

Improving Performance in POLCA Controlled High Variety Shops: An Assessment by Simulation

Matthias Thürer (corresponding author), Nuno O. Fernandes, Sílvio Carmo-Silva and Mark Stevenson

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

Name: Professor Nuno O. Fernandes
Institution1: Instituto Politécnico de Castelo Branco
Address: Av. do Empresário
6000-767 Castelo Branco - Portugal
Institution2: University of Minho
Address: ALGORITMI Research Unit
4710-057 Braga – Portugal
Email: nogf@ipcb.pt

Name: Professor Sílvio Carmo-Silva
Institution: University of Minho
Address: ALGORITMI Research Unit
Campus de Gualtar
4710-057 Braga – Portugal
Email: scarmo@dps.uminho.pt

Name: Professor 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
Tel: 00 44 1524 593847

Keywords: *Job Shop; Order Release; Dispatching; Operations Management; Discrete Event Simulation.*

Improving Performance in POLCA controlled High Variety Shops: An Assessment by Simulation

Abstract

POLCA (i.e. Paired-cell Overlapping Loops of Cards with Authorization) is a card-based production control approach developed to support the adoption of Quick Response Manufacturing. The approach has received significant research attention but has remained largely unchanged since its introduction in the late 1990s. The main improvements have occurred in the context of an electronic POLCA system, but such developments undermine the simplicity of the original card-based concept. We ask: is there any refinement possible to enhance the performance of POLCA without jeopardizing its simplicity? By analyzing POLCA, two possible refinements are identified: (i) the choice of rule to support both the card allocation and dispatching decisions; and (ii) the use of a starvation avoidance mechanism to overcome premature station idleness, as reported in the context of load limiting order release. Using simulation, we demonstrate that performance gains can be obtained by using different rules for card allocation and dispatching other than the earliest release date rule typically applied in POLCA for both decisions. Further, results demonstrate performance improvements for all combinations of card allocation and dispatching rules considered via the addition of a simple starvation avoidance mechanism. Both refinements significantly enhance POLCA performance, potentially furthering its application in practice.

Keywords: *Job Shop; Order Release; Dispatching; Operations Management; Discrete Event Simulation.*

1. Introduction

POLCA (i.e. Paired-cell Overlapping Loops of Cards with Authorization) is a production planning and control system that combines a card-based pull element (the “POLC” in “POLCA”) with a higher-level Material Requirements Planning (HL/MRP) system for release Authorization (the “A” in “POLCA”). Suri [32] was the first to present POLCA as a production planning and control approach to support the adoption of his Quick Response Manufacturing philosophy, or the pursuit of time-based competition (Stalk [30]). POLCA has been argued to be an alternative to *kanban* systems specifically for companies that produce a high variety of products on a make-to-order basis (e.g. Krishnamurthy & Suri [18]; Riezebos [26]).

POLCA however has remained largely unchanged since its introduction in the late 1990s (Riezebos [26]). One of the few improvements reported has been the introduction of color-coded cards by Pieffers & Riezebos (2006, cited in Riezebos [26]). Stations were given a specific color, meaning each POLCA card consists of two colors, which allowed POLCA cards and routes to be identified more easily. Vandaele *et al.* [35] presented two further refinements but in the context of an electronic POLCA system. First, a method for dynamically determining lead time allowances based on a queuing model and a so-called ARP (Advanced Resources Planning) system instead of fixed lead time allowances. Second, an approach for setting the work-in-process limit per POLCA loop based on input data derived from the ARP system. The first refinement addressed a weakness in the prioritization of orders while the second addressed a weakness in capacity control. Vandaele *et al.*’s [35] refinements however rely on the use of an electronic POLCA system, which in turn requires rather specific expert knowledge. This undermines POLCA’s simplicity and, as a consequence, hinders its application to smaller shops with limited resources. These shops often operate as high variety make-to-order companies – the type of shop for which POLCA was originally designed (e.g. Suri [32]; Krishnamurthy & Suri [18]; Riezebos [26]).

Against the above backdrop, the question remains: is there any refinement possible to enhance the performance of POLCA without jeopardizing its simplicity? This paper seeks to address this question in order to enhance the performance of POLCA, thereby furthering its application in practice. To achieve our objective, POLCA is first analyzed to identify possible refinements before extensive simulation experiments are used to assess the effectiveness of our proposed refinements.

The remainder of this paper is structured as follows. In Section 2, we review the mechanisms underlying POLCA and outline possible means of refinement. The simulation model used to evaluate performance is then described in Section 3 before the results are presented, discussed, and analyzed in Section 4. Finally, conclusions are drawn in Section 5, where managerial implications and future research directions are also outlined.

2. Background – The POLCA Production Planning and Control System

POLCA is a production planning and control system that combines a card-based pull element with a HL/MRP system for release authorization. It can consequently be classified as a hybrid push/pull system (Esmaeilian *et al.* [8]); using a pull system local information about the status of production and inventories to control order release, while a push system relies on global information (Selçuk [28]). This section does not aim to present a comprehensive review of the POLCA literature; although no explicit review paper on POLCA exists, an extensive literature review is provided within the work of Riezebos [26]. The aim of this section is twofold: (i) to outline the POLCA system in order to provide insights into its underlying mechanisms; and, (ii) to outline proposals for refinement. Section 2.1 describes POLCA before Section 2.2 discusses proposals for refinement.

2.1 Mechanisms Underpinning a POLCA System

POLCA links the different stations in the routings of orders using card loops. POLCA uses card-loops between pairs of stations, e.g. between stations A and B. Each pair of consecutive stations, often referred to as cells in the POLCA literature, in the routing of a job has a POLCA card that identifies the two stations. A major difference between POLCA and, for example, *kanban* systems is that POLCA cards are job anonymous (Riezebos *et al.* [27]; Ziengs *et al.* [36]) while *kanban* cards are not (Shingo [29]). In this aspect, POLCA cards resemble ConWIP (Constant Work-In-Process) cards. In other words, POLCA cards do not indicate which job to work on – just that a job requires processing at two consecutive stations of the loop (e.g. A-B). As a consequence, there is still a need to choose between alternative jobs waiting in the queue of a station. In the POLCA system, this is provided by the release Authorization – the “A” in POLCA. POLCA’s authorization element uses earliest job release dates for each station, calculated by a HL/MRP system.

Let us 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 processing the order:

- (i) Station A must be available;
- (ii) A 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; and,
- (iii) The order must be authorized, i.e. the earliest release date calculated for this order at Station A must have been reached.

If this is the case, the POLCA A-B card is attached to the order and the order can be processed at Station A. Once complete at Station A, the order moves to Station B (and the A-B card remains attached to the order) where the same three conditions as above have to be met, replacing Station A by Station B, and so on. When the order is finished at Station B (and only then), the A-B card is freed and moves back to Station A; and the order moves to Station C, and so on. The overall POLCA system is depicted in Figure 1.

[Take in Figure 1]

2.2 Proposals for Refinement

This study started by asking:

Is there any refinement possible to enhance the performance of POLCA without jeopardizing its simplicity?

A first indication of where to look when refining POLCA is given by the refinements proposed by Vandaele *et al.* [35] in the context of an electronic POLCA system. These refinements focused on two areas: (i) improved prioritization of orders; and, (ii) capacity control. Therefore, and focusing on the structure of POLCA systems, two aspects with potential for improvement can be identified: (i) improved prioritization through the choice of different rules for card allocation and dispatching; and, (ii) improved use of capacity via the introduction of a starvation avoidance mechanism. Both will be discussed below, in sections 2.2.1 and 2.2.2, respectively.

2.2.1 Refinement 1: Improved Prioritization through Card Allocation and Dispatching Rules

In the original POLCA system, an order must be authorized, i.e. the earliest release date calculated for a particular order at a station must have been reached. But this condition is myopic

since there could be a better sequence of jobs for processing at a station; better in terms of different performance objectives. Therefore, we argue that the highest priority should be given to the job that is likely to contribute the most to system performance regardless of whether the earliest release date calculated for this job has been reached or not.

