Card-Based Production Control:

A Review of the Control Mechanisms Underpinning *Kanban*, ConWIP, POLCA and COBACABANA Systems

Matthias Thürer (corresponding author), Mark Stevenson and Charles W. Protzman

Name: Prof. Matthias Thürer
Institution: Jinan University
Address: Jinan University
No 601, Huangpu Road
510632 Guangzhou, PR China
E-mail: matthiasthurer@workloadcontrol.com

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

Name: Charles W. Protzman, MBA, CPM
Institution: BIG – Business Improvement Group LLC.
Address: 1403 Midmeadow Road
Towson, MD 21286, USA
Email: charlieprotzman@biglean.com

Keywords: *Card-based Control; Kanban; ConWIP; POLCA; COBACABANA.*

Card-Based Production Control: A Review of the Control Mechanisms Underpinning *Kanban*, ConWIP, POLCA and COBACABANA Systems

Abstract

Since the emergence of *Kanban*, there has been much research into card-based control systems. This has included attempts to improve *kanban* and/or develop alternative systems, particularly ConWIP (i.e. Constant Work-In-Process), POLCA (i.e. Paired-cell Overlapping Loops of Cards with Authorization), and COBACABANA (i.e. Control of Balance by Card-Based Navigation). Yet, to date, no unifying review of the mechanisms underpinning these systems has been presented. As a consequence, managers are not provided with sufficient support for choosing an appropriate system for their shop; and researchers lack a clear picture of how the mechanisms compare, leading to several misconceptions. This paper reviews the control mechanisms underpinning the *kanban*, ConWIP, POLCA and COBACABANA systems. By comparing the "control mechanism" (i.e. the loop structure and card properties) and "contextual factors" (i.e. routing variability, processing time variability, and whether stations are decoupled by inventory or the flow of jobs is controlled), we provide managers with guidance on which system to choose. For research, we show for example that most criticisms put forward against *kanban* systems, e.g. to justify the development of ConWIP, POLCA, or COBACABANA, only apply to work-in-process *kanban* systems and not to production *kanban* systems. Future research directions for each control system are outlined.

Keywords: *Card-based Control; Kanban; ConWIP; POLCA; COBACABANA.*

1. Introduction

This study investigates card-based control systems. All card-based control systems use information on output from the system to control input to the system – so they are input/output control systems (Wight, 1970; Plossl & Wight, 1971). This cybernetic control cycle also makes them pull systems. Pull systems are here defined in accordance with Hopp & Spearman (2004) as control systems that explicitly limit the work-in-process that can be in the system. The information on output that is used to control input to the system is usually provided via physical entities, e.g. cards; hence the name "card-based control systems".

Card-based control systems provide a simple, visual approach to controlling production. Yet, although they are widely applied in practice (e.g. White *et al*., 1999; White & Prybutok, 2001; Slomp *et al*., 2009; Krishnamurthy & Suri, 2009; Riezebos, 2010), their underlying control mechanisms remain poorly understood. This is an important short-coming both in practical and theoretical terms. First, practitioners may find it hard to make the right choice of system for their particular shop; and, second, not comparing the different systems may introduce misconceptions into the literature that hinder further theory development. In response, this paper presents a critical review on the control mechanisms underpinning four key card-based control systems: *kanban* (e.g. Sugimuri *et al*., 1977; Shingo, 1989), Constant Work-in-Process (ConWIP; e.g. Spearman *et al*., 1990; Hopp & Spearman, 2001), Paired-cell Overlapping Loops of Cards with Authorization (POLCA; e.g. Suri, 1998; Riezebos, 2010), and Control of Balance by Card-Based Navigation (COBACABANA; e.g. Land, 2009; Thürer *et al*. 2014). These four key systems have been chosen since, to the best of our knowledge, they build the foundations for all card-based control systems available in the literature to date.

By reviewing the control mechanisms underpinning our four card-based control systems, we reveal the loop structure (i.e. how cards circulate) and card properties (i.e. what information cards convey) of each system. Taking these characteristics into account, we discuss implications for applicability in terms of three contextual factors: (i) routing variability (e.g. variability in the sequence in which stations need to be visited); (ii) processing time variability; and, (iii) whether stations are decoupled by inventory (a so-called inventory control problem) or whether the flow of individual jobs needs to be controlled (a so-called order control problem). The first two factors represent job characteristics while the third factor represents a management decision often dependent on the degree of customization. We do not consider a related contextual factor that is typically used in the literature – whether a system produces to-stock or to-order. This is sometimes used as a proxy for customization but, in our understanding, this factor only determines when an order is placed (i.e. whether the placement precedes production or *vice versa*). The major determining factor in our study is instead whether it is inventory or the flow of (often highly customized) orders that is controlled. We argue that these three factors have the greatest impact on which card-based control system should be chosen for a given context (Thürer *et al*., 2016). Providing managers with guidance on which system to choose for their shop represents the first objective of our paper.

The second objective of our paper is to dispel misconceptions that may hinder the further theoretical or conceptual development of card-based control systems in the literature. In other words, we highlight discrepancies between how the card-based control systems actually work and the beliefs often held in the literature. For example, *kanban* is typically viewed as an inventory replenishment system, often operationalized by circulating *kanban* containers (e.g. Spearman *et al*., 1990; Berkley, 1992; Graves *et al*., 1995; Lage Junior & Godinho Filho, 2010; Riezebos, 2010; Suri, 2010; Gonzalez *et al*., 2012). But we will show that *kanban* was originally described by Ohno (1988) as a system that allows for perfect synchronization at maximum customization. It is argued that dispelling these types of misconceptions is an important step towards clarifying existing work and providing a direction for future research.

Our four different card-based control systems will be reviewed next in chronological order of emergence: *kanban* systems in Section 2, ConWIP in Section 3, POLCA in Section 4, and COBACABANA in Section 5. Note that our study is different from a 'classical' literature review that seeks to systematically scan and present the literature. For this type of literature review, the reader is referred, to, e.g. Berkley (1992), Framinan *et al*. (2003), Lage Junior & Godinho Filho (2010), and Gonzalez *et al*. (2012). While we did systematically scan the literature, a paper will only be presented if it is of relevance to the argument. Previous reviews have focused on one type of card-based control system while we examine and compare four different approaches. Each review section first explores the mechanisms underlying the respective system, before the related literature is discussed and managerial implications are outlined. A discussion summarizing the key managerial implications is then provided in Section 6. Final conclusions are presented in Section 7, which includes the most important future research directions.

2. *Kanban* Systems

Taiichi Ohno introduced *kanban* systems at Toyota in the 1960s. But there was not one universal type of *kanban* system – there were several different *kanban* systems. How the different *kanban* systems work will first be explored in Section 2.1. In the light of the control mechanisms underpinning *kanban* systems that are revealed, Section 2.2 then discusses the *kanban* literature before Section 2.3 provides managerial implications.

2.1 Mechanisms Underlying *Kanban* Systems

Kanban systems were originally developed to co-ordinate the confluence of different product flows. This may either be within a company (the internal supply chain) or across different companies (the external supply chain). For example, lines producing parts that are linked to other lines producing sub-assemblies that are linked to a final assembly line. We use the term 'line' here, but this may in fact be any resource type or set of resources, e.g. cell, work centre, shop floor, supplier, etc. There is typically an inventory decoupling point between each line; and this is often called a "supermarket", most likely due to Ohno's statement that: "*From the supermarket we got the idea of viewing the earlier process in a production line as a kind of store. The later process goes to the earlier process (supermarket) to acquire the required parts (commodities) at the time and in the quantity needed*" (Ohno, 1988; p 26).

When we described a confluent product flow above, we moved from part to subassembly to final assembly following the material flow. But a *kanban* signals or transmits information in the opposite direction. The first of a set of *kanban* rules in Ohno (1988) states that the latter line goes to the earlier line to pick up the products it needs. The second *kanban* rule then states that the earlier line produces certain items in the quantity indicated by the *kanban* card. This creates the loop structure underlying all *kanban* systems. The third and fourth *kanban* rules presented in Ohno (1988, p 30) – "*No items are made or transported without a kanban*" and "*Always attach a kanban to the goods*" – link *kanban* cards to work-in-process. Therefore, controlling the number of cards in each loop controls the work-in-process in the loop.

Section 2.1.1 and 2.1.2 below outline two *kanban* systems for this original environment of confluent product flows: the work-in-process *kanban* system and the production *kanban* system. Section 2.1.3 then discusses the implications that arise from using *kanban* systems to co-ordinate the flow of independent product flows through a set of capacity resources and outlines the common *kanban* system. There are terms given to many *kanban* systems in the literature, with overlapping definitions. Here, we follow the names given to the *kanban* (card) types in the seminal work of Monden (1983), Ohno (1988), and Shingo (1989). Note that we argue that it is not so important what name is given to a system but how its underlying control mechanism can be defined.

2.1.1 Mechanisms Underlying a Work-in-Process Kanban System

A work-in-process *kanban* system is illustrated in Figure 1. It uses two card loops that can be described as follows:

(i) The later or receiving line uses a product from the earlier or feeding line (e.g. assembly uses a subassembly), which frees up a withdrawal *kanban*.

(ii) The withdrawal *kanban* is then taken to the supermarket. Each product in the supermarket has a work-in-process *kanban* attached to it, which identifies the product.

(iii) A product with a work-in-process *kanban* corresponding to the withdrawal *kanban* is taken, the work-in-process *kanban* is detached, the withdrawal *kanban* is attached, and the product moves to the later line, closing the withdrawal *kanban* loop.

(iv) The freed work-in-process *kanban* moves to the beginning of the earlier line, which signals that a new product needs to be produced. The work-in-process *kanban* is attached to this new product. The product is then produced and stored in the supermarket. This closes the work-in-process *kanban* loop.

[Take in Figure 1]

The movement of *kanbans* can occur in batches or a card can be connected to an individual container. But the card should be detachable – otherwise, if the card is connected to a container, the withdrawal *kanbans* will become work-in-process *kanbans* (and *vice versa*). This prohibits accurate control of the number of *kanbans* (and thus the level of work-in-process) in the system.

