Improving workload control order release: Incorporating a starvation avoidance trigger into continuous release

Nuno O. Fernandes\textsuperscript{a,d,*}, Matthias Thürer\textsuperscript{b}, Cristóvão Silva\textsuperscript{c}, Silvio Carmo-Silvad

\textsuperscript{a} Instituto Politécnico de Castelo Branco, Av. do Empresário, 6000-767 Castelo Branco, Portugal
\textsuperscript{b} Institute of Physical Internet, School of Electrical and Information Engineering, Jinan University (Zhuhai Campus), Zhuhai 519070, China
\textsuperscript{c} CEMUC research unit - University of Coimbra, Mechanical Engineering Department, Polo II Pinhal de Marrocos, 3030 Coimbra, Portugal
\textsuperscript{d} ALGORITMI Research Unit - University of Minho, Depart. of Production and Systems, Campus de Gualtar, 4710-057 Braga, Portugal

\textbf{ABSTRACT}

Order release is a key component of Workload Control - a production planning and control system that aims at balancing workloads across workstations, while ensuring the timely release of jobs (or orders) to the shop floor in order to meet due dates. Several release methods have been proposed and evaluated in the WLC literature. A major criterion to distinguish between release methods is whether they take the release decision at periodic time intervals or continuously. This paper aims at improving WLC order release by incorporating a starvation avoidance trigger into continuous release. Using simulation, we demonstrate that significant performance improvements in terms of mean tardiness and standard deviation of lateness can be obtained. These results are expected to have important implications for industrial practice and for future research on WLC.

\textbf{1. Introduction}

Global competition and changing customer requirements pose major challenges on small and medium sized companies that often produce-to-order. In order to remain competitive, companies need to simultaneously reduce the time it takes to manufacture and deliver products to customers, while realizing high delivery reliability. One means of achieving this is improved Production Planning and Control (PPC).

Workload Control (WLC) is a PPC system specifically designed for make-to-order production (Stevenson et al., 2005) that attempts to overcome the quandary of increasing delivery speed while maintaining and possibly improving delivery reliability. WLC is based on the concept of input/output control (Wight, 1970), where the input rate of work to the shop is controlled in accordance with the output rate. One of the key decision levels within WLC is order release (Land and Gaalman, 1996; Kingsman, 2000; Land, 2006). Jobs or orders are not directly released to the shop floor but withheld in a so-called pre-shop pool, from where they are released, i.e. moved into production, to meet certain performance targets. Order release aims to control the workload levels in the shop and balances the workload across workstations, thus reducing and stabilising shop floor throughput times. Stabilised shop floor throughput times, in turn, enable to quote shorter and more reliable delivery dates (Land and Gaalman, 1996; Breithaupt et al., 2002).

Several release methods have been proposed in the WLC literature (for a review see e.g., Land and Gaalman, 1996, Bergamaschi et al., 1997, Fredendall et al., 2010, Thürer et al., 2014). A major criterion to distinguish between release methods is whether they take the release decision - the decision on which jobs should enter production – at periodic time intervals or continuously, i.e. at any moment in time (Sabuncuoglu and Karapinar, 1999, Thürer et al., 2012). Periodic release methods typically apply an upper workload limit, i.e. jobs are only released if they fit a certain workload norm imposed across workstation (Bechte, 1988; Wiendahl, 1995; Perona and Portioli, 1998; Cigolini and Portioli-Staudacher, 2002; Land, 2006). This creates a mix of jobs released to the shop floor that levels the workload across resources. However, it suffers from two weaknesses: (i) periodic release does not react if the buffer of work in front of a workstation is depleted in between releases, leading to premature workstation idleness; and, (ii) the upper bound may introduce further premature workstation idleness (Kanet, 1988; Land and Gaalman, 1998), i.e. work is withheld in the pool (due to the workload situation at another workstation) although the workstation is starving. On the other hand, continuous release methods are typically based on the re-order point methodology and release new work to the shop floor whenever a certain lower limit is

\* Corresponding author at: Instituto Politécnico de Castelo Branco, Av. do Empresário, 6000-767 Castelo Branco, Portugal.

