are: the four different backlog sequencing rules; the four different measures of the workload; and the six levels of WIP-Cap. A full factorial design was used with 96 (4*4*6) scenarios, where each scenario was replicated 100 times. All results were collected over 13,000 time units following a warm-up period of 3,000 time units. These parameters allow us to obtain stable results while keeping the simulation run time to a reasonable level.

Three main performance measures are considered in this study as follows: the *total throughput time* – the mean of the completion date minus the pool entry date across jobs; the *percentage tardy* – the percentage of jobs completed after the due date; and the *mean tardiness* – $T_j = \max(0, L_j)$, with L_j being the lateness of job *j* (i.e. the actual delivery date minus the due date of job *j*). In addition, we also measure the average shop floor throughput time as an instrumental performance variable. While the total throughput time includes the time that an order waits before being released, the shop floor throughput time only measures the time after an order is released to the shop floor.

5. Results

Statistical analysis has been conducted by applying an Analysis of Variance (ANOVA); results are summarized in Table 3. All three factors – the workload measure, the backlog sequencing rule, and the level of WIP-Cap – are shown to be significant, as are all two-way interactions and the three-way interaction in terms of total throughput time and mean tardiness.

[Take in Table 3]

The Scheffé multiple-comparison test has been used to further assess the significance of the differences between the outcomes of the different workload measures and backlog sequencing rules, respectively. The results – as presented in Table 4 and Table 5 – indicate significant differences for all considered performance measures, except for using the number of jobs and correcting the shop load by the routing length (i.e. using the average workload across operations in the routing of a job). When comparing these two workload measures, statistically significant differences can only be observed in terms of the percentage tardy. Meanwhile, using the shop load appears to outperform all other workload measures.

[Take in Table 4 & Table 5]

Detailed results are presented in Figure 3a to Figure 3d for FCFS, STWK, CS, and CSjobdir backlog sequencing, respectively. Rather than comparing one specific parameter setting, parameters are varied for each policy and the results presented in the form of performance curves. These performance or operating characteristic curves are an important means of obtaining a 'fair' comparison across different control policies (Olhager & Persson, 2008). The relative positioning of the different curves (where each curve represents one policy) allows the performance of each policy to be compared. The left-hand starting point of

each curve represents the tightest WIP-Cap. The WIP-Cap increases step-wise by moving from left to right, with each data point representing one level of WIP-Cap. Loosening the cap increases the workload level and, as a result, throughput times on the shop floor become longer. On the far right are the results for infinite load norms or no limit. This single point is located to the right of the curves as it leads to the longest throughput times on the shop floor.

[Take in Figure 3]

In terms of the direct impact of our workload measures and their interaction with the backlog sequencing rule, the following can be observed from the results:

- *Direct Impact of the Workload Measure (within Figures)*: Changing the meaning of cards and controlling the shop load rather than the number of jobs leads to significant performance improvements for all performance measures considered in our study. Meanwhile, the use of either correction (dividing by the routing length or routing position) does not lead to any performance improvement compared to simply using the shop load; both measures appear to rely on the use of a limit for each station (as in Workload Control). The shop load can therefore be considered to be the best workload measure to be used within our load-based ConWIP system.
- *Interaction between the Workload Measure and Backlog Sequencing Rule (across Figures)*: The impact of the backlog sequencing rule when the number of jobs is controlled confirms results in Thürer *et al*. (2017a). FCFS is outperformed by STWK in terms of the percentage tardy and both FCFS and STWK are outperformed by capacity slack-based sequencing rules, with CSjobdir leading to the best performance. However, there are significant two-way interactions between the workload measures and backlog sequencing rules. Load balancing improves if the workload of the shop rather than the number of jobs in the system is controlled; and, as a result, total throughput times are reduced. This effect – obtained by changing the meaning of cards – diminishes performance differences between the different backlog sequencing rules. Still, the combination of limiting the shop load (rather than the number of jobs) and using a capacity slack-based backlog sequencing rule leads to the best performance in terms of all three performance measures. It is therefore this combination that should be applied in practice.