Jobs in a queue waiting to be processed may or may not have the required card from the next station in their routing. So, two rules for prioritization are required within POLCA. First, a card allocation rule that determines which job from the set of jobs without a card should receive the next card. Second, a dispatching rule that determines which job in the set of jobs with a card will be processed next at the station. In this aspect, POLCA is significantly different from ConWIP, where the acquisition of a card triggers release, and from *kanban* systems, where a *kanban* card always is associated to a specific order making a card-acquisition rule meaningless (see the third and fourth *kanban* rules presented in Ohno [23] (p.30) – ‘No items are made or transported without a *kanban*’ and ‘Always attach a *kanban* to the goods’).

POLCA typically assumes the same rule – earliest release dates (ERDs) – is applied for both card acquisition and dispatching decisions. This assumption is revisited in our study and POLCA is refined as follows: (i) other rules in addition to ERD are considered; and, (ii) the use of different rules for card allocation and for dispatching is trialed. A first indication of the potential impact of the dispatching rule was given by Braglia *et al.* [3]; however, this was in the context of an m-POLCA system in which POLCA cards are part number specific – as for m-ConWIP (Duenyas [6]) – rather than job anonymous as for the original POLCA system and ConWIP. Meanwhile, the card acquisition rule is important since there may be a relevant time lag between the time the card allocation decision is taken and the time that the processing of the job starts. Therefore, different jobs may be accumulated in the queue of a station. The card allocation decision determines the set of eligible jobs for the dispatching rule on the shop floor.

2.2.2 Refinement 2: Better use of Capacity through Starvation Avoidance

As early as Kanet [16], load limiting release methods – such as *kanban*, ConWIP (Constant Work-in-Process), Workload Control, and POLCA – have been criticized for introducing premature idleness. Premature idleness of a station means that it is starving due to the workload restriction at another station in the system despite the availability of jobs that could be processed directly at the starving station. Thürer *et al.* [34] recently demonstrated that premature idleness can be reduced by a starvation avoidance mechanism, which injects work – i.e. ‘feeds’ a station –

if a station is starving regardless of the workload limit at other stations. This leads to our second refinement. POLCA is refined to allow for the temporary violation of the card limit to avoid starvation. In our study, starvation avoidance (SA) cards are used for injecting work to a starving station. It should be noted that our focus is on premature idleness and starvation and not on POLCA-specific blocking, as observed, e.g. in Lödging *et al.* [21] and Harrod & Kanet [13].

Simulation is used to assess the impact of our two proposals for refinement. The following section outlines the simulation model and describes how both of our refinements have been implemented.

3. Simulation Model

In this study we consider a high variety make-to-order environment. A powerful tool for analyzing this kind of complex stochastic system is discrete event simulation (Negahban & Smith [22]). The shop and job characteristics modeled in the simulations are first outlined in Section 3.1. Section 3.2 then details how POLCA and our two refinements were modeled. Finally, the experimental design is outlined and the measures used to evaluate performance are presented in Section 3.3.

3.1 Overview of Modeled Shop and Job Characteristics

In recent simulation studies on POLCA (e.g. Germs & Riezebos [12]; Ziengs *et al.* [36]), a simple divergent shop structure was used. However, make-to-order companies that produce a high variety of products often use a functional layout and operate as some form of job shop (e.g. Hendry [14]). Enns [7] (p.2804) further argued that ‘*routeing in most real job shops lies somewhere between the pure job shop and pure flow shop extremes.*’ This *in-sequence with bypassing flow* is characteristic of the general flow shop (Aneke & Carrie [1]), as can be seen from Figure 2 which gives the flow characteristics of a six station pure job shop, general flow shop, and pure flow shop. Therefore, the general flow shop is considered in our study. The general flow shop also avoids the problem of feedback in the routing that leads to POLCA specific blocking (Lödging *et al.* [21]; Harrod & Kanet [13]). This kind of blocking will be explicitly avoided in our experimental design to omit interaction effects with premature idleness.

[Take in Figure 2]

A simulation model of a general flow shop 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 two 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.

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 as part of the operation processing time. Meanwhile, the inter-arrival time of orders follows an exponential distribution with a mean of 0.738, 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 35 and 55 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. While any individual high variety shop in practice will certainly differ from our stylized model, our model captures the high routing variability, processing time variability, and arrival variability that defines this context in practice. Finally, Table 1 summarizes the simulated shop and job characteristics.

[Take in Table 1]

3.2 POLCA and Refinements

As in previous simulation studies on POLCA (Lödding *et al.* [21]; Fernandes & Carmo-Silva [10]; Germs & Riezebos [12]; Harrod & Kanet [13]; Farnoush & Wiktorsson [9]; Braglia *et al.* [4]), it is assumed that materials are available and all necessary information regarding shop floor routing and processing times is known upon the arrival of an order to the shop. POLCA loops reflect every possible routing step of orders. Six levels for the number of cards per loop are considered: 8, 10, 12, 14, 16 cards per loop, and infinite cards. The same number of cards is used within each loop in a given experiment. This is justified by the balanced shop considered in our

study. How our two refinements – card allocation/dispatching rule and starvation avoidance – have been implemented is discussed in Section 3.2.1 and Section 3.2.2, respectively.

3.2.1 Card Allocation and Dispatching Rules

The card allocation and dispatching rule advocated in POLCA is the Earliest Release Date (ERD) rule, where the earliest release date is calculated by backward scheduling from the job due date based on throughput time allowances for each operation in the routing of a job. As suggested for POLCA (Riezebos [26]), and as is typical in the literature on modeling MRP systems (e.g. Krishnamurthy *et al.* [17]; Steele *et al.* [31]; Jodlbauer & Huber [15]), we use a constant allowance for the planned operation throughput time that is offset at each level. This allowance is based on preliminary simulation experiments. The ERD rule for card allocation and dispatching is our baseline measure. The rules considered for assessing the impact of our first refinement – improved prioritization through different card allocation and dispatching rules – will be introduced next.

For priority dispatching, we will consider the Shortest Processing Time and Modified ERD rule (MERD). The SPT rule is a load-based rule that has been previously shown to reduce throughput times in flow shops (e.g. Conway [5]). It selects the job with the shortest processing time from the queue. Meanwhile, the MERD rule combines the SPT and ERD rule. The MERD rule is a variant of the Modified Operation Due Date (MODD) rule proposed, e.g. by Baker & Kanet [2]. MERD essentially subdivides the set of eligible jobs into two subsets: a subset of urgent jobs for which the ERD has already passed and a subset of non-urgent jobs. Urgent jobs always receive priority over non-urgent jobs, whereby urgent jobs are selected for processing according to SPT and non-urgent jobs are selected according to ERD. The MERD rule shifts between a focus on ERDs, to complete jobs on time, and a focus on speeding up jobs – through SPT effects – during periods of high load, i.e. when multiple jobs exceed their ERD (Land *et al.* [14]).