The work-in-process *kanban* system was presented, e.g. in Sugimuri *et al*. (1977) and Shingo (1989), and is essentially a re-order point system. It does not fit in with Ohno's mantra of zero inventory/overproduction since some decoupling inventory is always required. Moreover, while some customization can be achieved by having different products in the supermarket (although at the cost of higher inventory), higher degrees of customization are not economically feasible (Spearman *et al*., 1990; Suri, 2010; Riezebos, 2010). However, Ohno (1988) outlined a quite different, production *kanban* system – as presented in the next subsection – that overcomes these shortcomings.

2.1.2 Mechanisms Underlying a Production Kanban System

A *kanban* system that (theoretically) allows for zero-decoupling inventory (i.e. perfect synchronization between the different processes) and full customization is the so-called production *kanban* system outlined by Ohno (1988, Fig 2, p 49). It is illustrated in Figure 2.

[Take in Figure 2]

While a work-in-process *kanban* signals to the beginning of the earlier line that a product was used, a production *kanban* signals to the beginning of the earlier line that a product will be used. It is consequently sent before the product on the later line (e.g. the assembly line) arrives at the station where the product from the earlier line (e.g. the subassembly line) is needed. If the flow times on the later and earlier lines are synchronized, there is no need for decoupling inventory. Meanwhile, since each production *kanban* can be bound to one individual product, full customization is possible.

2.1.3 Mechanisms Underlying a Common Kanban System

Sections 2.1.1 and 2.1.2 described *kanban* systems in the context of confluent product flows. In the production lines described above, the individual stations making up each line are not connected by *kanban* loops; only the beginning of the line receives a signal to start production, triggered because a product at the end of the line was either used (in the case of a work-in-process *kanban*) or will be used (in the case of a production *kanban*). But the *kanban* systems most typically presented in the literature are provided for another context – the co-ordination of independent product flows through a series of capacity resources, here referred to as 'stations' (Schonberger, 1983; Spearman *et al*., 1990; Berkley, 1992; Graves *et al*., 1995; Dallery & Liberopolous, 2000; Takahashi, 2003; Lage Junior & Godinho Filho, 2010; Gonzalez *et al*., 2012; Thürer *et al*. 2014, 2015). This is the problem typically encountered within the individual lines above; so we now change the level of analysis from lines to (the co-ordination of) its constituting stations.

If the above two *kanban* systems are applied to the co-ordination of stations on a line, we obtain the so-called dual-*kanban* system described, e.g. in Schonberger (1983), Berkley (1992), and Lage Junior & Godinho Filho (2010). For example, a work-in-process *kanban* system used for co-ordinating two stations is depicted in Figure 3.

[Take in Figure 3]

However, often the supermarket (or output buffer) for Station A is not required. Rather, products completed at Station A move directly to the queue at Station B. This eliminates the need for a withdrawal *kanban*. Since the *kanban* in the single, remaining loop can be considered either a work-in-process or production *kanban,* it is called a common *kanban*. The common *kanban* system is depicted in Figure 4.

[Take in Figure 4]

If the common *kanban* functions as a work-in-process *kanban*, i.e. signaling that a product was used, it is an inventory control system. In other words, the *kanban* system consists of a chain of decoupled common *kanban* loops.

The common *kanban* system can equally be used as an order control system, i.e. to control the flow of individual orders. This occurs when the common *kanban* functions as a production *kanban*. In this case, the common *kanban* signals that a certain product will be used. This information that concerns which specific product will be used and should thus be started by the first station needs to be propagated from the last to the first station. Meanwhile, the *kanban* associated with the order (that signaled the need) has to wait at the station for the order to arrive. Thus, the *kanban*s of a station in an order control problem represent direct load (queuing at the station) and indirect load (still upstream). It follows that the number of *kanban*s allowed in each loop should increase the further downstream a station is positioned.

The above has unraveled the control mechanisms underpinning *kanban* systems. Next, we will explore the resulting implications for existing and future research. This involves a discussion on the general state of the *kanban* literature rather than an in-depth review. For the latter, the reader is referred to Berkley (1992) or Lage Junior & Godinho Filho (2010).

2.2 Discussion of the *Kanban* Literature

Most of the *kanban* literature has focused on the context of co-ordinating independent product flows through a series of capacity resources; and especially on sequential production lines. However, our analysis of the underlying control mechanisms shows that *kanban* systems can easily be applied to all forms of routing characteristics as long as routing variability is low. Similarly, Hopp & Spearman (2001, p 470) stated that *kanban* systems naturally provide a mechanism for sharing a resource among different routings.

The only exception that could be identified from the literature that did explore the use of *kanbans* in job shops, where routing variability is high, was by Gravel & Price (1988) – but, in their work, *kanbans* were used to co-ordinate assembly operations, i.e. confluent product flows. More recently, Harrod & Kanet (2013) used simulation to explore the performance impact of a *kanban* system in the context of an order control problem with undirected routings. However, the *kanban* system the authors used was modified to avoid the problem of indirect loads. *Kanban* control was realized by restricting the number of jobs in the input and output buffers of each individual station. But since there is no signal of need from the end of the process (i.e. the last station in the routing of a job) to the beginning (i.e. the first station in the routing of a job), it cannot be considered a *kanban* (or pull) system according to our definition. Rather, it may be considered a shop floor with limited storage space (see, e.g. Buzacott, 1976; Thürer *et al*., 2013). In the light of this discussion, a first important research question is:

- What is the performance impact of *kanban* systems in shops with varying routing characteristics?

Meanwhile, much of the available literature sought to address one of the main weaknesses of *kanban* systems: sensitivity to processing time variability. The main means of accommodating processing time variability has been to adjust the number of *kanbans* allowed in the system (see, e.g. Takahashi & Nakamura, 1999; Dallery & Liberopolous, 2000; Tardif & Maaseidvaag, 2001; Takahashi, 2003). It has, however, recently been argued in Thürer *et al*. (2015) that an increase in the number of *kanbans* when the workload increases is counter-productive since it leads to the well-known lead-time syndrome (Mather & Plossl, 1978). Instead, some kind of capacity adjustment or workload balancing is required. Yet, while workload balancing and associated capacity adjustments across lines (i.e. *heijunka*) is a key feature of the work-in-process and production *kanban* systems described by Ohno (1988) and Shingo (1989), this technique was largely neglected by the majority of studies that followed in the 1990s and 2000s.

To the best of our knowledge, only one study in the *kanban* literature has focused on load balancing. Driven by the need to accommodate variation in processing times, Gupta & Al-Turki (1997) developed a centralized algorithm that fits the workload contribution of parts (measured in processing time) to the capacity available during a day by adjusting the number of *kanbans*. For example, if parts have a processing time of 0.2 days, then the number of *kanbans* should be equal to 5; if parts have a processing time of 0.5 days, then the number of *kanbans* should be equal to 2; and so on. The general neglect of load balancing considerations leads to another important research question:

- How can workload balancing (or *heijunka*) be realized in common *kanban* systems?

Finally, most of the literature has viewed *kanban* as an inventory control system. For example, the generalized *kanban* system (e.g. Frein *et al*., 1995; Liberopoulos & Dallery, 2000) and the extended *kanban* system (e.g. Dallery & Liberopolous, 2000; Liberopoulos & Dallery, 2000) combine a *kanban* based pull element with a base-stock element that keeps stocks in the system to increase speed of delivery. But our analysis has revealed that *kanban* can be far more than this. Two papers that have explored the implications of using a common *kanban* system for controlling the flow of individual orders (i.e. an order control problem) were presented by Chang & Yih (1994a, 1994b). In their work, the authors presented a so-called generic *kanban* system that worked backwards from station to station with an order having to acquire a *kanban* card for each station in its routing before entering the system. This was justified by the fact that *kanbans* represent a specific order, and production cannot start until the order belonging to the *kanban* arrives at a station. Once an operation was completed at a station, the *kanban* was freed and could be acquired by a different order. However, the authors did not recognize that this implies the concept of indirect load. In fact, the indirect load that is represented by *kanbans* explains the main result in Chang & Yih (1994b) – that having a larger number of *kanbans* at a downstream station than at an upstream station leads to better performance.

It should be noted that there are more papers that investigate the performance of a common *kanban* system for controlling the flow of orders rather than for controlling the level of inventory. These include Spearman *et al*. (1990) and Gstettner & Kuhn (1996), where it was also shown that a larger number of *kanbans* at downstream stations leads to better performance. But these papers did not address the consequences of their assumption, i.e. it was not recognized that controlling the flow of individual orders significantly changes the control problem compared to an inventory control problem.

2.3 Managerial Implications: *Kanban* Systems

Kanban systems are powerful means of controlling confluent product flows in any context: the work-in-process *kanban* system as an inventory replenishment system; and the production *kanban* system as a means of achieving flow synchronization. But there are important restrictions on the common *kanban* system used to co-ordinate independent product flows through a series of capacity resources (or stations). By examining our three contextual factors – routing variability, processing time variability, and whether the control problem is an inventory or order control problem – through the lens of the revealed control mechanism, the following can be concluded:

- Routing Variability: Each routing step has to be represented by a *kanban* loop. This means routing variability should be low to avoid a high number of overlapping loops that would otherwise be cumbersome to control.

- Processing Time Variability: *Kanban* systems do not generally incorporate load balancing, which impedes their application when processing time variability is high (e.g. in a job shop environment).
- Inventory *vs.* Order Control: *Kanban* systems are highly effective control mechanisms if stations are decoupled (the inventory control problem). However, there are several problems with their use for order control. For example, the last station, which is used to control the process, requires the largest number of *kanbans;* and *kanbans* remain in each loop, hindering the propagation of specific order information.

3. Constant Work-In-Process (ConWIP)

Mark L. Spearman, Wallace J. Hopp, and David L. Woodruff developed ConWIP as an alternative to *kanban* systems (see, e.g. Spearman *et al*., 1990, Hopp & Spearman, 2001). A brief description of ConWIP is given next in Section 3.1. In the light of the revealed control mechanisms, the ConWIP literature is then discussed in Section 3.2 before managerial implications are provided in Section 3.3.