6. Conclusions

ConWIP is a simple yet effective means of implementing pull production – jobs are only allowed to enter the shop floor if the number of jobs on the shop floor is below a certain limit

(the WIP- Cap). As a consequence, ConWIP has received much research attention, where a core focus has been on overcoming ConWIP's main shortcoming – a lack of load-balancing capability that hinders its use in high-variety contexts. While a major advantage of ConWIP is its simplicity, this simplicity also limits the opportunities available to improve the concept. There are arguably only two major search directions: (i) to alter the meaning of cards away from controlling jobs; and, (ii) to adopt alternative backlog sequencing rules for considering jobs at release. In this study, we propose that a ConWIP card should be adapted such that it represents a measure of workload rather than a job, and we present a simple, practical solution for implementing this load-based ConWIP system in practice. Using controlled simulations, we asked: *Can ConWIP performance be improved by associating ConWIP cards with a workload?* Our results have demonstrated the positive performance impacts of limiting the shop load instead of the number of jobs in the system. More specifically, limiting the shop load improves load balancing and reduces total throughput times. Further, when considering alternative measures of the load that is controlled, we observed that using a correction to the shop load, as suggested in the Workload Control literature, leads to worse performance than using the shop load. Therefore, prior results from the Workload Control literature are not directly transferable to ConWIP.

6.1 Managerial Implications

Our simulation experiments have demonstrated that limiting the load of all jobs on the shop floor significantly enhances the load balancing capabilities of ConWIP. As a result, significant performance improvements can be obtained when compared to the original ConWIP system in which the number of jobs is controlled. Another means of improving the load balancing capabilities of ConWIP is via the backlog sequencing decision (Thürer *et al*., 2017a). Thus, load balancing can be improved via a workload measure and/or via the backlog sequencing rule; but which improvement(s) to adopt may depend on the degree of simplicity required. Changing the meaning of cards is arguably a simpler solution to introducing a capacity slack-based backlog sequencing rule, which requires regular feedback from the shop floor on job progress. The best performance however is achieved by combining a shop load measure with a capacity slack-based sequencing rule. While our results demonstrate that load-based ConWIP reduces performance differences across the various backlog sequencing rules, the best performance is still achieved by capacity slack-based rules. It is therefore suggested that a gradual approach is adopted. Managers can first implement load-based ConWIP and then later introduce capacity slack-based sequencing if further performance

improvement is desired and the benefits are perceived to outweigh the drawbacks of an increased level of sophistication.

6.2 Limitations and Future Research

The main limitation of our study is the narrow set of environmental and control variables considered. For example, we have only modeled one level of processing time variability. Similarly, only one dispatching rule for controlling the progress of jobs on the shop floor has been evaluated. While these choices are arguably justified by results from prior studies and the need to keep the study focused, future research could extend our research by exploring the performance of ConWIP and its contingency factors in a broader context.

References

Baker, K.R., and Kanet, J.J., 1983, Job shop scheduling with modified operation due-dates, *Journal of Operations Management*, 4, 1, 11-22.

Bonvik, A.M., Couch, C.E. and Gershwin, S.B., 1997, A comparison of production-line control mechanisms, *International Journal of Production Research*, 35, 3, 789-804.

Cao, D. and Chen, M., 2005, A mixed integer programming model for a two line CONWIP-based production and assembly system, *International Journal of Production Economics*, 95, 3, 317-326.

Duenyas, I., 1994, A simple release policy for networks of queues with controllable inputs, *Operations Research*, 42, 1162-1171.

Framinan, J. M., Ruiz-Usano, R., and Leisten, R., 2000, Input control and dispatching rules in a dynamic CONWIP flow-shop, *International Journal of Production Research*, 38, 18, 4589-4598.

Framinan, J. M., Ruiz-Usano, R., and Leisten, R., 2001, Sequencing CONWIP Flow-shops: Analysis and Heuristics, *International Journal of Production Research*, 39, 12, 2735-2749.

Framinan, J.M., Gonzalez, P.L., and Ruiz-Usano, R., 2003, The CONWIP production control system: Review and research issues, *Production Planning & Control*, 14, 3, 255-265.

Germs, R., and Riezebos, J., 2010, Workload balancing capability of pull systems in MTO production, *International Journal of Production Research*, 48, 8, 2345-2360.

Golany, B. Dar-El, E.M., and Zeev, N., 1999, Controlling shop floor operations in a multi-family, multi-cell manufacturing environment through constant work-in-process, *IIE Transactions*, 31, 8, 771-781

Herer Y. T. and Masin M., 1997, Mathematical programming formulation of CONWIP based production lines; and relationships to MRP, *International Journal of Production Research,* 35, 4, 1067-1076.