For card allocation, the SPT rule is substituted by a capacity slack-based rule (see, e.g. Philipoom *et al.* [25]; Fredendall *et al.* [11]; Thüerer *et al.* [33]). The SPT rule focuses only on the station where the job is queuing. Hence, it has a local view and does not take into account the next station(s) in the routing of an order, which may potentially be contained in the same loop. Therefore, instead of prioritizing jobs in the set of jobs without a card according to shortest processing times, they are prioritized according to a capacity slack ratio as given by Equation (1)

below. The lower the capacity slack ratio S_j of job j , the higher is the priority of job j . The rule integrates three elements into one priority measure: the *workload contribution* of a job (i.e. 1); the *load gap*, (i.e. the difference between a load limit or norm N_s and the current workload W_s at station s corresponding to operation i : $N_s - W_s$); and, the *routing length* (i.e. the number of operations in the remaining 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 remaining routing of a job. The workload W_s and the limit N_s are associated with a station. Since the limit is enforced on station pairs, the limit N_s is obtained by dividing the number of jobs allowed per loop by two. The resulting card allocation rule will be referred to as CS. Meanwhile, MERD transforms into the MODCS rule by replacing the SPT rule for the set of urgent orders by the CS rule.

$$S_j = \frac{\sum_i \frac{1}{N_s - W_s}}{n_j} \quad i: 1 \dots n_j \quad (1)$$

POLCA does not limit the workload W_s at each station; therefore, this workload may exceed the limit N_s resulting in a negative priority value. As a result, 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 $N_s - W_s \leq 0$, then the job is positioned at the back of the queue by replacing the component $\left(\frac{1}{N_s - W_s} \right)$ related to this station in the priority value S_j with M , where M is a sufficiently large number.

The final set of card allocation and dispatching rules considered in our study is summarized in Table 2.

[Take in Table 2]

3.2.2 Starvation Avoidance Mechanism

On some occasions, a station may be starving although there is work in the queue, e.g. when all available POLCA cards that authorize production at that station are at the succeeding station. This form of premature idleness can be resolved by attaching a Starvation Avoidance (SA) card

to the job from the set of jobs without cards that has the highest priority (according to the card allocation rule applied) thereby allowing it to be processed at the starving station. Using an SA card means that the work-in-process cap or limit will be exceeded. Thus, similar to Workload Control, POLCA recognizes the need to temporarily violate the work-in-process limit to avoid premature idleness (Thürer *et al.* [34]). In order to restore the limit, POLCA cards do not become available after being detached from jobs as long as SA cards are in use. Only after all SA cards are returned can the POLCA card be used.

In order to test the impact of different levels of SA cards within each loop, four scenarios are considered: none (the original POLCA system), 1 SA card, 2 SA cards, and infinite SA cards, i.e. as many cards as required.

3.3 Experimental Design and Performance Measures

The experimental factors, as summarized in Table 3, are: (i) the starvation avoidance mechanism (none, 1, 2, and infinite SA cards per loop); (ii) the card allocation rule (ERD, CS, and MODCS); (iii) the dispatching rule (ERD, SPT, and MERD); and, (iv) the six different levels for the number of POLCA cards (8, 10, 12, 14, 16, and infinite cards). A full factorial design was used with 216 ($4 \times 3 \times 3 \times 6$) 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 to minimize initialization bias. These parameters allow us to obtain stable results while keeping the simulation run time to a reasonable level.

[Take in Table 3]

Four main performance measures are considered in this study as follows: mean *total throughput time* – the mean of the completion date minus the pool entry date across jobs; *percentage tardy* – the percentage of jobs completed after the due date; *mean tardiness* – that is, $T_j = \max(0, L_j)$, with L_j being the lateness of job j (i.e. the actual delivery date minus the due date of job j); and, the *standard deviation of lateness*.

The total throughput time is used as the main indicator of the balancing capabilities of the approaches being tested. It also reflects the average lateness of jobs, which can be derived directly from this measure (it is equal to the realized average total throughput time minus the average delivery time allowance). The main indicator of delivery performance is the percentage

of tardy jobs, which is influenced by both the average lateness and the dispersion of lateness across jobs. Finally, in addition to the four main performance measures, 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. The average shop floor throughput time is a useful indicator of the work-in-process level on the shop floor as, according to Little's Law (Little [20]), it is linked directly to the level of work-in-process.

4. Results

To obtain a first indication of the relative impact of the experimental factors, statistical analysis has been conducted by applying an Analysis of Variance (ANOVA). ANOVA is here based on a block design with the number of POLCA cards in each card loop as the blocking factor, i.e. the six levels of cards per loop were treated as different systems. A block design allowed the main effect of the number of cards and both the main and interaction effects of our other three factors (starvation avoidance mechanism, card allocation rule, and dispatching rule) to be captured. The results – as summarized in Table 4 – show that all main effects and most two-way and three-way interactions are significant at $\alpha=0.05$. There are no significant two-way interactions between starvation avoidance and the card allocation rule and starvation avoidance and the dispatching rule in terms of shop floor throughput time performance. Meanwhile, the three-way interactions are not significant for the shop floor throughput time and the total throughput time.

[Take in Table 4]

The Scheffé multiple-comparison procedure was used to further examine the significance of the differences between the outcomes of the individual card allocation and dispatching rules. Table 5 and Table 6 summarize the 95% confidence intervals for the card allocation and dispatching rules, respectively. Meanwhile, Table 7 gives the confidence interval for the two-way interaction between the card allocation and dispatching rules. Differences are considered not significant if the interval includes zero.

[Take in Table 5, Table 6 & Table 7]

Significant differences between the outcomes of all card allocation and dispatching rules can be identified from Table 5 and Table 6, except for the CS and MODCS card allocation rules, which perform statistically equivalent in terms of the percentage tardy, and the MODCS and ERD card allocation rules, which perform statistically equivalent in terms of the total throughput time. Meanwhile, Table 7 shows significant differences between the outcomes of the different combinations of card allocation and dispatching rules. In order to further assess performance differences, detailed performance results will be presented next in Section 4.1 where we focus on the impact of our refinements if dispatching follows ERD. This means, we focus on: (i) the impact of different combinations of our three allocation rules and ERD; and, (ii) starvation avoidance. The performance impact of SPT and MERD dispatching is then assessed in Section 4.2.

4.1 Performance Assessment (Under ERD Dispatching)

A major challenge when comparing different control policies is the creation of comparable states; a certain parameter setting may favor one policy over another thereby making conclusions dependent on parameter settings rather than on the actual policy. A means of realizing a ‘fair’ comparison is via the use of operating characteristic curves (Olhager & Persson [24]). Rather than comparing one specific parameter setting, parameters are varied for each policy and the results presented in the form of performance curves. The relative positioning of the different curves (each representing one policy) then allows for comparing the relative performance of each policy.

In our study, the main parameter determining POLCA performance is the number of cards allowed in a POLCA loop. This parameter is therefore used to create our performance curves. The left-hand starting point of the curves represents the lowest number of cards allowed in a POLCA loop (i.e. 8 cards). The number of cards allowed increases step-wise by moving from left to right in each graph, with each data point representing one card level (i.e. 8, 10, 12, 14, 16 cards and infinite cards). Increasing the number of cards increases the level of work-in-process and, as a result, increases the shop floor throughput time. Figures 3a, 3b, and 3c show the total throughput time, percentage tardy, mean tardiness, and standard deviation of lateness results over the shop floor throughput time results for the ERD, CS, and MODCS card allocation rules, respectively. Only results for ERD dispatching are shown here, with the impact of the dispatching rule assessed in Section 4.2.

[Take in Figure 3]

In terms of our two refinements, the following can be observed from the results:

- *Refinement 1 - Different Combinations of Card Allocation and Dispatching Rule (across Figures 3a-3c):* The original POLCA system used ERD for card allocation and dispatching. The results for this scenario are given in Figure 3a. Compared to these results, we see that the combination of CS and ERD (i.e. replacing ERD by a CS allocation rule) significantly reduces total throughput times and the percentage tardy. The former can be attributed to a strong reduction in shop floor throughput times (i.e. a shift to the left of the performance curves). Meanwhile, mean tardiness and the standard deviation of lateness performance is maintained. Hence, it can be concluded that the CS rule is a better choice than ERD for card allocation. Meanwhile, MODCS (Figure 3c) leads to the lowest percentage tardy, but this is at the expense of mean tardiness and standard deviation of lateness performance. POLCA does not restrict the number of jobs queuing in front of a work station – it just restricts the number of jobs that hold a card. Thus, there are significant fluctuations in the workload queuing at each station, resulting in high-load periods during which the workload measure W_s at each station exceeds the limit N_s . During these periods the ratio between the load contribution and load gap elements is substituted by M , where M is a sufficiently large number. As a result, specifically in high load periods, capacity slack rules are less effective, even though these are the periods when improved load balancing is needed the most (Land *et al.* [19]).
- *Refinement 2 - Starvation Avoidance (within each Figure):* The use of SA cards to avoid premature idleness significantly improves the performance of POLCA for all four main performance measures considered. As somewhat expected, the positive effect of SA increases as the number of cards in each POLCA loop reduces, i.e. if we move to the left on each curve. Reducing the number of POLCA cards increases the risk of premature idleness and consequently increases the importance of starvation avoidance. While the use of one SA card leads to significant performance improvement, the full potential of SA cards is realized by allowing for an infinite number of cards, i.e. when no limit is put on the number of SA cards. Our results thus confirm the validity and positive impact of this refinement.