3.1 Mechanisms Underlying a ConWIP System

ConWIP is arguably the simplest card-based control system of those considered in this paper. Whenever the number of jobs in the line is below a pre-established limit, a new job is started by the first station. The completion of a job is signaled by the last to the first station in the line using a card. A ConWIP system is illustrated in Figure 5; and from this illustration, it becomes clear that the ConWIP system is similar to a work-in-process *kanban* loop. There are two differences when compared with a *kanban* system:

- Since ConWIP is applied to a single line and not to link different lines, the signal does not start from a later line but from the last station (the end) of the same line.
- ConWIP uses job-anonymous cards. Therefore, rather than indicating that 'a certain job' can be started, a ConWIP card indicates that 'a job' can be started. This allows ConWIP to shift the decision concerning which job to start next from the end to the beginning of the line.

[Take in Figure 5]

What follows is a discussion on the general state of the ConWIP literature rather than an in-depth review. For a more in-depth review, the reader is referred to Framinan *et al*. (2003) and Prakash & Chin (2015).

3.2 Discussion of the ConWIP Literature

Performance comparisons between a common *kanban* system and ConWIP are somewhat unfair since any evaluation would take place in the context of an order control problem – otherwise, ConWIP simply would not work. This change in the control problem is typically not recognized (e.g. Spearman *et al*., 1990; Gstettner & Kuhn, 1996) yet it is significant. In particular, it requires an increase in the number of *kanban* cards at downstream stations to allow for the indirect load; and by making this adjustment, *kanban* may in fact outperform ConWIP (see results in Gstettner & Kuhn, 1996). Meanwhile, Sato & Khojasteh-Ghamari (2012) argued that *kanban* outperforms ConWIP in more complex production systems, e.g. confluent production lines.

Hopp & Spearman (2001, p. 470) stated that: "… *kanban can be viewed as tandem CONWIP loops carried to the extreme of having only a single machine in each loop. So from a CONWIP enthusiast's perspective, kanban is just a special case of CONWIP*." This assumes that *kanbans* are used to link the individual stations or machines of a process, i.e. the use of a common *kanban* system. In contrast, our analysis reveals that ConWIP can be understood as a work-in-process or production *kanban* loop, as used for co-ordinating the confluence of different product flows. The main difference is that ConWIP cards are job-anonymous. In fact, Liberopoulos & Dallery (2000) had earlier argued that the ConWIP control system is a special case of a single-stage *kanban* control system.

The only justification that could be found for job-anonymous cards is that it overcomes the problems of a work-in-process *kanban* system, where each card is bound to a certain type of job (meaning many different *kanbans* and associated inventories are needed if there are many different types of jobs). However, ConWIP should only be applied if routings are constant and processing times are similar (Hopp & Spearman, 2001, p 461), i.e. in repetitive environments where only a limited set of job types is typically produced, anyway. Meanwhile, the criticism does not hold for a production *kanban* system – the last station could simply signal to the first station which job should be started next. For example, Framinan *et al*. (2000) compared three different ConWIP systems and showed that M-closed systems (where individual card-counts for each job type are maintained) outperform S-closed systems (where a single card-count for all job types is maintained). So an important research question is:

- What is the real advantage of job-anonymous cards (i.e. ConWIP) compared to cards that identify the product (i.e. *kanban* systems)?

Framinan *et al*.'s (2000) work included an S-closed/Minimum(WIP) ConWIP system that is similar to an S-closed system – the difference being that, in the former, a completed job triggers the release of a specific job type (the one with the least work-in-process in the system). This kind of system changes the actual sequence in which jobs are released to the system. Thürer *et al*. (2015) recently demonstrated the potential of this sequencing rule to improve performance. Therefore, another important research question is:

- How can the sequence in which jobs are considered for release be used to improve the performance of ConWIP?

3.3 Managerial Implications: ConWIP

ConWIP is a straightforward solution for controlling the flow of jobs (i.e. the order control problem); however, there are important restrictions on its applicability. By examining our three contextual factors through the lens of the revealed control mechanism, the following can be concluded:

- Routing Variability: Since there is only one loop, all jobs need to enter the shop at the same station and leave the shop at the same station. The flow should also not be split and the number of stations in the loop should not be too long (Hopp & Spearman, 2001) in order to control the level of work-in-process at each station. As a result, ConWIP essentially only applies to a pure flow shop where all jobs visit all stations in the same sequence.
- Processing Time Variability: ConWIP does not apply to shops with high processing time variability since it does not support load balancing (Germs & Riezebos, 2010).
- Inventory *vs*. Order Control: ConWIP does not apply if stations are decoupled (i.e. an inventory control problem) since this would imply a work-in-process limit for each station – and ConWIP only provides a limit for the shop as a whole (in the form of the number of cards/jobs). Therefore, ConWIP only applies to an order control problem.

4. Paired-cell Overlapping Loops of Cards with Authorization (POLCA)

Suri (1998) was the first to present POLCA before it was later extended by authors such as Vandaele *et al*. (2008) and Riezebos (2010). POLCA was argued to be an alternative to *kanban* specifically for the context of Quick Response Manufacturing or time-based competition (Suri, 1998). It is different from the other card-based systems discussed here in the sense that it combines a card-based component with a Material Requirements Planning (MRP) system. It is therefore described as a push/pull system. How this works will be described next in Section 4.1 before the POLCA literature is discussed in Section 4.2 in the light of the revealed underlying control mechanism. Finally, managerial implications are outlined in Section 4.3. While no explicit review paper on the POLCA literature exists, an extensive literature review is provided within the work of Riezebos (2010).

4.1 Mechanisms Underlying a POLCA System

POLCA uses card-loops between station pairs, e.g. between stations A and B. Each pair of stations has a POLCA card. The POLCA literature typically refers to cells, but this is just a question of the level of analysis. To keep the discussion here consistent with the rest of the paper, we will continue to refer to stations. As an example, consider an order that moves from Station A to Station B to Station C. When the order arrives at Station A, three conditions have to be met to start the order:

(i) Station A must be available;
(ii) The earliest release date (which will be described below) for this order at Station A must have been reached; and,
(iii) The POLCA A-B card (which circulates between the station pair A and B) must be available, indicating the future availability (of capacity) at Station B.

If this is the case, the POLCA A-B card is attached to the order and the order is processed at Station A. Then, the order moves to Station B (and the A-B card remains attached to it). Here the same three conditions as above have to be met, substituting Station B for A and Station C for B. When the order is finished at Station B (and only then), the A-B card is freed and moves back to Station A. The order then moves to Station C. It is apparent that here the third condition above has to be neglected since there is no future station (a problem recognized in, e.g. Vandaele *et al*., 2008).

Taking a closer look, we see that the above POLCA loop system is equivalent to a *kanban* system. This is often hidden since the workings of a *kanban* system are typically described from the end of the line (following the information ‘flow’) while the POLCA literature describes the workings of POLCA from the beginning of the line (following the material ‘flow’). That both are equivalent can easily be seen by describing the same process as above but executed by a common *kanban* system. When the order arrives at Station A, three conditions need to be met:

(i) Station A must be available;
(ii) The order must have the highest priority amongst the orders in the queue; and,

(iii)A *kanban* card from Station B (which circulates between the station pair A and B) must be available, indicating the need for work at Station B.

If this is the case, the *kanban* card circulating between stations A and B (*kanban* A-B) is attached to the order and the order is processed at Station A. The order then moves to Station B (and the A-B card remains attached to it). Here, the same three conditions as above have to be met, substituting Station B for A and Station C for B. If the order starts to be processed at Station B, the A-B card is freed and moves back to Station A, and so on.

So there are essentially only two differences between *kanban* and POLCA:

- In POLCA, the order stays with the A-B card, for example, during processing at Station B while, in *kanban*, the A-B card is freed as soon as the order starts being processed at Station B – and this is why P<u>OL</u>CA refers to overlapping loops.
- Unlike *kanban* cards, POLCA cards are job-anonymous (as with ConWIP cards).

The first difference only delays the feedback of information by the processing time (if there is no pre-emption). The second deprives cards of one of their main functions in a *kanban* system – indicating what work to do – and introduces the need for another means of prioritization (similar to ConWIP). This means of prioritization is provided by the MRP system, which calculates earliest release dates for each station. So, there is a need for Authorization – the "A" in POLC<u>A</u>. But this arguably introduces all of the costs and weaknesses of an MRP system. If earliest release dates are too long, there is needless starvation and if earliest release dates are too short, prioritization is jeopardized. Finally, the POLCA system is illustrated in Figure 6.

[Take in Figure 6]

4.2 Discussion of the POLCA Literature

POLCA has remained largely unchanged since its introduction (Riezebos, 2010). One of the few improvements reported has been the introduction of color-coded cards by Pieffers & Riezebos (2006, cited in Riezebos, 2010) – stations are given a specific color, meaning each POLCA card (e.g. the POLCA A-B card) consists of two colors. Meanwhile, Vandaele *et al*. (2008) presented an approach for setting the number of POLCA cards in accordance with expected demand in the context of an electronic POLCA system.

We have shown above that a POLCA system is essentially a *kanban* system with job-anonymous cards, and there already exists a broad literature on determining the number of *kanban* cards (see Section 2.2 above). Therefore, an important research question is:

- What are the implications of the existing literature for determining the number of *kanban* cards in the context of POLCA?

Another important aspect in need of further investigation is the use of an MRP system. While there have been several case studies reported in the literature (e.g. Krishnamurthy & Suri, 2009; Riezebos, 2010), the accuracy of the earliest release dates calculated by MRP remains unclear, which makes it impossible to evaluate the effectiveness of POLCA's Authorization element. Meanwhile, simulation studies considering POLCA, such as by Germs & Riezebos (2010) and Harrod & Kanet (2013), have completely neglected this aspect and only modeled the card-based component. As a result, the actual performance of POLCA has not been fully assessed. To the best of our knowledge, there exists no simulation study assessing the performance of (a complete) POLCA system. This leads to another important question:

- What is the impact of the earliest release date (which puts the "A" into "POLCA") on POLCA performance?