Hopp, W.J. and Spearman M.L., 1991, Throughput of a constant working process manufacturing line subject to fails, *International Journal of Production Research*, 29, 3, 635- 655.

Hopp, W.J. and Spearman M.L., 2001, *Factory Physics: Foundations of Manufacturing Management*, Irwin/McGraw-Hill.

Hopp, W.J., and Spearman, M.L., 2004, To pull or not to pull: What is the question?, *Manufacturing & Service Operations Management*, 6, 2, 133-148.

Huang, G., Chen, J., Wang, X., and Shi, Y., 2016, An approach of designing CONWIP loop for assembly system in one-of-a-kind production environment, *International Journal of Computer Integrated Manufacturing*, 29, 7, 805-820.

Jodlbauer, H. and Huber, A., 2008, Service-level performance of MRP, kanban, CONWIP and DBR due to parameter stability and environmental robustness, *International Journal of Production Research*, 46, 8, 2179-2195.

Land, M.J., 2009, Cobacabana (control of balance by card-based navigation): A card-based system for job shop control, *International Journal of Production Economics*, 117, 97-103.

Land, M.J., and Gaalman, G.J.C, 1996, Workload control concepts in job shops: A critical assessment, *International Journal of Production Economics*, 46-47, 535-538.

Land, M.J., Stevenson, M., Thürer, M., and Gaalman, G.J.C., 2015, Job Shop Control: In Search of the Key to Delivery Improvements, *International Journal of Production Economics*, 168, 257-266.

Leu, B.Y., 2000, Generating a backlog list for a CONWIP production line: A simulation study, *Production Planning & Control*, 11, 4, 409-418.

Muhammad, N.A., Chin, J.F., Kamarrudin, S., Chik, M.A. and Prakash, J., 2015, Fundamental simulation studies of CONWIP in front-end wafer fabrication, *Journal of Industrial and Production Engineering*, 32, 4, 232-246.

Olhager J., and Persson F., 2008, Using Simulation-Generated Operating Characteristics Curves for Manufacturing Improvement, In: Koch T. (eds) *Lean Business Systems and Beyond. IFIP – The International Federation for Information Processing*, 257, Springer, Boston, MA

Oosterman, B., Land, M.L., and Gaalman, G., 2000, The influence of shop characteristics on workload control, *International Journal of Production Economics*, 68, 1, 107-119.

Plossl, G.W., and Wight, O.W., 1971, Capacity planning and control, *Working paper presented at the APICS International Conference in St.Louis, Missouri*.

Philipoom, P.R., Malhotra, M.K., and Jensen, J.B., 1993, An evaluation of capacity sensitive order review and release procedures in job shops, *Decision Sciences*, 24, 6, 1109-1133.

Prakash, J., and Chin, J.F., 2015, Modified CONWIP systems: a review and classification, *Production Planning & Control*, 26, 4, 296-307.

Renna, P., Magrino, L., and Zaffina, R., 2013, Dynamic card control strategy in pull manufacturing systems, *International Journal of Computer Integrated Manufacturing*, 26, 9, 881-894.

Riezebos, J., 2010, Design of POLCA material control systems, *International Journal of Production Research*, 48, 5, 1455-1477.

Rose, O., 1999, CONLOAD – A New Lot Release Rule for Semiconductor Wafer Fabs, In *Proceedings of the 1999 Winter Simulation Conference*, 850-855.

Ryan, S.M., and Vorasayan, J., 2005, Allocating work in process in a multiple-product CONWIP system with lost sales, *International Journal of Production Research*, 43, 2, 223-246.

Spearman, M.L., Woodruff, D.L., and Hopp, W.J., 1990, CONWIP: a pull alternative to kanban, *International Journal of Production Research*, 28, 5, 879-894.

Suri, R., 1998, *Quick Response Manufacturing: A Companywide Approach to Reducing Lead Times*, Productivity Press, Portland, OR.

Tatsiopoulos, I.P., 1993, Simplified production management software for the small manufacturing firm, *Production Planning & Control*, 4, 1, 17-26.

Thürer, M., Stevenson, M., and Silva, C., 2011, Three decades of workload control research: a systematic review of the literature, *International Journal of Production Research*, 49, 23, 6905-6935.

Thürer, M., Land, M.J., Stevenson, M., 2014, Card-Based Workload Control for Job Shops: Improving COBACABANA, *International Journal of Production Economics*, 147, 180-188.