4.2 Performance Assessment – The Impact of the Priority Dispatching Rule

The above section focused on the effect of the card allocation rule and starvation avoidance where ERD dispatching was applied. The objective of this section is twofold: first, to assess the impact of combinations of card allocation and dispatching rules that include SPT and MERD dispatching; and, second, to assess the robustness of our results for starvation avoidance to these combinations. The results for SPT dispatching are given in Figure 4 (4a to 4c) and for MERD dispatching in Figure 5 (5a to 5c) for the ERD, CS, and MODCS card allocation rules, respectively. Again, the total throughput time, percentage tardy, mean tardiness, and standard deviation of lateness results over the shop floor throughput time results are given.

[Take in Figure 4 & Figure 5]

The SPT dispatching rule leads to an expected reduction in total and shop floor throughput times and the percentage of tardy jobs compared to ERD (Figure 3). This is at the expense of mean tardiness performance and the standard deviation of lateness. Reducing the number of cards (i.e. moving from right to left in each figure) reduces the effect of the dispatching rule since it restricts the set of eligible jobs. This leads to performance improvements in terms of mean tardiness and the standard deviation of lateness and to deterioration in terms of the total throughput time at tighter card levels. MERD dispatching (Figure 5) leads to the best performance across all three dispatching rules when applied in isolation; this can be observed by comparing the right-hand starting point of the curves in Figure 3, Figure 4, and Figure 5 which gives the results for infinite cards, i.e. where control is only exercised by the dispatching rule.

In terms of our two refinements, the following can be observed from the results:

- *Refinement 1 - Different Combinations of Card Allocation and Dispatching Rule:* For SPT dispatching (Figure 4), using the ERD rule for card allocation leads to better performance than the CS rule. SPT effects are already provided by the dispatching rule, so focusing on the urgency of orders leads to better performance in terms of the percentage tardy, mean tardiness, and standard deviation of lateness. Meanwhile, for MODCS, the same observation as for ERD dispatching (Figure 3 above) applies – while it reduces the percentage tardy compared to alternative card allocation rules, this is at the expense of mean tardiness and the standard deviation of lateness performance. This observation is also valid for MERD dispatching (Figure 5). The relative performance of the card allocation rules resembles that observed for

ERD dispatching (Figure 3 above), with the best performance achieved with the CS rule. This confirms our refinement in two aspects. First, ERD is not the preferred choice of card allocation/dispatching rule. Second, using different rules for card allocation (CS) and dispatching (MERD) leads to the best performance.

- *Refinement 2 - Starvation Avoidance:* The use of SA cards to avoid premature idleness significantly improves the performance of POLCA for all main performance measures under all scenarios considered, i.e. regardless of the combination of card allocation rule and dispatching rule applied. Our results thus further confirm the validity and positive impact of this refinement.

5. Conclusions

POLCA is an important production planning and control concept developed to support the adoption of Quick Response Manufacturing (e.g. Suri [32]) in companies that produce a high variety of products on a to-order basis. There has been significant research attention on POLCA, and a number of studies have reported on implementations of the approach in practice. While POLCA as a card-based system has remained largely unchanged since its introduction, there has been significant improvement in the context of an electronic POLCA system. This system however relies on electronic data availability and rather specific expert knowledge. This increase in complexity arguably hinders its application to smaller shops with limited resources, although it is often this type of shop that operates on a high variety make-to-order basis. In response, we asked: is there any refinement possible to enhance the performance of POLCA without jeopardizing its simplicity? Based on previous literature and an analysis of POLCA's underlying structure, two possible refinements were identified: (i) the choice of different combinations of card allocation and dispatching rules; and (ii) the introduction of a starvation avoidance mechanism.

Using simulation, we demonstrated the effectiveness of both refinements. First, results suggest a combination of a capacity slack (CS) based card allocation rule from the Workload Control literature in combination with MERD dispatching, a modified earliest release date rule. Thus, not only should the ERD card allocation/dispatching rule typically applied in the POLCA literature be replaced, but additional performance gains can be obtained by using different rules for card allocation and dispatching. Second, significant performance improvements for all

combinations of card allocation and dispatching rule considered in this study were realized through the use of a starvation avoidance mechanism. This mechanism itself relies on the use of simple SA cards that allow for the release of work to a starving station even if this temporarily violates the card limit applied. It is hoped that the improved performance observed for the refined POLCA system furthers POLCA application in practice.

A major limitation of our study is its restricted environmental setting. We have focused on a general flow shop as it was argued in the literature that many high variety make-to-order shops operate as general flow shops. Moreover, this shop type avoids feedback in the routing, thereby avoiding POLCA-related blocking. Future research could however consider more complex shops such as those that include feedback in the routing. This would also allow means of combining starvation avoidance with a resolution for POLCA-related blocking to be explored. Finally, while our refinements were proven to be effective through simulation, future research is required to assess their impact and suitability in practice.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (71550110254); COMPETE: POCI-01-0145-FEDER-007043 and FCT – Fundação para a Ciência e Tecnologia within the Project Scope: UID/CEC/00319/2013.

References

- [1] Aneke, N.A.G., and Carrie, A.A., 1986, A design technique for the layout of multi-product flow lines, *International Journal of Production Research*, 24, 3, 471-481.
- [2] Baker, K.R., and Kanet, J.J., 1983, Job shop scheduling with modified operation due-dates, *Journal of Operations Management*, 4, 1, 11-22.
- [3] Braglia, M., Castellano, D., and Frosolini, M., 2015, A study on the importance of selection rules within unbalanced MTO POLCA-controlled production systems, *International Journal of Industrial and Systems Engineering*, 20, 4, 457-468.
- [4] Braglia, M., Castellano, D., and Frosolini, M., 2014, Optimization of POLCA-controlled production systems with a simulation-driven genetic algorithm, *International Journal of Advanced Manufacturing Technology*, 70, 385 – 395.

- [5] Conway, R., Maxwell, W.L., and Miller, L.W., 1967, *Theory of Scheduling*, Reading, MA: Addison-Wesley.
- [6] Duenyas, I., 1994, A simple release policy for networks of queues with controllable inputs, *Operations Research*, 42, 1162–1171.
- [7] Enns, S.T., 1995, An integrated system for controlling shop loading and work flows, *International Journal of Production Research*, 33, 10, 2801-2820.
- [8] Esmailian, B., Behdad, S., and Wang, B., 2016, The evolution and future of manufacturing: A review, *Journal of Manufacturing Systems*, 39, 79-100.
- [9] Farnoush, A., and Wiktorsson, M., 2013, POLCA and CONWIP performance in a divergent production line: an automotive case study, *Journal of Management Control*, 24, 159 – 186.
- [10] Fernandes, N.O. and Carmo-Silva, S., 2006. Generic POLCA - A production and materials flow control mechanism for quick response manufacturing, *International Journal of Production Economics*, 104, 1, 74-84.
- [11] Fredendall, L.D., Ojha, D., and Patterson, J.W., 2010, Concerning the theory of workload control, *European Journal of Operational Research*, 201, 1, 99 – 111.
- [12] Germs, R., and Riezebos, J., 2010, Workload balancing capability of pull systems in MTO production, *International Journal of Production Research*, 48, 8, 2345-2360.
- [13] Harrod, S., and Kanet, J.J., 2013, Applying work flow control in make-to-order shops, *International Journal of Production Economics*, 143, 620-626.
- [14] Hendry, L., 1998, Applying world class manufacturing to make-to-order companies: problems and solutions, *International Journal of Operations & Production Management*, 18, 11, 1086-1100.
- [15] 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.
- [16] Kanet, J.J., 1988, Load-limited order release in job shop scheduling systems, *Journal of Operations Management*, 7, 3, 44 – 58.
- [17] Krishnamurthy, A., Suri, R., and Vernon, M., 2004. Re-Examining the Performance of MRP and Kanban Material Control Strategies for Multi-Product Flexible Manufacturing Systems, *International Journal of Flexible Manufacturing Systems*, 16, 1, 123–150.