\textsuperscript{a} Corresponding author at: Instituto Politécnico de Castelo Branco, Av. do Empresário, 6000-767 Castelo Branco, Portugal.

\textsuperscript{b} Institute of Physical Internet, School of Electrical and Information Engineering, Jinan University (Zhuhai Campus), Zhuhai 519070, China

\textsuperscript{c} CEMUC research unit - University of Coimbra, Mechanical Engineering Department, Polo II Pinhal de Marrocos, 3030 Coimbra, Portugal

\textsuperscript{d} ALGORITMI Research Unit - University of Minho, Depart. of Production and Systems, Campus de Gualtar, 4710-057 Braga, Portugal

\textsuperscript{E-mail addresses:} nogf@ipcb.pt (N.O. Fernandes), matthiasthurer@workloadcontrol.com (M. Thürer), cristovao.silva@dem.uc.pt (C. Silva),
sarmo@dps.uminho.pt (S. Carmo-Silva).

http://dx.doi.org/10.1016/j.ijpe.2016.12.029
Received 13 April 2016; Received in revised form 11 October 2016; Accepted 21 December 2016
0925-5273/ © 2016 Elsevier B.V. All rights reserved.

Please cite this article as: Fernandes, N.O., International Journal of Production Economics (2016),
http://dx.doi.org/10.1016/j.ijpe.2016.12.029
violated (e.g., Melnyk and Ragatz, 1989; Hendry and Wong, 1994; Qi et al., 2009). While this avoids the weaknesses associated with periodic release, this kind of methods misses workload balancing capability, which is essential for realizing short and stable shop floor throughput times. Only recently a new type of continuous release methods emerged which applies an upper workload bound (see, e.g., Fernandes and Carmo-Silva, 2011a, 2011b, Thürer et al., 2014b). While these methods provide the load balancing capabilities of periodic release methods and overcomes the first weakness associated with periodic release, the issue of premature idleness caused by the upper workload bound remains unsolved.

Thürer et al. (2012b) recently addressed the issue of premature idleness in the context of periodic WLC release methods by combining periodic release for workload balancing with a continuous starvation avoidance workload trigger. This trigger mechanism releases available direct work (i.e., jobs with the first operation at the starving workstation) to a starving workstation regardless of whether or not the release would violate the workload norm at another workstation. The resulting release method - LUMS COR (Lancaster University Management School Corrected Order Release) - significantly improved performance compared to purely periodic release. In this study we base on Thürer et al. (2012b) to outline two new release methods that combine Continuous Release (CR) with Starvation Avoidance (SA). These methods use the starvation avoidance trigger to avoid premature workstation idleness, in combination with the new breed of continuous release methods that uses an upper workload bound for workload balancing. Using simulation, we then assess the performance impact of these two new methods compared to continuous release and LUMS COR in a pure job shop.

The remainder of this paper is organized as follows. Section 2 reviews order release methods and outlines our new order release method. Section 3 then presents the simulation model and the experimental design used to evaluate the performance of our new methods. In Section 4, the simulation results are presented and analysed. Finally, conclusions are summarized and future work discussed in Section 5.

2. Literature review and development of new release method

This section is structured as follows. Section 2.1 first introduces periodic release methods before continuous release methods are introduced in Section 2.2. Literature is then assessed and the new release methods introduced in Section 2.3.

2.1. Periodic order release

Periodic WLC release methods keep the workload $W_i^s$ released to a workstation $s$ within limits or norms by selectively releasing jobs. While different periodic release methods have been presented in the literature, the release procedure executed across the methods is similar. It can be formulated as follows:

1. All jobs in the set of jobs $J$ in the pre-shop pool are sorted according to a given pool sequencing rule.
2. The job $j$ with the highest priority is considered for release first.
3. Take $R_j$ to be the ordered set of operations in the routing of job $j$. If job $j$'s processing time $p_{ij}$ at the $ith$ operation in its routing - corrected for workstation position $i$ - together with the current workload $W_i^s$ at workstation $s$ (corresponding to operation $i$) fits within the workload norm $N_i^s$ at this workstation, that is $p_{ij} + W_i^s \leq N_i^s \forall i \in R_j$, then the job is selected for release. That means the job is removed from $J$ and its load contribution is added to the existing workload of workstation $s$, i.e. $W_i^s = W_i^s + \sum_{i \in R_j} p_{ij} \forall i \in R_j$. Otherwise, the job remains in the pool and its processing time does not contribute to the workstation workload.
4. If the set of jobs $J$ in the pool still contains any jobs that have not yet been considered for release, then return to Step 2 and consider the job with the next highest priority. Otherwise, the release procedure is complete.

