Performance analysis of fluid flow production lines with finite buffers and generally distributed up and downtimes

Citation for published version (APA):

Document status and date:
Published: 01/01/2011

Document Version:
Publisher’s PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:

• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher’s website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the “Taverne” license above, please follow below link for the End User Agreement:
www.tue.nl/taverne

Take down policy
If you believe that this document breaches copyright please contact us at:
openaccess@tue.nl
providing details and we will investigate your claim.

Download date: 12. Dec. 2019
In this paper we analyze continuous-flow production lines consisting of a number of machines or servers in series and a finite buffer between each pair of machines. Each machine suffers from breakdowns, for instance because of failures, cleaning, changeover, etcetera. The resulting up- and downtime periods are generally distributed. Because of breakdowns and non-identical machine speeds, the machines are influenced by each other, resulting in starvation, blocking, and speed adaption. We are interested in throughput and mean buffer content of this production line. The newly developed method relies on decomposition of the production line into two-stage, one-buffer subsystems. The parameters for each subsystem are determined iteratively.

A crucial feature of the model is that we deal with generally distributed up- and downtimes. The use of aggregation techniques on the subsystem level is important to be able to analyze larger production lines of up to 16 machines. The proposed method performs very well on a large test set consisting of over 49,000 cases. Remarkably, the performance of the method does not deteriorate for cases with high variation in up- and downtimes, making this model suitable for highly unpredictable up- and downtimes as often seen in practice.

Key words: production line; finite buffer; fluid flow; approximation; decomposition

1. Introduction

This paper deals with production lines consisting of a number of machines in series. Each pair of machines is separated by a buffer of fixed size. The flow through the machines is continuous. Figure 1 shows an example of a four-machine production line, where $M_i$ is the $i$th machine and $B_i$ is the $i$th buffer. Buffer $B_i$ has a size of $b_i$ and machine $M_i$ produces at a maximum speed of $s_i$ per time unit. Since we do not assume the maximum speeds to be equal, machines adjust their speeds constantly in case of empty or full buffers. Furthermore, each machine suffers from breakdowns, after which a period of repair follows. During this period, the machine under repair is not able to produce and it possibly affects other machines in the form of starvation of downstream machines and blocking of upstream machines. For each machine, we construct a cycle consisting of a breakdown- or uptime, followed by a repair- or downtime. The length of the up- and downtime periods are assumed to be independent and generally distributed. The length of an uptime period $U_i$ of $M_i$ is characterized by rate $\lambda_{U_i}$ and squared coefficient of variation (scv) $c^2_{U_i}$, where the scv is defined as the variance divided by the squared mean. The length of a downtime period $V_i$ is characterized by rate $\lambda_{V_i}$ and scv $c^2_{V_i}$. These rates and scv’s can be obtained directly from industrial data. We assume that a machine cannot break down when it is not producing because of starvation or blocking. This assumption is called "operationally dependent failures" throughout this paper. Besides this assumption, we do not assume any relationship between speeds and breakdown rates,
since we found that such relationships do not exist for practical cases under our investigation. Since these type of production lines are too complex to analyze exactly, we aim to find a reliable and robust analytical approximation.

There is a huge literature on production lines with continuous flows. The idea of decomposition to analyze production lines was firstly introduced by Gershwin (1987). This idea was extended in several other papers, e.g. by Dallery et al. (1988), Burman (1995), and Bierbooms et al. (2010). Levantesi et al. (2003) take into account general up- and downtime distributions by assuming phase-type distributed up- and downtimes. Although this approximation performs well, it is unable to analyze larger production lines because of a too large state space. Another approach is the use of homogenization methods, see e.g. Dallery and Bihan (1997). In these methods, a non-homogeneous production line with non-identical machine speeds is replaced by an equivalent homogeneous line.

The novelty of our approach is that it is able to analyze longer production lines with generally distributed up- and downtimes. The new method relies on decomposition of the production line into subsystems, each subsystem consisting of an "arrival server", a "departure server", and a buffer in between. The arrival server describes the behavior of the upstream part of the production line and the departure server describes the behavior of the downstream part. The parameters for the arrival and departure machines are determined iteratively. The key to our approach is the use of aggregation techniques: instead of keeping the state of the whole upstream or downstream part of an arrival or departure server, we use aggregation to avoid a state space explosion for longer production lines.

In Section 2, we decompose the production line into two-machine, one-buffer subsystems and we define the elements of a subsystem. Section 3 constructs the iterative method to obtain the throughput and mean total buffer content of the production line as a whole. In Section 4, we analyze a subsystem by going through the steps of the iterative algorithm. Finally, Section 5 is devoted to results and discussion.

2. Decomposition

We decompose the production line $L$ into two-stage subsystems, as illustrated in Figure 2. Each subsystem $L_i$ consists of arrival server $A_i$, buffer $B_i$, and departure server $D_i$.