- [18] 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.
- [19] 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.
- [20] Little, J., 1961, A proof of the theorem $L = \lambda W$, *Operations Research* 8, 383-387.
- [21] Lödding, H., Yu, K.-W., and Wiendahl, H.-P., 2003, Decentralized WIP-oriented manufacturing control (DEWIP), *Production Planning & Control*, 14, 1, 42-54.
- [22] Negahban, A., and Smith, J.S., 2014, Simulation for manufacturing system design and operation: Literature review and analysis, *Journal of Manufacturing Systems*, 33, 241-261.
- [23] Ohno, T., 1988, *Toyota Production System: Beyond Large-Scale Production*, 1st ed. Cambridge, MA: Productivity Press.
- [24] 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
- [25] 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.
- [26] Riezebos, J., 2010, Design of POLCA material control systems, *International Journal of Production Research*, 48, 5, 1455-1477.
- [27] Riezebos, J., Klingenberg, W., and Hicks, C., 2009, Lean Production and information technology: Connection or contradiction?, *Computers in Industry*, 60, 237 – 247.
- [28] Selçuk, B., 2013, Adaptive lead time quotation in a pull production system with lead time responsive demand, *Journal of Manufacturing Systems*, 32, 138-146.
- [29] Shingo, S., 1989, *A Study of the Toyota Production System from an Industrial Engineering Viewpoint*, Cambridge, MA: Productivity Press.
- [30] Stalk, G., 1990, *Competing against time: How time-based competition is reshaping global markets*. Simon and Schuster.

- [31] Steele, D.C., Philipoom, P.R., Malhotra, M.K., and Fry T.D., 2005, Comparisons between drum-buffer-rope and material requirements planning: a case study, *International Journal of Production Research*, 43, 15, 3181-3208.
- [32] Suri, R., 1998, *Quick Response Manufacturing: A companywide approach to reducing leadtimes*, Productivity Press.
- [33] 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.
- [34] 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.
- [35] Vandaele, N., Van Nieuwenhuysse, 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.
- [36] Ziengs, N., Riezebos, J., & Germs, R., 2012, Placement of effective work-in-progress limits in route-specific unit-based pull systems, *International Journal of Production Research*, 50, 16, 4358-4371.

Table 1: Summary of Simulated Shop and Job Characteristics

Shop Characteristics	Routing Variability No. of Work Centers Interchange-ability of Work Centers Work Center Capacities Work Center Utilization Rate	Random routing; directed, no re-entrant flows 6 No interchange-ability All equal 90%
Job Characteristics	No. of Operations per Job Operation Processing Times Due Date Determination Procedure Inter-Arrival Times	Discrete Uniform[2, 6] Truncated 2–Erlang; (mean = 0.99; max = 4) Due Date = Entry Time + d ; $d \sim U[35, 55]$ Exp. Distribution; mean = 0.738

Table 2: Summary of the Three Card Allocation Rules and Three Dispatching Rules Applied in This Study

Card Allocation Rule	Dispatching Rule	Notes
Earliest Release Date (ERD) The job with the earliest release date is considered first.	Earliest Release Date (ERD) The job with the earliest release date is considered first.	Time-based reference rule from the POLCA literature
Capacity Slack (CS) The job with the lowest capacity slack ratio (see Eq. 1) is considered first.	Shortest Processing Time (SPT) The job with the shortest processing time is considered first.	Load-based rule
Modified Capacity Slack (MODCS) Jobs are divided into two classes: urgent, i.e. jobs with an ERD that has already passed the current date; and non-urgent. Urgent jobs are considered first according to the CS rule. Non-urgent jobs are considered according to the ERD rule.	Modified Earliest Release Date (MERD) Jobs are divided into two classes: urgent, i.e. jobs with an ERD that has already passed the current date; and non-urgent. Urgent jobs are considered first according to the SPT rule. Non-urgent jobs are considered according to the ERD rule.	Combines time-based and load based rule

Table 3: Summary of Experimental Factors

Starvation Avoidance Mechanism	none, 1, 2, and infinite SA cards per loop
Card Allocation Rule	ERD, CS and MODCS
Dispatching Rule	ERD, SPT and MERD
Number of POLCA Cards	8, 10, 12, 14, 16, and infinite cards

Table 4: ANOVA Results

	Source of Variance	Sum of Squares	Degree of Freedom	Mean Squares	F-Ratio	p-Value
Total Throughput Time	Number of Cards	614.01	5	122.80	24.32	0.00
	Starvation Avoidance (SA)	1395.55	3	465.18	92.14	0.00
	Card Allocation Rule (CA)	832.71	2	416.36	82.47	0.00
	Dispatching Rule (D)	692461.43	2	346230.71	68578.45	0.00
	SA x CA	90.01	6	15.00	2.97	0.01
	SA x D	608.39	6	101.40	20.08	0.00
	CA x D	335.42	4	83.86	16.61	0.00
	SA x CA x D	50.58	12	4.22	0.83	0.61
	Residual	108844.52	21559	5.05		
Percentage Tardy	Number of Cards	1.24	5	0.25	201.58	0.00
	Starvation Avoidance (SA)	0.10	3	0.03	27.94	0.00
	Card Allocation Rule (CA)	2.11	2	1.05	856.64	0.00
	Dispatching Rule (D)	4.34	2	2.17	1765.17	0.00
	SA x CA	0.09	6	0.01	11.64	0.00
	SA x D	0.05	6	0.01	6.29	0.00
	CA x D	1.06	4	0.27	215.92	0.00
	SA x CA x D	0.04	12	0.00	2.86	0.00
	Residual	26.50	21559	0.00		
Mean Tardiness	Number of Cards	1004.85	5	200.97	247.70	0.00
	Starvation Avoidance (SA)	183.31	3	61.10	75.31	0.00
	Card Allocation Rule (CA)	568.32	2	284.16	350.23	0.00
	Dispatching Rule (D)	3221.79	2	1610.89	1985.45	0.00
	SA x CA	35.29	6	5.88	7.25	0.00
	SA x D	88.30	6	14.72	18.14	0.00
	CA x D	221.07	4	55.27	68.12	0.00
	SA x CA x D	20.15	12	1.68	2.07	0.02
	Residual	17491.91	21559	0.81		
Standard Deviation of Lateness	Number of Cards	69698.24	5	13939.65	363.77	0.00
	Starvation Avoidance (SA)	7143.38	3	2381.13	62.14	0.00
	Card Allocation Rule (CA)	66133.29	2	33066.64	862.91	0.00
	Dispatching Rule (D)	226684.47	2	113342.24	2957.80	0.00
	SA x CA	4867.61	6	811.27	21.17	0.00
	SA x D	3685.31	6	614.22	16.03	0.00
	CA x D	28873.29	4	7218.32	188.37	0.00
	SA x CA x D	2512.95	12	209.41	5.46	0.00
	Residual	826134.97	21559	38.32		
Shop Floor Throughput Time	Number of Cards	12355.94	5	2471.19	718.51	0.00
	Starvation Avoidance (SA)	87.49	3	29.16	8.48	0.00
	Card Allocation Rule (CA)	519.81	2	259.90	75.57	0.00
	Dispatching Rule (D)	582598.10	2	291299.05	84696.92	0.00
	SA x CA	31.78	6	5.30	1.54	0.16
	SA x D	42.82	6	7.14	2.08	0.05
	CA x D	221.03	4	55.26	16.07	0.00
	SA x CA x D	14.65	12	1.22	0.36	0.98
	Residual	74148.10	21559	3.44		