Finally, as early as Schonberger (1983), there have been reports on the use in practice of a combination of *kanban* and MRP for scheduling. So an important question remains:

- How does POLCA compare to other combinations of *kanban* systems and MRP?

4.3 Managerial Implications: POLCA

Our analysis of the underlying control mechanism reveals that POLCA is equivalent to a *kanban* system with job-anonymous cards. This means that largely the same limitations in terms of our three contextual factors apply:

- <u>Routing Variability:</u> Each routing step has to be represented by a POLCA loop. This means routing variability must be low for POLCA to be effective. In addition, POLCA may lead to blocking if there are feedback loops in the routing. This will be discussed in Section 4.3.1 below. Thus, POLCA systems should only be applied to lines (or shops) with simple, directed routings.
- <u>Processing Time Variability:</u> POLCA systems do not incorporate load balancing, which impedes their application if processing time variability is high (Germs & Riezebos, 2010).

- Inventory *vs*. Order Control: By using job-anonymous cards, POLCA's card-based element treats the order control problem as an inventory control problem. The flow of jobs is co-ordinated by the MRP system. While this makes POLCA more applicable to the order control problem than *kanban* systems, it introduces the weaknesses of an MRP system and may lead to blocking, as will be discussed below.

4.3.1 Blocking in the Context of POLCA

The POLCA system may lead to blocking if there are feedback loops in the routing (Lödding *et al*., 2003; Harrod & Kanet, 2013). For example, a station may not be able to start work since it requires a card from another station, which in turns needs a card from this station. This situation is illustrated in Figure 7.

[Take in Figure 7]

Interestingly, Harrod & Kanet (2013) reported on more severe blocking for *kanban* systems. But it is argued here that this was due to the way in which *kanban* was modelled (see our discussion in Section 2.2 above). *Kanban* systems should not lead to blocking since: in an inventory control problem, card loops are decoupled so jobs have to be interchangeable; and, in an order control problem, information needs to be propagated via each station, which means there is a *kanban* card assigned to the order (representing the indirect load) waiting at each station. In this case, premature station idleness, as reported in Kanet (1988) and Land & Gaalman (1998), may occur if the number of *kanban* cards allowed is not appropriate. But blocking in the form described here for POLCA should not occur. It is the specific characteristics of the POLCA system – that treats the order control problem as an inventory control problem by using job-anonymous cards (e.g. no POLCA cards for the indirect workload are needed) – that leads to the blocking described.

A solution to this blocking behavior suggested by Harrod & Kanet (2013) is the use of so-called 'safety cards' that are available to any job. However, a POLCA system can only function appropriately when the number of safety cards is limited (Riezebos, 2010). Hence, a further research question emerges:

- What is the appropriate number of safety cards to strike the best balance between the risk of blocking and an increase in work-in-process?

5. Control of Balance by Card-Based Navigation (COBACABANA)

Land (2009) developed COBACABANA as a card-based version of Workload Control. Since Workload Control is a production control concept specifically designed for job shops, COBACABANA is a unique card-based solution for job shop control. A description of COBACABANA will be given next in Section 5.1. This follows the refinements to Land's (2009) COBACABANA concept proposed in Thürer *et al*. (2014). The COBACABANA literature is then discussed in Section 5.2 before managerial implications are provided in Section 5.3. Finally, note that although COBACABANA can be extended to incorporate a means for estimating delivery time allowances, in this study we focus on its shop floor control mechanism only – this enables comparison with the other card-based control systems under consideration.

5.1 Mechanisms Underlying a COBACABANA System

COBACABANA is different from the other systems described in this paper in that it uses a centralized release function that precedes the line. Jobs are not released directly to the line but have to wait in a so-called pre-shop pool from which they are released. This release function is used to stabilize work-in-process across stations while meeting other performance targets, such as due date adherence. COBACABANA uses a pair of cards for each operation: one release card, which stays with the central planner and is used for workload calculations, supporting the release decision; and one operation card, which moves with the order and signals to the central planner when an operation is complete. The size of the cards represents the workload of the operation. This workload is measured in corrected workload terms, i.e. the operation processing times are divided by the station's position in a job's routing (Oosterman *et al*., 2000). This recognizes that the operation card for the second operation stays on the shop floor about twice as long as the operation card for the first operation. Thus, rather than increasing the limit on the work-in-process at a downstream station to account for the indirect load, the load contribution itself is corrected. This is important in job shops where station positions are not fixed, meaning a station may be at one time downstream and at another time upstream in the mix of jobs currently released to the shop floor.

When orders are considered for release, they are first sequenced according to a priority measure, e.g. sorted according to earliest due date. Starting with the first order in the sequence, the planner places the release card that corresponds to the corrected workload of the order at each station in its routing in each station's area on a planning board (as depicted in Figure 8). The planner then compares the station workloads with predetermined workload limits or norms. If, for any station in the routing of an order, the workload represented by the release cards on the planning board exceeds 100% of the workload limit, the order is retained in the pool and the order's release cards are removed from the planning board. Otherwise, the order's release cards remain on the planning board, the planner attaches the corresponding operation cards to an order guidance form that travels with an order through the shop, and the order is released. This process continues until there are no unexamined orders in the order pool. The shop floor returns each operation card to the planner as soon as the operation is completed – this closes the information loop and signals to the planner that they can remove the release card that matches the operation card from the planning board.

[Take in Figure 8]

Figure 8 illustrates how the planning board is used when making a release decision. In this example, a new order with two operations is considered for release: one operation at Station B and one operation at Station C. In this example, since both operations can be loaded into their respective stations without exceeding the workload norm, the order is released and its corresponding operation cards are sent to the shop. The stack of release cards in each station's area on the planning board summarizes the corrected workload released to the stations. So the planning board can also be understood as a real-time Yamazumi board. The COBACABANA system is summarized in Figure 9.

[Take in Figure 9]

Finally, the release of orders in COBACABANA may occur periodically or continuously at any moment in time. While the former is recommended in job shops, the latter is likely to be preferred if the routing is directed, such as in a pure flow shop (Thürer *et al.*, 2015). In order to avoid premature station idleness (Kanet, 1988; Land & Gaalman, 1998), periodic release needs to be supported by a (continuous) starvation avoidance trigger, which releases work that can be processed directly at a station immediately when a station is starving regardless of a job's workload contribution (Thürer *et al.*, 2012). This starvation avoidance trigger becomes dysfunctional if there is a gateway station, i.e. a common first routing step across jobs, since direct work can only be injected to this single gateway station.

5.2 Discussion of the COBACABANA Literature

COBACABANA is the most recently developed of the card-based control systems discussed here. But although it was originally developed by Land (2009), it was derived from the more mature Workload Control literature that has now existed for more than 35 years (see, e.g. Thürer *et al*., 2011 for a review). COBACABANA shares significant overlap with the *kanban* system proposed by Chang & Yih (1994a, 1994b). This generic *kanban* system prohibits jobs from entering the system until the required *kanban* card for each station in a job's routing is available. Once the job has been completed at a station, the *kanban* card is freed and another job can acquire this *kanban* card. But quite how this acquisition procedure is supposed to be executed in a general *kanban* system has not been made explicit.

Thürer *et al*. (2014) recently updated COBACABANA in the light of advances in the Workload Control literature by introducing the starvation avoidance trigger referred to above (based on Thürer *et al*., 2012). Further, Land (2009) had used just one set of cards – where the (operation) cards *missing* from the planning board represented the released workload (rather than using explicit release cards). To allow the workload to be represented by the size of the cards, Thürer *et al*. (2014) doubled the number of cards according to function: one card (the release card) to represent the workload; and one card (the operation card) to provide feedback. Using simulation, it was shown that just three card sizes – e.g. for small, medium and large operations – realizes most of the performance benefits of COBACABANA (Thürer *et al*., 2014). This allows processing time estimations to be simplified.

While there exists broad evidence on the potential of COBACABANA to improve the performance of complex shops, this evidence is based on simulation experiments. Although evidence has been provided on the impact of Workload Control in practice (e.g. Bechte, 1994; Wiendahl *et al*., 1992; Hendry *et al*., 2013; Silva *et al*., 2015), empirical evidence for COBACABANA is missing. It therefore remains to be seen whether the performance effects observed in practice for Workload Control can be replicated by its card-based variant, COBACABANA. Thus, the most important research question in the context of COBACABANA is:

- What is the performance impact of COBACABANA in practice?

5.3 Managerial Implications: COBACABANA

COBACABANA was developed independently from the literature on card-based control systems. In other words, it emerged from the separate stream of Workload Control literature. Consequently, it is specifically suited for order control in high variety contexts. In the light of our three contextual factors, the following can be concluded:

- Routing Variability: COBACABANA uses (operation) card loops between each station and a centralized pool that precedes the line. This means that jobs can enter the line at any station and leave the line at any station. Since card loops are also decoupled from the routing characteristics of jobs, all possible routing permutations can be accommodated.

- Processing Time Variability: COBACABANA creates a mix of jobs on the line that balances the workload across stations; this is supported by the release cards on the planning board. It allows processing time variability to be accommodated.
- Inventory *vs*. Order Control: COBACABANA uses a centralized release function, which avoids problems with the propagation of information that is inherent to a *kanban* system in the order control problem.

6. Discussion and Managerial Implications

So far, this study has examined the mechanisms underlying our four different card-based control systems: *kanban*, ConWIP, POLCA, and COBACABANA. This penultimate section contrasts the different systems in terms of the underlying control mechanisms using two dimensions: the loop structure (i.e. how cards circulate) and card properties (i.e. what information cards convey); and it assesses the implications for applicability according to our three environmental factors: (i) routing variability (e.g. variability in the sequence in which stations need to be visited); (ii) processing time variability; and, (iii) whether stations are decoupled by inventory (a so-called inventory control problem) or the flow of individual jobs needs to be controlled (a so-called order control problem).