In the description of the arrival server we include the influence of the upstream part of the production line on original machine $M_i$, comprising starvation and speed adaption. Arrival server $A_i$ can be up, down, or starved if $A_{i-1}$ is down or starved and $B_{i-1}$ is empty. Similarly, in the description of the departure server we include the influence of blocking and speed adaption caused by the downstream part of the production line on original machine $M_{i+1}$. Departure server $D_i$ can be up, down, or blocked if $D_{i+1}$ is down or blocked and $B_{i+1}$ is full.

We describe server $A_i$ as a continuous-time Markov chain with $k_A^{(i)}$ states and generator $Q_A^{(i)}$. Each state of the Markov chain has a corresponding production speed. The $j$th element of column vector $r_A^{(i)}$ denotes the speed in state $j \leq k_A^{(i)}$. Server $D_i$ is modeled as a continuous-time Markov chain with $k_D^{(i)}$ states, generator $Q_D^{(i)}$, and speed vector $r_D^{(i)}$. The challenge is to determine the structure of the Markov chains and the elements of $Q_A^{(i)}$, $Q_D^{(i)}$, $r_A^{(i)}$, and $r_D^{(i)}$, which will be done in an iterative way. In the next section, we present the iterative method to obtain these elements, and ultimately, the throughput of the system.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{production_line.png}
\caption{A production line $L$ with four servers, labeled $M_1$ up to $M_4$.}
\end{figure}
3. Iterative method

In this section, we construct an iterative method to obtain the throughput and mean buffer content distribution of an $N$-stage production line $L$. This algorithm relies on the decomposition into subsystems $L_1, \ldots, L_{N-1}$ as explained in the previous section.

**Step 0:** Initialize characteristics. We assume that initially $A_i$ is not affected by starvation and $D_i$ is not affected by blocking. The corresponding parameters are set accordingly.

**Step 1:** For each subsystem $L_i$, $i = 1, \ldots, N-1$:

(a) We construct a continuous-time Markov chain describing the behavior of $A_i$. The elements of $Q_A^{(i)}$ and $r_A^{(i)}$ corresponding to this Markov chain are determined. Similarly, we construct a Markov chain for $D_i$ and we determine the elements of $Q_D^{(i)}$ and $r_D^{(i)}$. This step is explained in detail in Subsection 4.1.

(b) We merge the Markov chains for $A_i$ and $D_i$ into another Markov chain with $k^{(i)} = k_A^{(i)} \times k_D^{(i)}$ states, generator $Q^{(i)}$, and net speed vector $r^{(i)}$. Next, we determine the steady state distribution of $Q^{(i)}$. The pdf $f_j^{(i)}(x)$ is the density of the subsystem in state $j \in S^{(i)}$ and at buffer level $x$, $0 \leq x \leq b_i$. We elaborate further on this step in Subsection 4.2.

(c) Using the steady state distribution of $Q^{(i)}$ from the previous step, we obtain the estimated throughput $T_h^{(i)}$ of subsystem $L_i$. The subscript $h$ means that the estimate is obtained from iteration $h$.

(d) If $i < N-1$, we obtain the rate at which $A_{i+1}$ goes starved and the rate and coefficient of variation of the duration of a starvation period for $A_{i+1}$. We also determine the average speed $s_A^{(i+1)}$ at which $A_{i+1}$ is producing, whenever it is actually generating output. If $i > 1$, we update the rate at which $D_{i-1}$ goes blocked and the rate and coefficient of variation of the duration of a blocking period for $D_{i-1}$. Also for $D_{i-1}$, we determine the average production speed $s_D^{(i-1)}$. This step is explained in Subsection 4.3.

**Step 2:** We repeat step 1 until the throughput for all subsystems has converged. If for some small $\epsilon$ it holds that

$$\frac{T_h^{(i)}}{T_h^{(i-1)}} < \epsilon$$

for all $i \leq N-1$, we stop, otherwise we do another iteration.

Note that in this algorithm it is not guaranteed that the throughput values for all subsystems are equal. However, this appeared to be true in all our experiments. In the next section, we explain the steps of the iterative algorithm in more detail.
4. Subsystem analysis

This section describes the analysis of a subsystem $L_i, i = 1, ..., N - 1$, by consecutively going through steps 1(a)-1(d) of the iterative method in detail. Firstly, Subsection 4.1 models the behavior of arrival server $A_i$ and departure server $D_i$ as continuous-time Markov chains. Subsection 4.2 is devoted to the determination of the steady state distribution of the subsystem. From this distribution, we update parameters for the arrival server of subsystem $L_{i+1}$ and the departure server of $L_{i-1}$ in Subsection 4.3.