Thürer, M., Stevenson, M., and Protzman, C.W., 2016, Card-Based Production Control: A Review of the Control Mechanisms Underpinning Kanban, ConWIP, POLCA and COBACABANA Systems, *Production Planning & Control*, 27, 14, 1143-1157.

Thürer, M., Fernandes, N.O., Stevenson, M., and Qu, T., 2017a, On the Backlog-sequencing Decision for Extending the Applicability of ConWIP to High-Variety Contexts: An

Assessment by Simulation, *International Journal of Production Research*, 55, 16, 4695-4711.

Thürer, M., Stevenson, M., Silva, C., and Qu, T., 2017b, Drum-Buffer-Rope and Workload Control in High Variety Flow and Job Shops with Bottlenecks: An Assessment by Simulation, *International Journal of Production Economics*, 188, 116-127.

Woodruff D.L. and Spearman M.L., 1992, Sequencing and batching for two classes of jobs with deadlines and setup times, *Production & Operations Management*, 1, 1, 87-102

Zhang, W., and Chen, M., 2001, A mathematical programming model for production planning using CONWIP, *International Journal of Production Research*, 39, 2723-2734.

Table 1: Refinements in the Context of ConWIP

Refinement to:	Type of Refinement:	Addresses Variability in: Routing	Processing Time	Notes
Meaning of Cards	m-ConWIP	X		Cannot be applied if product variety is large
	Workload (Instead of Number of Jobs)		**X**	**Proposed Refinement**
Backlog Sequencing	Capacity Slack-based Rules	X	X	Rather complex

Table 2: Summary of Simulated Shop and Job Characteristics

Shop Characteristics	Routing Variability	Random routing; directed, no re-entrant flows
	No. of Work Centers	6
	Interchange-ability of Work Centers	No interchange-ability
	Work Center Capacities	All equal
	Work Center Utilization Rate	90%
Job Characteristics	No. of Operations per Job	Discrete Uniform[1, 6]
	Operation Processing Times	Truncated 2–Erlang; (mean = 1; max = 4)
	Due Date Determination Procedure	Due Date = Entry Time + d; d U ~ [30, 50]
	Inter-Arrival Times	Exp. Distribution; mean = 0.648

Table 3: ANOVA Results

	Source of Variance	Sum of Squares	df[1]	Mean Squares	F-Ratio	p-Value
Total Throughput Time	Load Measure (LM)	2266.97	3	755.66	78.19	0.00
	Backlog Rule (BR)	17096.29	3	5698.76	589.71	0.00
	WIP-Cap (Cap)	25546.81	6	4257.80	440.60	0.00
	LM x BR	1876.07	9	208.45	21.57	0.00
	LM x Cap	2062.17	18	114.56	11.86	0.00
	BR x Cap	26616.37	18	1478.69	153.01	0.00
	LM x BR x Cap	2987.98	54	55.33	5.73	0.00
	Error	107151.55	11088	9.66		
Percentage Tardy	Load Measure (LM)	0.46	3	0.15	350.44	0.00
	Backlog Rule (BR)	0.14	3	0.05	109.21	0.00
	WIP-Cap (Cap)	3.34	6	0.56	1279.55	0.00
	LM x BR	0.03	9	0.00	6.86	0.00
	LM x Cap	0.12	18	0.01	14.79	0.00
	BR x Cap	0.05	18	0.00	5.96	0.00
	LM x BR x Cap	0.02	54	0.00	1.01	0.44
	Error	4.82	11088	0.00		
Mean Tardiness	Load Measure (LM)	544.08	3	181.36	33.33	0.00
	Backlog Rule (BR)	10195.03	3	3398.34	624.54	0.00
	WIP-Cap (Cap)	40292.60	6	6715.43	1234.15	0.00
	LM x BR	1331.64	9	147.96	27.19	0.00
	LM x Cap	722.00	18	40.11	7.37	0.00
	BR x Cap	19167.40	18	1064.86	195.70	0.00
	LM x BR x Cap	2338.34	54	43.30	7.96	0.00
	Error	60333.56	11088	5.44		