The characteristics of the loop structure underlying our four card-based systems are summarized in Table 1. Since the work-in-process and production *kanban* systems are designed for confluent product flows, which is different from ConWIP, POLCA and COBACABANA, our focus here for *kanban* is in terms of the common *kanban* system only. The implications for applicability are summarized in Table 2 in terms of the three environmental factors.

[Take in Table 1 & Table 2]

From Table 1 and Table 2, it can be observed that the loop structure prohibits high routing variability for the common *kanban* system, POLCA, and ConWIP systems since all routing steps have to be covered. Since ConWIP only uses a single loop, the single loop should also contain all routing steps. Meanwhile, COBACABANA decouples the routing and loop structure, which allows all possible routing permutations to be accommodated. All loops in COBACABANA contain the centralized release function; and the centralized information allows workload balancing across stations to be realized. Meanwhile, the common *kanban* system and POLCA keep information local, while ConWIP does not provide any information on individual stations, which prohibits effective load balancing calculations. In the order control problem, i.e. when the flow of orders is controlled, the common *kanban* system leads to problems with propagating information from station to station. This is avoided in the POLCA system via the use of job-anonymous cards; jobs are instead co-ordinated by an MRP system. But job anonymous cards lead to blocking in POLCA if there are feedback loops in the routing. This can also be observed from Table 3 and Table 4, which summarize the characteristics of the card properties underlying our four card-based systems and the implications for applicability, respectively.

[Take in Table 3 & Table 4]

Job-anonymous cards, as used in ConWIP and POLCA, lead to the need for another means of prioritization since the cards do not indicate which job should be worked on next. Therefore, these systems typically require some form of higher level planning, such as that provided by an MRP system. Cards in ConWIP and POLCA also do not indicate processing times, which impedes load balancing calculations. In contrast, cards in COBACABANA represent processing times according to their size, which does support load balancing calculations.

6.1 Summary of Applicability of Card-based Control Systems

The main implications for the applicability of the four card-based systems discussed here can be summarized as follows:

- *Kanban*: These systems are widely applied and tested in practice. They are the first choice for controlling confluent product/service flows. They are also a powerful solution for the control of independent product flows if each station is decoupled by decoupling inventory (i.e. inventory control). However, performance is jeopardized in the order control problem (i.e. if the flow of individual orders needs to be controlled) since *kanban* cards now represent direct and indirect workload. In general, routing variability should be low as should processing time variability since support for load balancing is not provided.
- ConWIP: This is a simple, straightforward solution for controlling the flow of individual orders. It is arguably the simplest card-based control system, requiring the fewest parameters to be set. However, it can only be applied in the pure flow shop since it uses a

single loop. It also does not provide support for load balancing, which impedes its use if there is high processing time variability.

- POLCA: This provides a solution that enhances an existing MRP system. It extends the use of a *kanban*-based inventory control system for order control. However, POLCA may introduce blocking if the routing includes feedback loops. Hence, POLCA can only be applied when there are simple, directed routings. It also requires an MRP system, and the earliest release date calculated by the MRP system may introduce starvation. Like *kanban* and ConWIP, POLCA does not provide support for load balancing, which impedes its use when there is processing time variability.
- COBACABANA: This is argued to be the first choice for complex (high routing and/or processing time variability) order control problems. The loop structure allows for all possible routing permutations. Moreover, the centralized planning board gives an overview of the current situation on the shop floor, which supports load balancing. However, it is arguably more complex than the *kanban* and ConWIP systems; and it does not apply to inventory control problems. Finally, while not discussed here, COBACABANA also provides a means for estimating delivery times.

This paper has examined the control mechanisms underpinning different card-based control systems. Using these insights, we have contrasted the different systems in terms of their applicability to different control problems, where the control problem is characterized by three contextual factors. But an important practical aspect is that most shops have more than one control problem. For example, there may be a flow line and a job shop-like cell, which constitutes a job shop that needs to be supplied by raw material. This may then require a nested system, e.g. where a COBACABANA system is used to control a job shop that consists of a series of ConWIP lines and where the supply is controlled by a *kanban* system.

7. Conclusion

Card-based control systems are simple, yet effective means of controlling production. Consequently, they have received much attention in the literature. However, a critical review comparing the control mechanisms underpinning the different systems has not been presented to date. This is a major shortcoming since managers find it hard to choose the right system for their shop and researchers lack a clear picture of how the mechanisms compare – leading to several misconceptions that hinder further progress. This paper set out to address both shortcomings.

Our first objective – to provide guidance on which system to choose in practice – has been addressed by contrasting the control mechanisms underpinning the different card-based systems (i.e. loop structure and card properties) and the contextual factors that characterize specific control problems (routing variability, processing time variability, and whether stations are decoupled by inventory or the flow of jobs is controlled). Our second, objective of dispelling misconceptions that may hinder further progress in the development of card-based control systems has been addressed by examining the literature through the lens of the revealed control mechanisms. For example, *kanban* systems are widely criticized for being part number-specific, meaning a large amount of inventory is required to handle customized products. Yet, we saw that this criticism only applies to work-in-process *kanban* systems while production *kanban* systems allow (at least theoretically) for zero decoupling inventory and full customization. Further, we showed that ConWIP and the card-based component of POLCA are essentially *kanban* systems with job-anonymous cards. This link has been hidden in the literature since *kanban* systems were generally interpreted as linking the stations in a pure flow shop. Moreover, the literature typically switches from an inventory control problem to an order control problem when comparing *kanban* systems with, e.g. ConWIP, without recognizing the implications of this change. Our paper now concludes with a summary of future research directions.

7.1 Limitations and Future Research Directions

A major limitation of our study is our focus on four card-based control systems only, although we argue that these are the key approaches in the literature. Similarly, we have also only considered two dimensions for characterizing the control mechanism and three dimensions for characterizing the contextual environment. These choices were motivated by a need to limit the study to a reasonable breadth that allowed us to go into depth on each approach and dimension. Future research is however invited to enhance our study by including any further systems, any other aspects of the control mechanism, and/or any additional relevant contextual factors. This includes research on customized systems. For example, Gaury *et al*. (2001) obtained promising results towards optimizing card loop structures using simulation.

Another important issue briefly referred to in the discussion of managerial implications was that card-based systems can (and arguably should) be nested. Yet, to the best of our knowledge, the literature has thus far neglected nested systems and focused on individual systems. This may lead to the wrong assumption in practice that only one system is needed –

most shops will face more than one control problem and require an idiosyncratic solution consisting of a nested system. Hence, the question remains:

- How can nested card-based control systems be realized in various contexts, and what is their performance impact?

The future research directions for each card-based system that emerged during our discussion of the literature are summarized in Table 5 together with potential ways of addressing them.

[Take in Table 5]

The final research direction that we propose concerns the use of physical cards. Vandaele *et al*. (2008) used an electronic signal rather than cards, which allowed for extending the capabilities of the original POLCA system. It therefore seems likely that the capabilities of card-based systems could in general be enhanced by embracing new technologies, such as the Internet of Things. A final question could therefore be: how can card-based control systems benefit from concepts such as the Internet of Things?

References

Bechte, W., 1994, Load-oriented manufacturing control just-in-time production for job shops, *Production Planning & Control*, 5, 3, 292 – 307.

Berkley, B.J., 1992; A review of the kanban production control research literature, *Production & Operations Management*, 1, 4, 393-411.

Buzacott, J. A., 1976, The Production Capacity of Job Shops with Limited Storage Space, *International Journal of Production Research*, 14, 5, 597–605.

Chang, T.M., and Yih, Y., 1994a, Generic kanban systems for dynamic environments, *International Journal of Production Research*, 32, 4, 889–902.

Chang, T.-M., and Yih,Y., 1994b, Determining the number of kanbans and lot sizes in a generic kanban system: a simulated annealing approach, *International Journal of Production Research*, 32, 8, 1991–2004.

Dallery, Y., and Liberopoulos, G., 2000, Extended kanban control system: combining kanban and base stock, *IIE Transactions*, 32, 4, 369–386.

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

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.

Frein, Y., Di Mascolo, M., Dallery, Y., 1995, On the design of generalized kanban control systems, *International Journal of Operations and Production Management*, 15, 9, 158 – 184.

Gaury, E.G.A., Kleijnen, J.P.C., and Pierreval, H., 2001, A methodology to customize pull control systems, *Journal of the Operational Research Society*, 52, 7, 789 – 799.

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

Gonzalez, P.L., Framinan, J.M, Pierreval, H., 2012, Token-based pull production control systems: an introductory overview, *Journal of Intelligent Manufacturing*, 23, 5-22.

Gravel, M., and Price, W.L., 1988, Using the Kanban in a job shop environment, *International Journal of Production Research*, 26, 6, 1105-1118.

Graves, R.J., Konopka, J.M., and Milne, R.J., 1995, Literature review of material flow control mechanisms, *Production Planning & Control*, 6, 5, 395 – 403.

Gstettner, S., and Kuhn, H., 1996, Analysis of production control systems kanban and CONWIP, *International Journal of Production Research*, 34, 11, 3253–3273.

Gupta, S.M., and Al-Turki, Y.A.Y, 1997, An algorithm to dynamically adjust the number of Kanbans in stochastic processing times and variable demand environment, *Production Planning & Control*, 8, 2, 133-141.

Harrod, S., and Kanet, J.J., 2013, Applying work flow control in make-to-order shops, *International Journal of Production Economics*, 143, 620-626.

Hendry, L.C., Huang, Y., and Stevenson, M., 2013, Workload control: Successful implementation taking a contingency-based view of production planning & control, *International Journal of Operations & Production Management*, 33, 1, 69-103.

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.

Hopp, W.J., and Spearman, M.L., 2001, *Factory Physics: Foundations of Manufacturing Management*, McGrawHill, 2nd Edition.

Kanet, J.J., 1988, Load-limited order release in job shop scheduling systems, *Journal of Operations Management*, 7, 3, 44-58.