4.1. Behavior of arrival and departure server

In this subsection, we model the behavior of arrival machine $A_i$ and departure machine $D_i$ as a continuous-time Markov chain. For $A_i$, we divide the state space into three sets: states where $A_i$ is producing (up), states where $A_i$ is not producing because of a breakdown of the underlying machine $M_i$ (down), and states where $A_i$ is not producing because it has no input (starved). These three sets are formally defined as follows.

- The set of up-states $S_{A,u}^{(i)}$: $A_i$ is up when $M_i$ is up, and either $B_{i-1}$ is not empty or $B_{i-1}$ is empty and $A_{i-1}$ is up.
- The set of down-states $S_{A,d}^{(i)}$: $A_i$ is down when $M_i$ is down.
- The set of starved-states $S_{A,st}^{(i)}$: $A_i$ is starved when $M_i$ is up, $B_{i-1}$ is empty, and $A_{i-1}$ is down or starved.

We define $Q_{A}^{(i)}$ as the generator of the Markov process for $A_i$, containing the transition rates within and between the three sets of states. The state space of this Markov chain is given by $S_{A}^{(i)} = S_{A,u}^{(i)} \cup S_{A,d}^{(i)} \cup S_{A,st}^{(i)}$. Because of operationally dependent failures, no transitions are possible from the starved-states to the down-states. Transitions in the opposite direction are also not possible.

To obtain the remaining transition rates, we fit phase-type distributions on one or two moments of the following random variables (see e.g. Tijms (1994)):

- The breakdown time of $M_i$ with rate $\lambda_{U_i}$ and coefficient of variation $c_{U_i}$.
- The repair time of $M_i$ with rate $\lambda_{V_i}$ and coefficient of variation $c_{V_i}$.
- The duration of a starvation period with rate $\lambda_{SU_i}$ and coefficient of variation $c_{SU_i}$. These parameters are determined from the analysis of $L_{i+1}$ in the previous iteration (see Subsection 4.4).
- The rate $\lambda_{US}^{(i)}$ at which $A_i$ is jumping to the starved-state, determined from the analysis of subsystem $L_{i+1}$. Since we do not have the coefficient of variation, we fit an exponential distribution on this variable.

With these phase-type distributions we can obtain the elements of $Q_{A}^{(i)}$.

**Remark:** Because of operationally dependent failures, the breakdown time is "freezed" during a starvation period. We solve this by storing the phase of the breakdown distribution $U_i$ in a starvation period, continuing in this phase whenever $A_i$ jumps back to the up-state again.

A similar analysis can be applied to departure machine $D_i$. The role of blocking for $D_i$ is symmetrical to the role of starvation for $A_i$.

4.2. Steady state distribution

This subsection is devoted to the determination of the steady state distribution of subsystem $L_i$. First, we drop the subscript $i$ and superscript $(i)$ referring to the $i$th subsystem. Using the Markov chains for arrival machine $A$ and departure machine $D$, we construct another Markov chain describing the behavior of the whole subsystem. This Markov chain has state space $S = S_A \cup S_D$, generator $Q$, and net speed vector $r$. The number of states is given by $k = k_A \times k_D$. By ordering the states of $A$ and $D$ lexicographically, we obtain
rates in $Q$. We obtain

$$Q = Q_A \otimes I_{k_D} + I_k \otimes Q_D,$$

$$r = r_A \otimes 1_{k_D} - 1_k \otimes r_D,$$

where $I_n$ is an identity matrix of size $n \times n$ and $1_n$ is a column vector of ones of size $n$.

Because of our assumptions, $A$ cannot go down or starved whenever $B$ is full and $D$ is not producing, and $D$ cannot go down or blocked whenever $B$ is empty and $A$ is not producing. This implies that some of the transition rates are different in case of an empty or full buffer. Therefore, we define a full-buffer process with generator $Q^F$ and an empty-buffer process with generator $Q^E$.

Starting with $Q^F$, we argue that when $D$ is in a state with zero-speed and $B$ is full, $A$ cannot jump to a state with zero-speed. In all other situations, the transition rates in $Q^F$ are the same as the rates in $Q$. We obtain

$$Q^F_{(i_A,j_D) \rightarrow (j_A,j_D)} = \begin{cases} 0 & \text{if } r_{jA} = 0, r_{jD} = 0, \\ Q^F_{(i_A,j_D) \rightarrow (j_A,j_D)} & \text{else.} \end{cases}$$

Similarly, when $A$ is in a state with zero-speed and $B$ is empty, $D$ cannot jump to a state with zero-speed. This gives

$$Q^E_{(i_A,j_D) \rightarrow (j_A,j_D)} = \begin{cases} 0 & \text{if } r_{jD} = 0, r_{jA} = 0, \\ Q^E_{(i_A,j_D) \rightarrow (j_A,j_D)} & \text{else.} \end{cases}$$

The state of the subsystem can be described by the pair of variables $(i, x)$, where $i \in S$ is the state of the phase process and $0 \leq x \leq b$ is the fluid level of the buffer. We define $f_i(x)$ as the probability density function in state $(i, x)$. Since the buffer can be full or empty, we have probability mass at the boundary levels $0$ and $b$. We define $p_i^{(0)}$ as the probability of being in state $(i,0)$, and $p_i^{(b)}$ as the probability of being in state $(i,b)$.