*Table 5: Results for Scheffé Multiple Comparison Procedure:
Card Allocation Rules*

Rule (x)	Rule (y)	Total Throughput Time		Percentage Tardy		Mean Tardiness		SD Late	
		lower ¹⁾	upper	lower	upper	lower	upper	lower	upper
CS	ERD	-0.54	-0.36	-0.02	-0.02	0.05	0.12	0.91	1.42
MODCS	ERD	-0.17*	0.01	-0.02	-0.02	0.34	0.42	3.90	4.41
MODCS	CS	0.28	0.46	-0.01*	0.00	0.26	0.33	2.74	3.24

¹⁾ 95% confidence interval; * not significant at $\alpha=0.05$

*Table 6: Results for Scheffé Multiple Comparison Procedure:
Dispatching Rules*

Rule (x)	Rule (y)	Total Throughput Time		Percentage Tardy		Mean Tardiness		SD Late	
		lower ¹⁾	upper	lower	upper	lower	upper	lower	upper
SPT	ERD	-12.59	-12.41	-0.04	-0.03	0.70	0.78	6.40	6.90
MERD	ERD	-1.14	-0.96	-0.02	-0.02	-0.18	-0.10	-0.68	-0.17
MERD	SPT	11.36	11.54	0.01	0.02	-0.92	-0.84	-7.33	-6.82

¹⁾ 95% confidence interval; * not significant at $\alpha=0.05$

*Table 7: Results for Scheffé Multiple Comparison Procedure:
Card Allocation Rule x Dispatching Rule*

Rule ¹⁾ (x, y)	Rule (x, y)	Total Throughput Time		Percentage Tardy		Mean Tardiness		SD Late	
		lower ²⁾	upper	lower	upper	lower	upper	Lower	upper
(1, 2)	(1, 1)	-12.98	-12.47	-0.06	-0.05	0.84	1.04	8.49	9.90
(1, 3)	(1, 1)	-1.27	-0.75	-0.02	-0.02	-0.22	-0.01	-0.94*	0.46
(2, 1)	(1, 1)	-0.90	-0.39	-0.03	-0.02	0.01	0.21	0.99	2.40
(2, 2)	(1, 1)	-13.03	-12.52	-0.06	-0.05	0.87	1.08	8.61	10.02
(2, 3)	(1, 1)	-1.93	-1.42	-0.05	-0.04	-0.11*	0.10	0.74	2.15
(3, 1)	(1, 1)	-0.33*	0.18	-0.04	-0.03	0.47	0.68	5.65	7.06
(3, 2)	(1, 1)	-12.98	-12.47	-0.06	-0.05	0.89	1.10	8.79	10.20
(3, 3)	(1, 1)	-1.43	-0.92	-0.06	-0.05	0.29	0.49	4.87	6.28
(1, 3)	(1, 2)	11.46	11.97	0.03	0.04	-1.16	-0.95	-10.14	-8.73
(2, 1)	(1, 2)	11.82	12.33	0.02	0.03	-0.93	-0.73	-8.20	-6.79
(2, 2)	(1, 2)	-0.30*	0.21	0.00*	0.00	-0.07*	0.14	-0.58*	0.82
(2, 3)	(1, 2)	10.80	11.31	0.01	0.01	-1.05	-0.84	-8.46	-7.05
(3, 1)	(1, 2)	12.39	12.90	0.02	0.02	-0.47	-0.26	-3.55	-2.14
(3, 2)	(1, 2)	-0.26*	0.25	0.00*	0.00	-0.05*	0.16	-0.41*	1.00
(3, 3)	(1, 2)	11.29	11.80	0.00*	0.01	-0.65	-0.45	-4.32	-2.92
(2, 1)	(1, 3)	0.11	0.62	-0.01*	0.00	0.12	0.33	1.23	2.64
(2, 2)	(1, 3)	-12.02	-11.51	-0.04	-0.03	0.99	1.19	8.85	10.26
(2, 3)	(1, 3)	-0.92	-0.41	-0.03	-0.02	0.01	0.21	0.98	2.39
(3, 1)	(1, 3)	0.68	1.19	-0.02	-0.01	0.59	0.79	5.89	7.30
(3, 2)	(1, 3)	-11.97	-11.46	-0.04	-0.03	1.01	1.21	9.03	10.44
(3, 3)	(1, 3)	-0.42*	0.09	-0.04	-0.03	0.40	0.61	5.11	6.52
(2, 2)	(2, 1)	-12.38	-11.87	-0.03	-0.02	0.76	0.97	6.91	8.32
(2, 3)	(2, 1)	-1.28	-0.77	-0.02	-0.01	-0.22	-0.01	-0.96*	0.45
(3, 1)	(2, 1)	0.31	0.82	-0.01	-0.01	0.36	0.57	3.95	5.36
(3, 2)	(2, 1)	-12.34	-11.83	-0.03	-0.03	0.78	0.99	7.09	8.50
(3, 3)	(2, 1)	-0.79	-0.28	-0.03	-0.02	0.18	0.38	3.17	4.58
(2, 3)	(2, 2)	10.84	11.35	0.01	0.01	-1.08	-0.88	-8.58	-7.17
(3, 1)	(2, 2)	12.44	12.95	0.01	0.02	-0.50	-0.29	-3.66	-2.26
(3, 2)	(2, 2)	-0.21*	0.30	0.00*	0.00	-0.08*	0.12	-0.53*	0.88
(3, 3)	(2, 2)	11.34	11.85	0.00*	0.01	-0.69	-0.48	-4.44	-3.04
(3, 1)	(2, 3)	1.34	1.85	0.01	0.01	0.48	0.68	4.21	5.61
(3, 2)	(2, 3)	-11.31	-10.80	-0.01	-0.01	0.90	1.10	7.35	8.75
(3, 3)	(2, 3)	0.24	0.75	-0.01*	0.00	0.29	0.50	3.43	4.84
(3, 2)	(3, 1)	-12.91	-12.40	-0.02	-0.02	0.32	0.52	2.44	3.84
(3, 3)	(3, 1)	-1.36	-0.85	-0.02	-0.01	-0.29	-0.08	-1.48	-0.07
(3, 3)	(3, 2)	11.29	11.80	0.00*	0.01	-0.71	-0.50	-4.62	-3.21

¹⁾ (card allocation rule, dispatching rule); ERD (1), CS (2) and MODCS (3); ERD (1), SPT (2) and MERD (3)