Krishnamurthy, A. and Suri, R., 2009, Planning and Implementing POLCA: a card-based control system for high variety or custom engineered products, *Production Planning & Control*, 20, 7, 596-610.

Lage Junior, M. and Godinho Filho, M., 2010, Variations of the kanban system: Literature review and classification, *International Journal of Production Research*, 125, 13-21.

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., 1998, The performance of workload control concepts in job shops: Improving the release method, *International Journal of Production Economics*, 56-57, 347-364.

Liberopoulos, G., and Dallery, Y., 2000, A unified framework for pull control mechanisms in multi-stage manufacturing systems, *Annals of Operations Research*, 93, 325 – 355.

Lödding, H., Yu, K.-W., and Wiendahl, H.-P., 2003, Decentralized WIP-oriented manufacturing control (DEWIP), *Production Planning & Control*, 14, 1, 42-54.

Mather, H., and Plossl, G.W., 1978, Priority fixation versus throughput planning, *Production & Inventory Management Journal*, 19, 27–51.

Monden, Y., 1983, *Toyota Production System: Practical Approach to Production Management*, Industrial Engineering and Management Press, Norcross, Georgia.

Ohno, T., 1988, *Toyota Production System: Beyond Large-Scale Production*, 1st Ed., Productivity Press.

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*.

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

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

Sato, R., and Khojasteh-Ghamari, Y., 2012, An integrated framework for card-based production control systems, Journal of Intelligent Manufacturing, 23, 717 – 731.

Schonberger, R.J, 1983, Application of Single-Card and Dual-Card kanban, *Interfaces*, 13, 4, 56-67.

Shingo, S., 1989, *A Study of the Toyota Production System from an Industrial Engineering Viewpoint*, Cambridge, MA: Productivity Press.

Silva, C., Stevenson, M., and Thürer, M., 2015, A Case Study of the Successful Implementation of Workload Control: A Practitioner-led Approach, *Journal of Manufacturing Technology Management*, 26, 2, 280-296.

Slomp, J., Bokhorst, J. A. C., and Germs, R., 2009, A lean production control system for high-variety/low-volume environments: a case study implementation, *Production Planning & Control, 20*, 7, 586-595.

Spearman, M.L., Hopp, W.J., and Woodruff, D.L., 1989, An hierarchical control architecture for Constant Work-in-Process (CONWIP) Production Systems, *Journal of Manufacturing & Operations Management*, 2, 147-171.

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.

Sugimori, Y., Kusunoki, K., Cho., F., and Uchikawa, S., 1977, Toyota production system and Kanban system Materialization of just-in-time and respect-for-human system, *International Journal of Production Research*, 15, 6, 553-564.

Suri, R., 1998, *Quick Response Manufacturing: A companywide approach to reducing leadtimes*, Productivity Press.

Suri, R., 2010, *It's about time: The competitive advantage of quick response manufacturing*, Productivity Press.

Takahashi, K., 2003, Comparing reactive kanban systems, *International Journal of Production Research*, 41, 18, 4317–4337.

Takahashi, K., and Nakamura, N., 1999, Reacting JIT ordering systems to the unstable changes in demand, *International Journal of Production Research*, 37, 10, 2293–2313.

Tardif, V., and Maaseidvaag, L., 2001, An adaptive approach to controlling kanban systems, *European Journal of Operational Research*, 132, 2, 411–424.

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., Stevenson, M., Silva, C., Land, M.J., and Fredendall, L.D., 2012, Workload control (WLC) and order release: A lean solution for make-to-order companies, *Production & Operations Management*, 21, 5, 939-953.

Thürer, M., Godinho Filho, M., and Stevenson, M., 2013, Coping with finite storage space in job shops through order release control: an assessment by simulation, *International Journal of Computer Integrated Manufacturing*, 26, 9, 830-838.

Thürer, M., Land, M.J., and 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., 2015, COBACABANA (Control of Balance by Card Based Navigation): An Alternative to Kanban in the Pure Flow Shop?, *International Journal of Production Economics*, 166, 143-151.

Thürer, M., Land, M.J., Stevenson, M., Fredendall, L.D., and Godinho Filho, M., 2015, Concerning Workload Control and Order Release: The Pre-Shop Pool Sequencing Decision, *Production & Operations Management*, 24, 7, 1179-1192.

Thürer, M., Stevenson, M., and Protzman, C.W., 2016; *Card-based Control Systems for a Lean Work Design: The Fundamentals of Kanban, ConWIP, POLCA and COBACABANA*, Productivity Press, ISBN 978-1498746946.

Vandaele, N., Van Nieuwenhuyse, I., Claerhout, D., and Cremmery, R., 2008, Load-Based POLCA: An Integrated Material Control System for Multiproduct, Multimachine Job Shops, *Manufacturing & Service Operations Management*, 10, 2, 181-197.

Wight, O.W., 1970, Input/Output control a real handle on lead time, *Production & Inventory Management Journal*, 11, 3, 9-31.

White, R.E., and Prybutok, V., 2001, The relationship between JIT practices and type of production system, Omega, 29, 113-124.

White, R.E., Pearson, J.N., and Wilson, J.R., 1999, JIT manufacturing: A survey of implementation in small and large US manufacturers, *Management Science*, 45, 1, 1-15.

Wiendahl, H.P., Gläßner, J., and Petermann, D., 1992, Application of load-oriented manufacturing control in industry, *Production Planning & Control*, 3, 2, 118 – 129.

Table 1: The Loop Structure of the (Common) Kanban, ConWIP, POLCA and COBACABANA Systems

	Common *Kanban*	ConWIP	POLCA	COBACABANA
Where established?	Between two stations.	Between entry and exit stations.	Between two stations.	Between stations and a central release function that precedes the shop floor.
Relation to the routing	Needs to be established for each possible routing step.	One single loop must contain all possible routings.	Needs to be established for each possible routing step.	Routing Independent (i.e. not related to the routing).
Contains (Operations per Order)	One operation	All operations	Two operations (an operation forms part of two loops for all except the first and last operation).	One operation
WIP-Cap (Limit on work in the loop)	Per station	On the shop floor load (the load at a single station is not limited).	Per station	Per station

Table 2: Consequences for the Applicability of the Loop Structure

	Common *Kanban*	ConWIP	POLCA	COBACABANA
Consequences: Routing Variability	Only allows for simple, directed routings.	Only allows for the Pure Flow Shop, i.e. where all work visits all stations in the same order.	Only allows for simple, directed routings. Leads to blocking if the loop structure is undirected.	Allows for all possible routing characteristics.
Consequences: Processing Time Variability	Individual loops keep processing time information local. Does not allow for load balancing across stations.	General loop does not provide processing time information. Does not allow for load balancing across stations.	Individual loops keep processing time information local. Does not allow for load balancing across stations.	Centralized information provides a global view of the shop floor, which facilitates load balancing across stations.
Consequences: Inventory *vs*. Order Control	Creates a problem of card propagation in the order control problem since information has to be transmitted for each routing step. This creates direct/indirect load in each loop and prohibits control in an order control problem.	Does not allow for controlling the work-in-process at each station, so should not be applied to an inventory control problem.	Similar structure to *kanban* but problems resolved by card properties. Allows for inventory and order control problems.	Uses a centralized release function to control the mix of orders released to the shop floor. Designed for the order control problem.

Table 3: Card Properties of the Common Kanban, ConWIP, POLCA and COBACABANA Systems

	Common *Kanban*	ConWIP	POLCA	COBACABANA
What does it say?	A part/product was or will be used.	We finished one of the jobs in the system, release another job.	We finished one of the jobs you sent us; you can send us another.	The operation belonging to this part/product at this station has been completed.
Card Type(s)	Originally, three (in the internal supply chain): Withdrawal *kanbans*; Work-in-Process *kanbans* (was used) and Production *kanbans* (will be used); For shop floor control, often reduced to one common *Kanban.*	Only one	Only one	Two (which appear in pairs): A release card for load balancing calculations, and an operation card for feedback.
Information Transmitted	Which part/product was or will be used and should thus be produced. This may include information on the processing time, due date etc.	That the shop floor has capacity to work on another job.	That the next station in the routing of the job has future capacity availability.	For the Operation Card: Which job has been completed at which station. For the Operation/Release Cards: The processing time of this operation (given by the size of the cards).

Table 4: The Consequences of Card Properties for Applicability

	Common *Kanban*	ConWIP	POLCA	COBACABANA
Consequences: Routing Variability	None	None	Prohibits feedback loops due to the risk of blocking.	None
Consequences: Processing Time Variability	Only gives information on jobs that were or will be used at a station. Does not allow for load balancing.	Only gives information on jobs completed by the system. Does not allow for load balancing.	Only gives information on jobs completed at a station. Does not allow for load balancing.	Release cards allow for visualizing the current load situation and job progress on the shop floor. Allows for load balancing. Load balancing calculations are facilitated by the planning board and the release cards.
Consequences: Inventory *vs*. Order Control	If cards are bound to a specific order (order control problem) they have to wait at a station until all preceding operations have been completed (indirect load). This prohibits *kanban*'s use for order control problems.	Jobs are not prioritized since cards are job-anonymous. Requires higher level IT support for creating an appropriate sequence in which jobs are released to the shop floor.	Cards are job-anonymous, which avoids the problems of *kanbans*. Requires an MRP system for prioritizing jobs according to urgency (an earliest release date for each operation).	The centralized release function avoids the problems of *kanban* and ensures prioritizing of jobs.