A released job contributes to $W_i^s$ until its operation at this workstation $s$ is completed. The main difference between periodic release methods is on how the workload is calculated (Land and Gaalman, 1996; Cigolini and Portioli-Staudacher, 2002). Early studies on Workload Control often calculated the (aggregate) load of a station as the sum of all the processing times of jobs released but not yet completed by a station (Bertrand and Wortmann, 1981; Tatsiopoulos, 1983; Hendry and Kingsman, 1991). But this ignored the fact that the amount of work still upstream (indirect load) may vary, depending on the position of a station in the routing of jobs. The corrected aggregate workload as presented by Oosterman et al. (2000) corrects the load contribution to a station by dividing the processing time of the operation at a station by the station's position in the job's routing. This recognizes that an order's contribution to a station's direct load is limited to only the proportion of time that an order is at the station. For example, an order's load contribution at the second station in its routing is set at 50% of the processing time at this station; similarly, its load contribution at the third station is set at 33.33%, and so on. Oosterman et al. (2000) demonstrated that this provides a good estimate of the expected average direct load resulting from a release decision leading to better performance than the classical aggregate or probabilistic approach (Bechte, 1988; Wiendahl, 1993).

2.2. Continuous order release

In contrast to periodic methods, 'classical' continuous order release methods do not apply a workload norm (upper bound); instead, a workload trigger is used. For classical continuous release methods, a critical workload (lower bound) is determined, which, if violated, triggers the release procedure, thereby pulling jobs from the pool onto the shop floor until the critical workload is no longer violated. This may allow the next job to be selected even if its workload means the critical workload is exceeded, i.e. there is no maximum workload constraint. Order release methods of this type can best be classified according to the workload used to trigger the release: bottleneck, workstation or shop load.

A different type of continuous release methods that apply a workload norm (upper bound) has recently been presented in the literature (see, e.g., Fernandes and Carmo-Silva, 2011a, 2011b). These release methods execute the release procedure described in Section 2.1 continuously, whenever a new job arrives to the system or an operation is completed. er et al., (2014a) recently demonstrated the potential of this new breed of continuous release methods to outperform 'classical' continuous release methods.

2.3. Assessment of the literature and new release methods

Periodic WLC release methods (as described in Section 2.1) provide unique workload balancing capabilities that makes them suitable for high variety contexts. However, they suffer from two weaknesses: (i) periodic release does not react if the buffer of work in front of a workstation is depleted in between periodic releases leading to premature workstation idleness; and, (ii) the upper bound may introduce further premature workstation idleness. None of these weaknesses applies to 'classical' continuous release methods as described in Section 2.2, but these 'classical' continuous release methods lack load balancing capabilities (Germis and Riezebos, 2010; Thürer et al., 2012a). The new breed of continuous release methods that uses an upper workload limit provides load balancing capabilities and has been shown to outperform 'classical' continuous release methods (Thürer et al., 2012a). However, while these methods address premature idleness caused by the periodicity of releases they still suffer
دریافت فوری متن کامل مقاله

امکان دانلود نسخه تمام متن مقالات انگلیسی
امکان دانلود نسخه ترجمه شده مقالات
پذیرش سفارش ترجمه تخصصی
امکان جستجو در آرشیو جامعی از صدها موضوع و هزاران مقاله
امکان دانلود رایگان ۲ صفحه اول هر مقاله
امکان پرداخت اینترنتی با کلیه کارت های عضو شتاب
دانلود فوری مقاله پس از پرداخت آنلاین
پشتیبانی کامل خرید با بهره مندی از سیستم هوشمند رهگیری سفارشات