[1]) degrees of freedom

Table 4: Results for Scheffé Multiple Comparison Procedure: Load Measure

Load Measure (x)	Load Measure (y)	Total Throughput Time		Percentage Tardy		Mean Tardiness	
		lower[1)]	upper	lower	upper	lower	upper
Corrected Routing Position	Shop Load	0.368	0.832	0.005	0.008	0.080	0.429
Corrected Routing Length	Shop Load	0.827	1.292	0.010	0.013	0.336	0.684
Number of Jobs	Shop Load	0.892	1.357	0.016	0.019	0.373	0.722
Corrected Routing Length	Corrected Routing Position	0.227	0.692	0.003	0.007	0.081	0.430
Number of Jobs	Corrected Routing Position	0.292	0.757	0.009	0.012	0.119	0.467
Number of Jobs	Corrected Routing Length	-0.167*	0.297	0.004	0.007	-0.137*	0.212

[1)] 95% confidence interval; * not significant at α=0.05

Table 5: Results for Scheffé Multiple Comparison Procedure: Backlog Sequencing Rule

Backlog Rule (x)	Backlog Rule (y)	Total Throughput Time		Percentage Tardy		Mean Tardiness	
		lower[1)]	upper	lower	upper	lower	Upper
CSjobdir	CS	-0.741	-0.276	0.000	0.003	-0.570	-0.222
FCFS	CS	2.465	2.930	0.008	0.011	1.909	2.258
STWK	CS	0.972	1.436	0.001	0.005	0.737	1.086
FCFS	CSjobdir	2.974	3.438	0.007	0.010	2.306	2.654
STWK	CSjobdir	1.480	1.945	0.000	0.003	1.134	1.482
STWK	FCFS	-1.726	-1.261	-0.008	-0.005	-1.346	-0.998

[1)] 95% confidence interval; * not significant at α=0.05

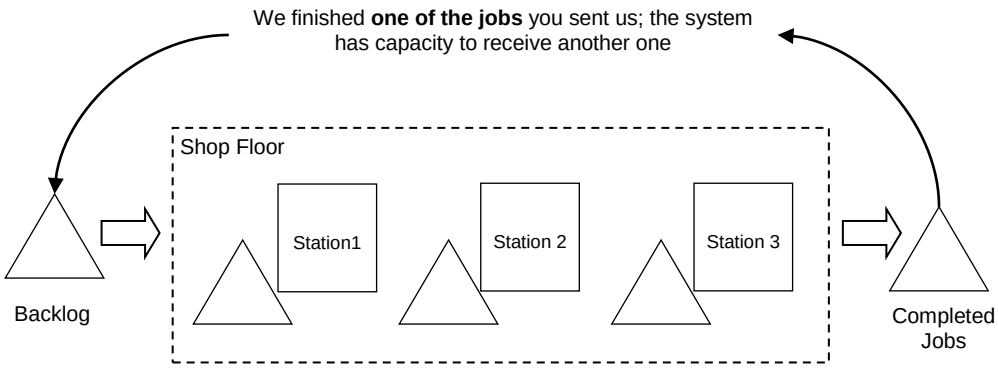


Figure 1: Constant Work-in-Process (ConWIP)