Table 5: Summary of Key Research Questions for Each Card-based System

Card-based System	Research Questions	How to Address?
Kanban	What is the performance impact of a ***kanban*** system in shops with varying routing characteristics?	Discrete event simulation can be used to model the different routing characteristics.
	How can workload balancing (or *heijunka*) be realized in a common ***kanban*** system?	Conceptual development is required. This can then be tested either through simulation or in practice.
ConWIP	What is the real advantage of job-anonymous cards (i.e. **ConWIP**) compared to cards that identify the product (i.e. ***kanban*** systems)?	Different versions of ConWIP could be developed and compared analytically or using simulation.
	How can the sequence in which jobs are considered for release be used to improve the performance of **ConWIP**?	Different sequencing rules could be designed and compared analytically or using simulation.
POLCA	What are the implications of the existing literature for determining the number of ***kanban*** cards in the context of **POLCA**?	A literature review could be conducted, existing methods refined, and new methods developed. Approaches can then be evaluated using simulation or in practice.
	What is the impact of the earliest release date (the "A" in POLCA) on **POLCA** performance?	Existing case studies could be revisited. New case studies could be conducted. The impact could also be assessed by means of discrete event simulation.
	How does **POLCA** compare to other combinations of ***kanban*** systems and MRP?	A meta-analysis based on existing case studies could be conducted or discrete event simulation used to compare different systems.
	What is the appropriate number of safety cards in a **POLCA** system to strike the best balance between the risk of blocking and an increase in work-in-process?	This can be assessed experimentally using simulation and/or analytically.
COBACABANA	What is the performance impact of **COBACABANA** in practice?	Case study research is required.
Nested Systems	How can nested card-based control systems be realized in various contexts, and what is their performance impact?	Conceptual development is required. This can then be tested either through simulation or in practice.

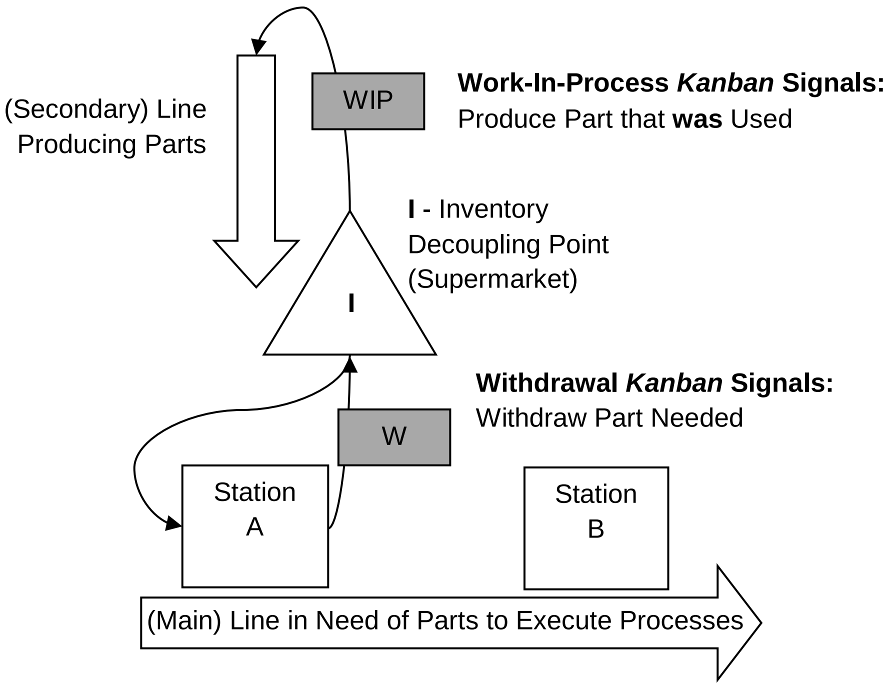

Figure 1: A Work-in-Process Kanban System for the Internal Supply Chain (Linking Product/Service Flows)

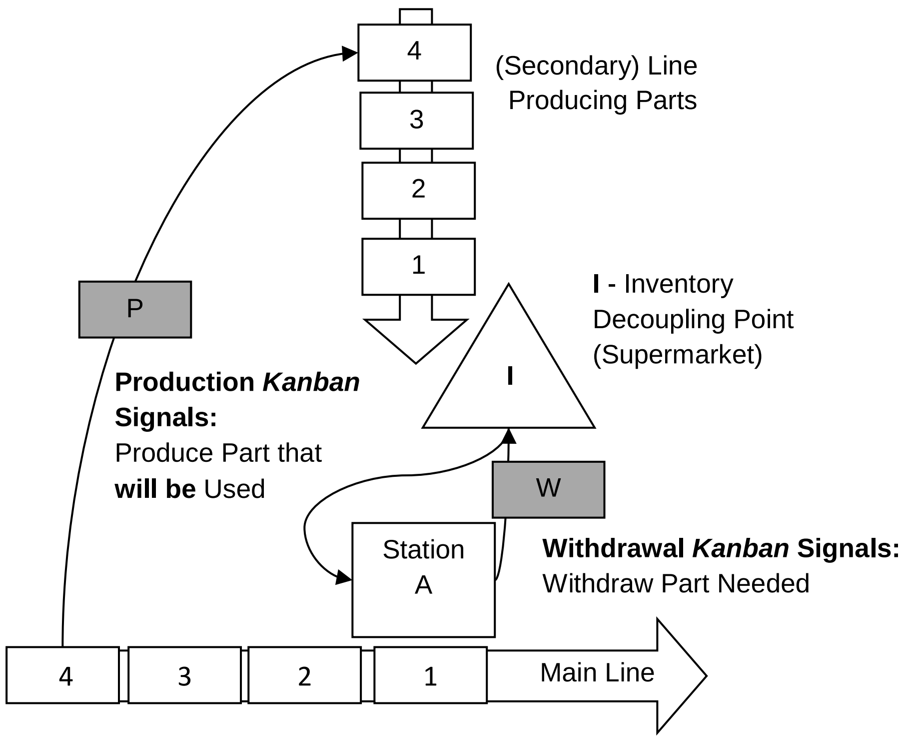

Figure 2: A Production Kanban System for the Internal Supply Chain (Linking Product/Service Flows)

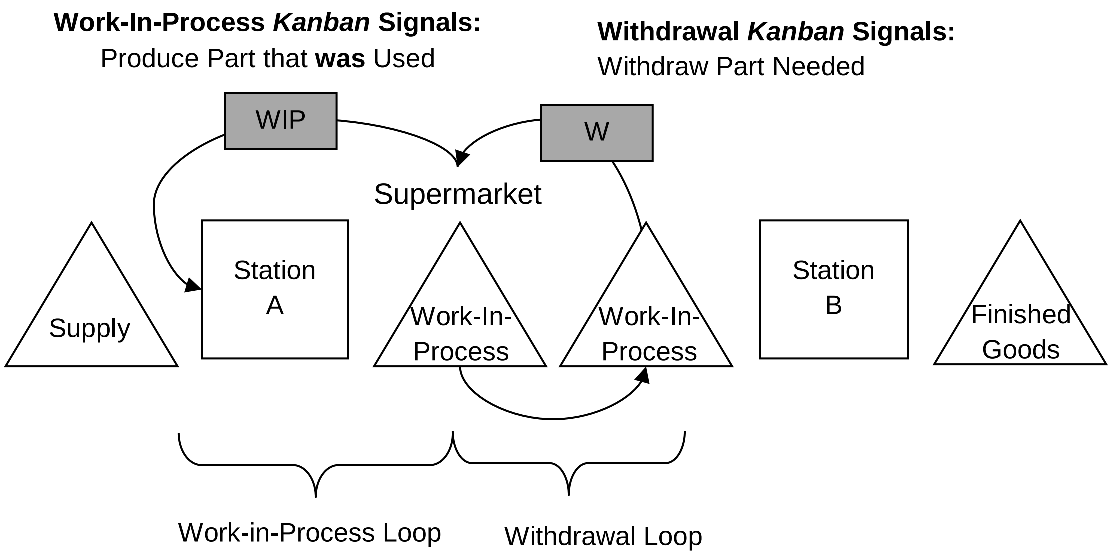

Figure 3: A Work-in-Process Kanban System for Coordinating Two Stations (Referred to as a Dual-Kanban System in the Literature)

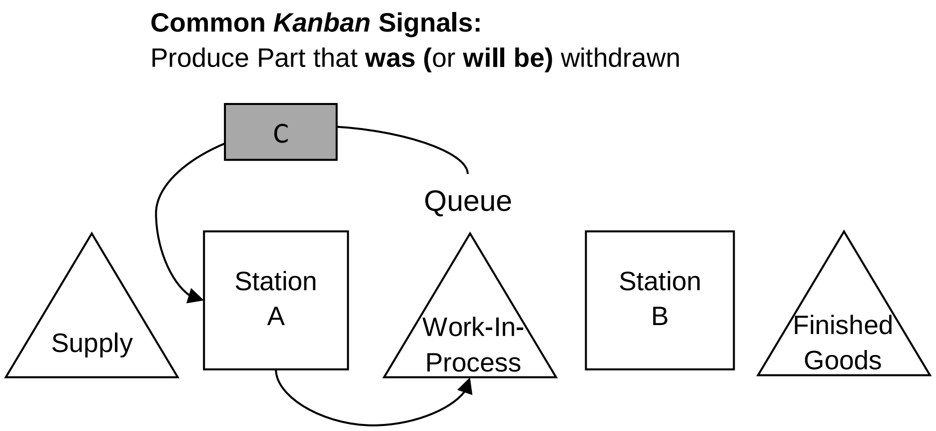

Figure 4: A Common Kanban System used for Coordinating Two Stations

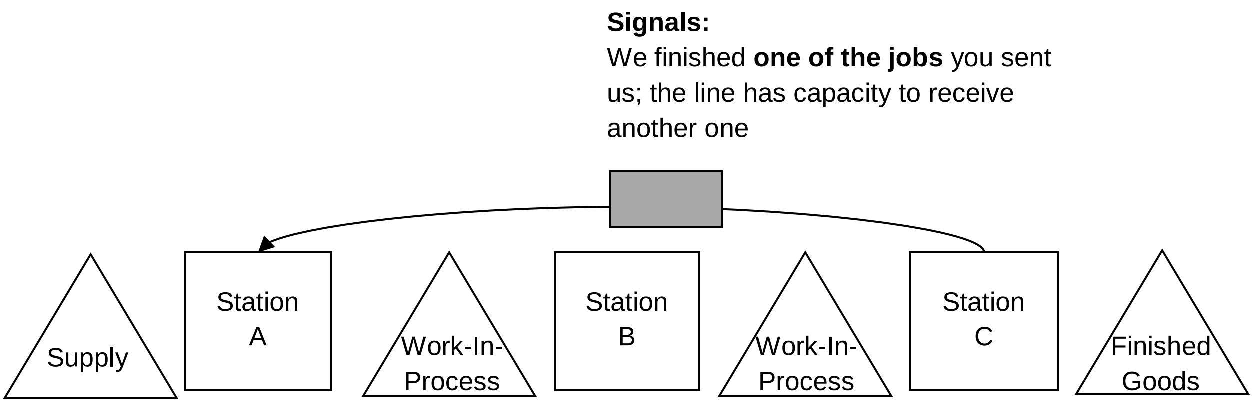

Figure 5: A ConWIP System (i.e. Anonymous Cards)