²⁾ 95% confidence interval; * not significant at $\alpha=0.05$

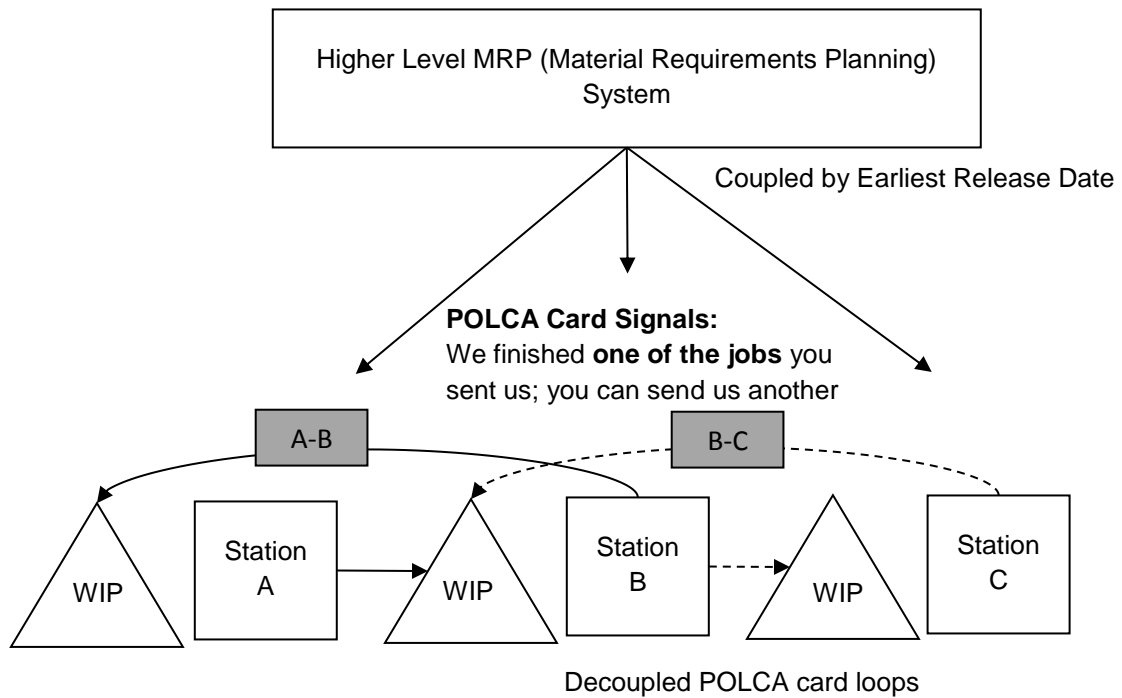


Figure 1: A POLCA System (with Decoupled POLCA Loops Coupled by an MRP system)

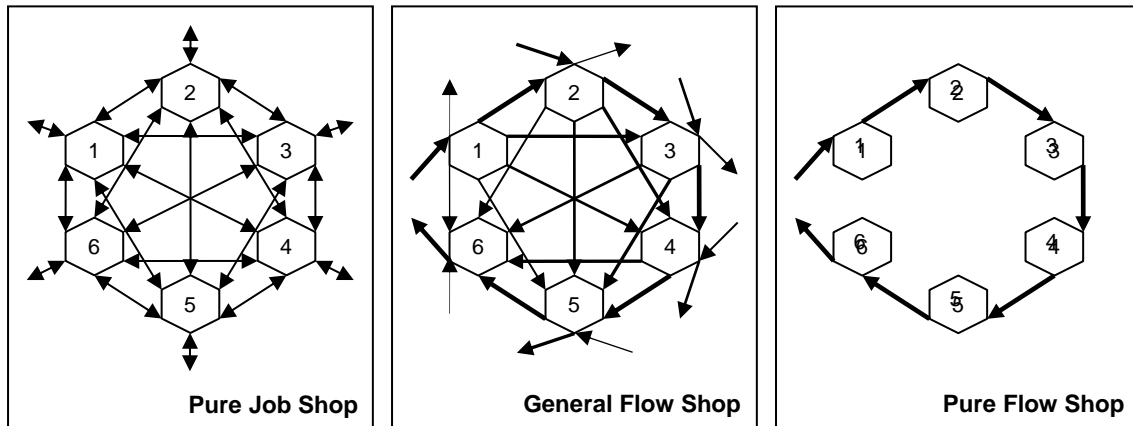


Figure 2: Illustration of The Three Key Types of Shop Floors According to Routing Characteristics (The Probability of Transition between Operations is indicated by the Strength of the Arrows)

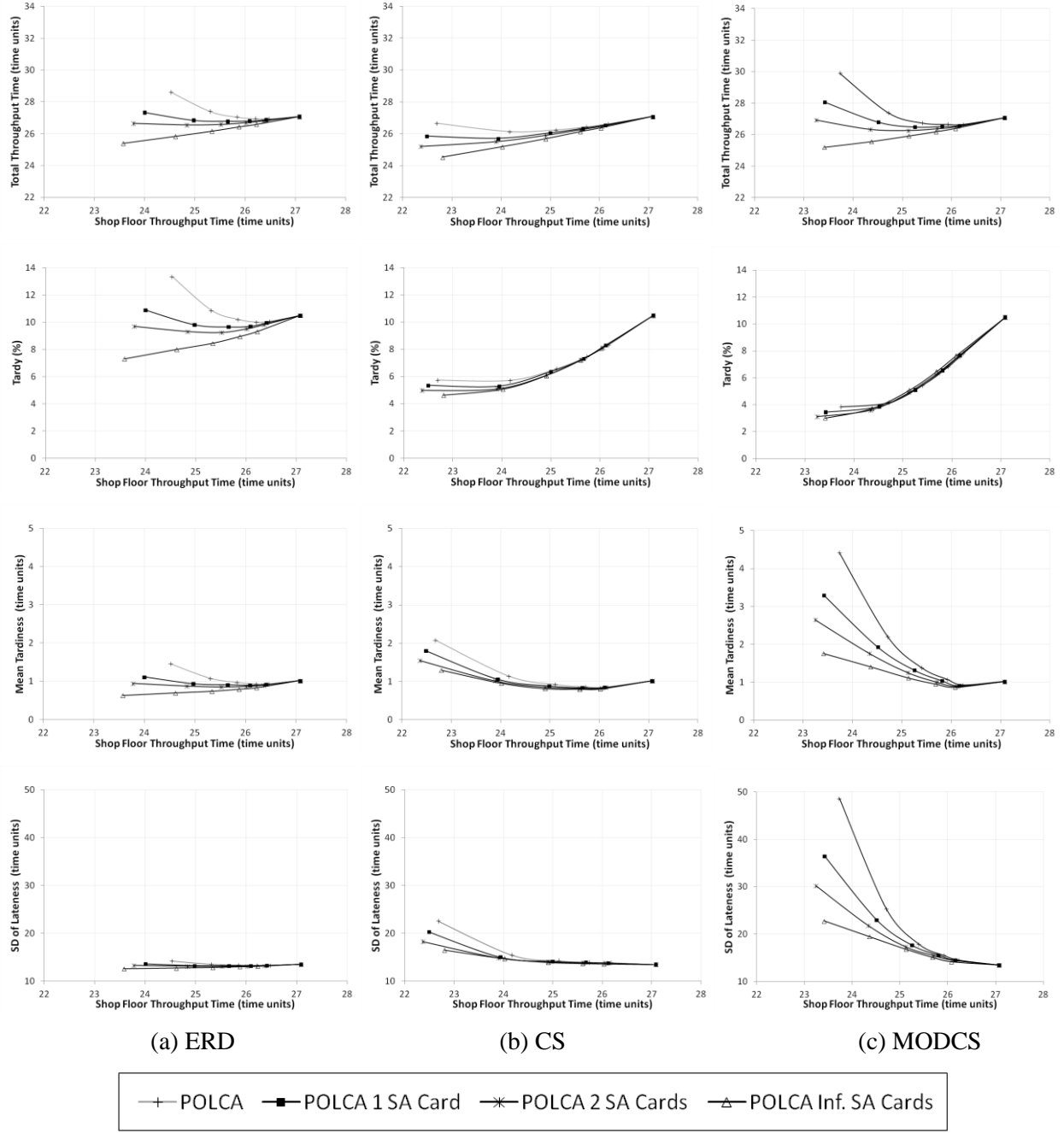


Figure 3: Performance Results for ERD Dispatching in Combination with the: (a) ERD; (b) CS; and, (c) MODCS Card Allocation Rule