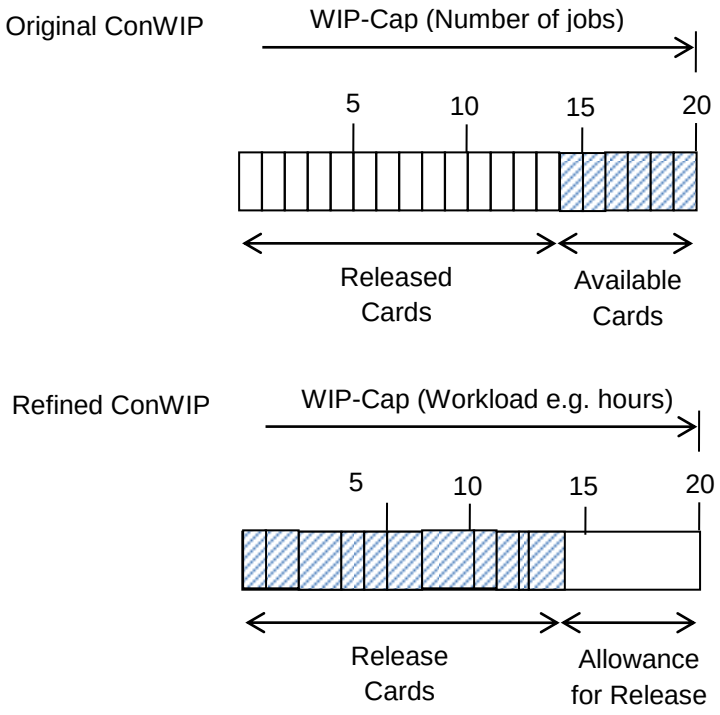
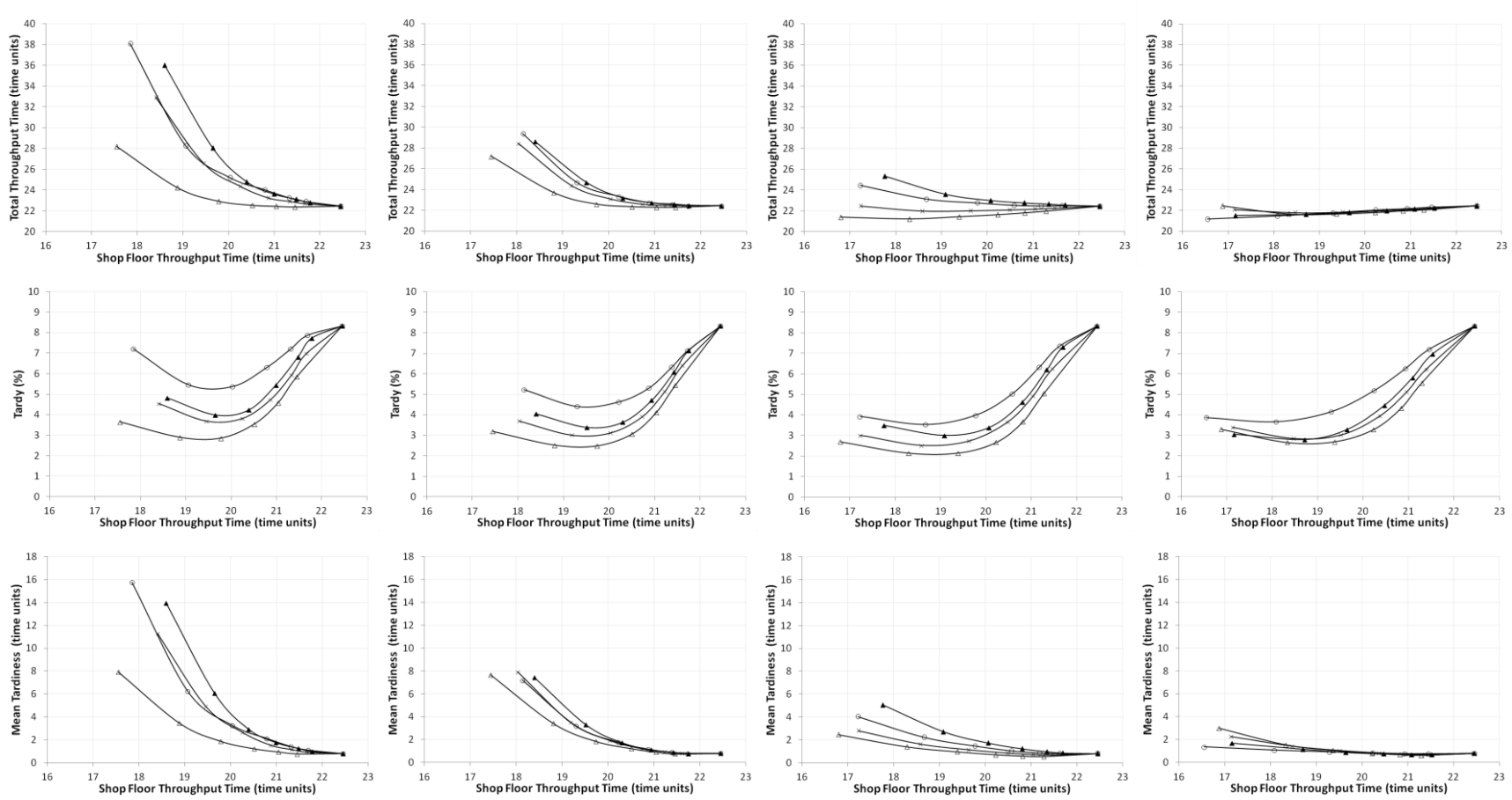


Figure 2: Changing the Meaning of Cards



(a) FCFS (b) STWK (c) CS (d) CSjobdir

Figure 3: Performance Curves for Different Workload Measures and Backlog Sequencing Rules