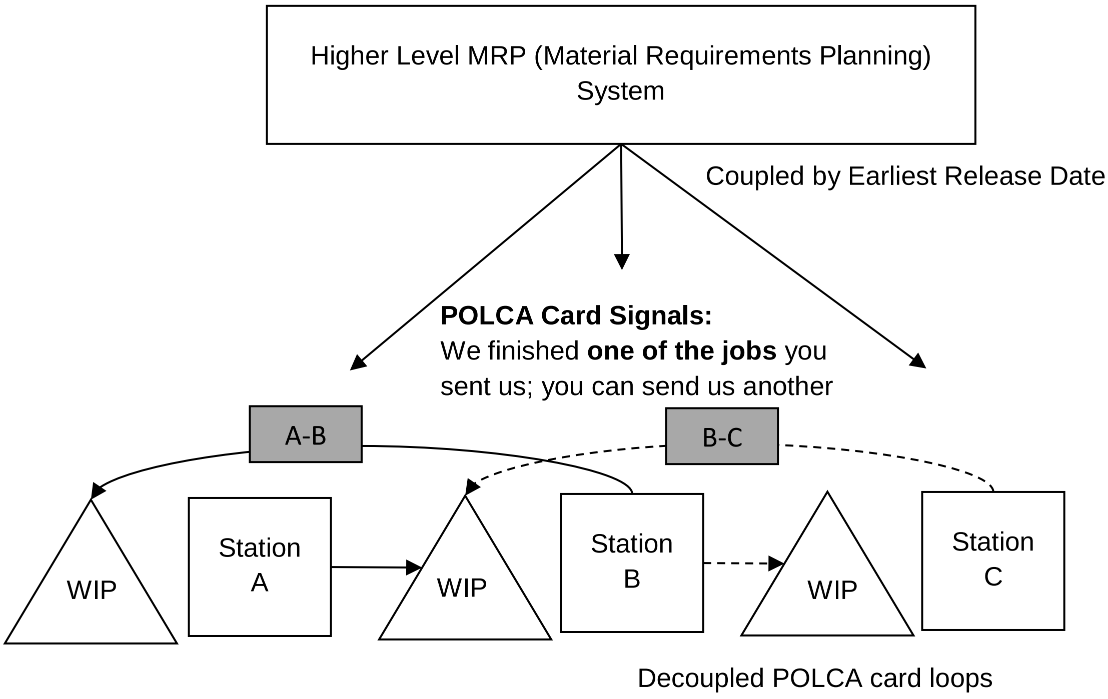

Figure 6: A POLCA System (Decoupled POLCA Loops Coupled by an MRP system)

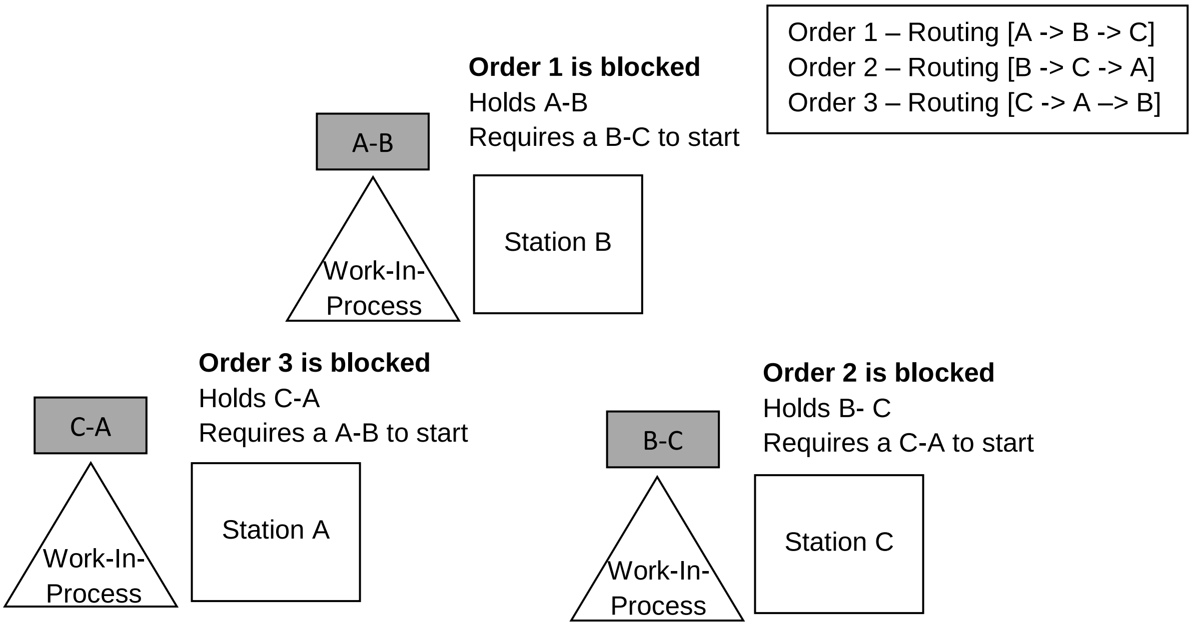

Figure 7: A Blocked POLCA System

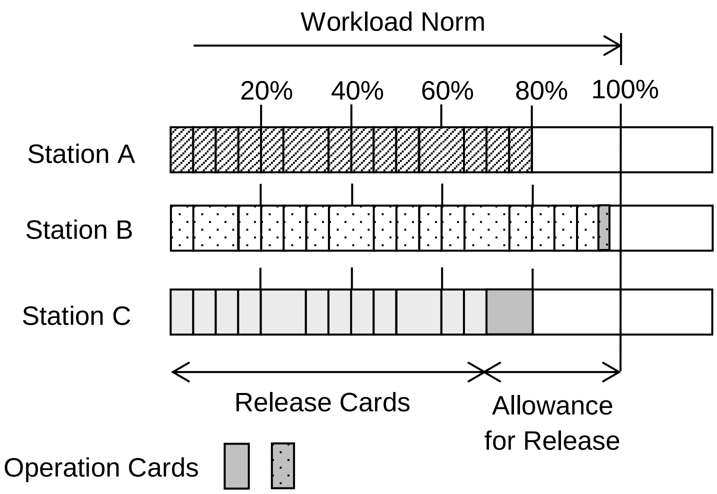

Figure 8: COBACABANA – The Planner's Planning Board for Order Release (with an Example Release Decision)

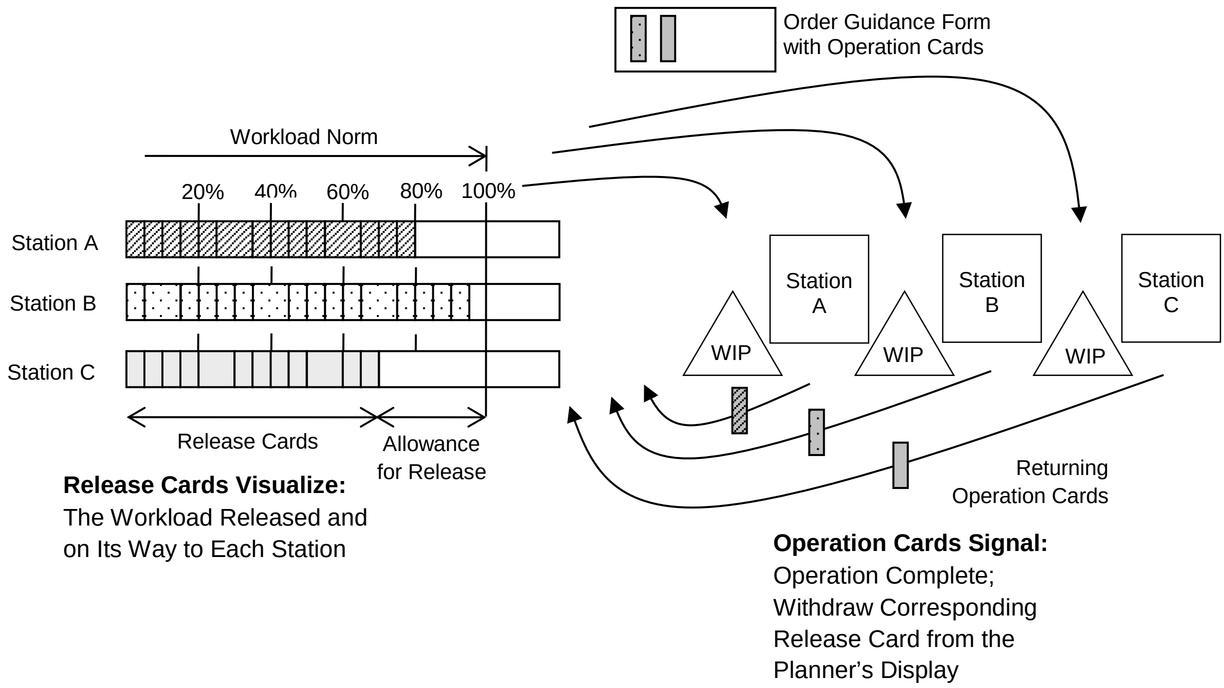

Figure 9: COBACABANA – Using Loops between a Central Release Function and Each Station on the Shop Floor