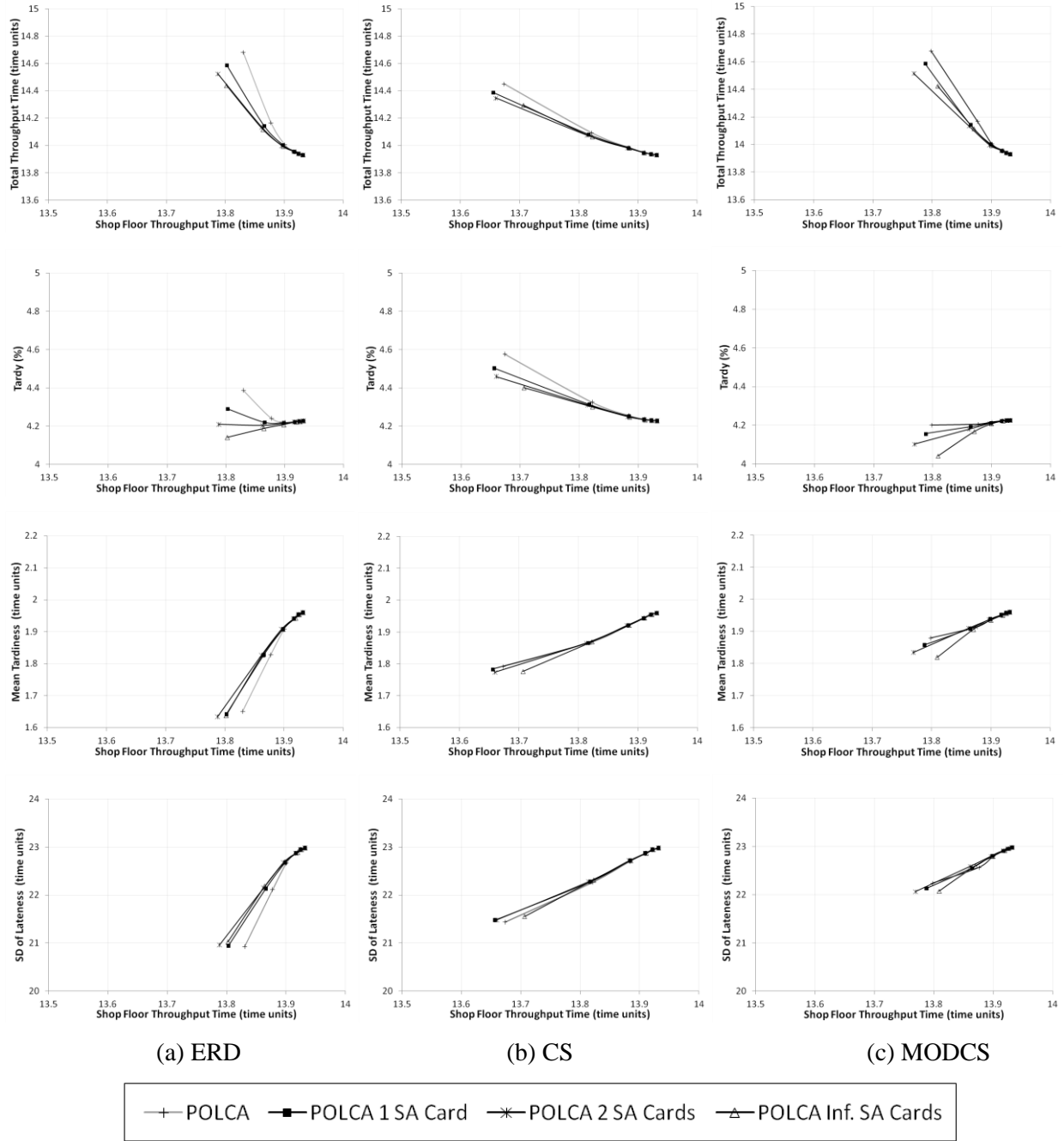


Figure 4: Performance Results for SPT Dispatching in Combination with the: (a) ERD; (b) CS; and, (c) MODCS Card Allocation Rule

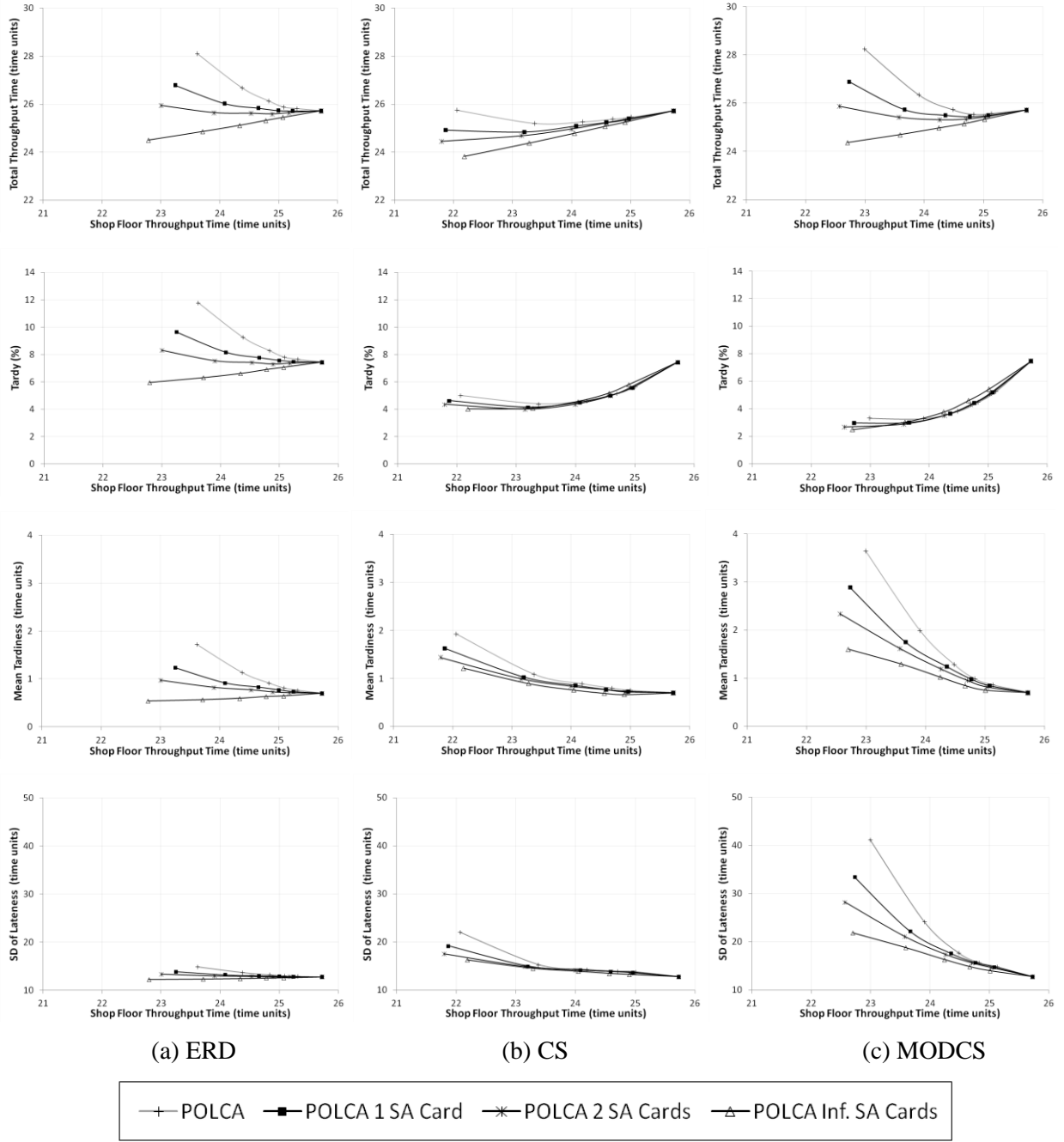


Figure 5: Performance Results for MERD Dispatching in Combination with the: (a) ERD; (b) CS; and, (c) MODCS Card Allocation Rule