The steady state distribution of $Q$ can be determined by solving a set of linear differential equations, which can be done by using numerically stable matrix-analytic techniques (see Soares and Latouche (2005)).

4.3. Update parameters

In this subsection, we obtain output parameters from subsystem $L_i$ that are used for the analysis of $L_{i+1}$ and $L_{i-1}$. For this, we use the following information obtained from the steady state distribution of subsystem $L_i$:

- $\Pi_{i,j}^{(i)}$, a matrix of steady state probabilities, the $(m,n)$th element of which is the probability that $A_i$ is in state $m \in S_{\lambda}^{(i)}$, $j \in \{u,d,sl\}$, and $D_i$ is in state $n \in S_{\lambda}^{(i)}$, $l \in \{u,d,bl\}$. The number $\pi_{j,i}$ is the sum of all probabilities in $\Pi_{i,j}^{(i)}$.

- $P_{j,i}^{(i)}(0)$ and $P_{j,i}^{(i)}(b)$, matrices of boundary probabilities at the levels $0$ and $b$. The subscript $(j,l)$ has the same interpretation as for $\Pi_{i,j}^{(i)}$. The numbers $p_{j,i}^{(i)}(0)$ and $p_{j,i}^{(i)}(b)$ are the sum of probabilities in $p_{j,i}^{(i)}(0)$ and $p_{j,i}^{(i)}(b)$ respectively.

- $F_{j,i}^{(i)}(0)$ and $F_{j,i}^{(i)}(b)$, pdf matrices at the levels $0$ and $b$. The numbers $f_{j,i}^{(i)}(0)$ and $f_{j,i}^{(i)}(b)$ are the sum of elements in $F_{j,i}^{(i)}(0)$ and $F_{j,i}^{(i)}(b)$ respectively. We can also see $F_{j,i}^{(i)}(0)$ and $F_{j,i}^{(i)}(b)$ as the number of level crossings at level $0$ and $b$ in state $(j,l)$, $j \in \{u,d,sl\}$, $l \in \{u,d,bl\}$.

We start with the determination of starvation parameters for $A_{i+1}$: the rate $\Lambda_{i+1}^{(i)}$ at which $A_{i+1}$ goes starved and the rate $\lambda_i^{(i+1)}$ and coefficient of variation $c_i^{(i+1)}$ of the length of a starvation period for $A_{i+1}$. These parameters are used for the generator $Q_{i}^{(i+1)}$ (see Subsection 4.2). Recall that $A_{i+1}$ is in the starved state when $M_{i+1}$ is up, $A_i$ is down or starved and $B_i$ is empty. A jump from the up-state to the starved-state can be caused by either of the following two events (note that the underlying machine of departure server $D_i$ is also $M_{i+1}$):
1. First, $A_i$ is down or starved, $D_i$ is up, and $B_i$ is non-empty. Then $D_i$ will take the fluid out of the buffer until eventually $B_i$ becomes empty, and thus, $A_{i+1}$ gets starved.

2. First, $A_i$ is up, $D_i$ is up, and $B_i$ is empty. Then $A_i$ goes down or starved, and simultaneously, $A_{i+1}$ gets starved.

To obtain $\lambda_{US}^{(i+1)}$, we add up the number of type-(1) jumps and the number of type-(2) jumps per time unit. Using this argument, it follows that $\lambda_{US}^{(i+1)}$ is given by
\[
\lambda_{US}^{(i+1)} = \frac{(f_{d,u}^{(i)}(0) + f_{st,u}^{(i)}(0))s_D^{(i)}}{\pi_{u,u}^{(i)} + \pi_{d,u}^{(i)} - p_{d,u}^{(i)}(0) + \pi_{st,u}^{(i)} - p_{st,u}^{(i)}(0)} + \frac{(P_{u,u}^{(i)}(0)1)'(Q_{A,u}^{(i)}Q_{A,u}^{(i)})}{\pi_{u,u}^{(i)} + \pi_{d,u}^{(i)} - p_{d,u}^{(i)}(0) + \pi_{st,u}^{(i)} - p_{st,u}^{(i)}(0)},
\]
where $1$ is a column vector of ones of appropriate size.

To determine the rate and coefficient of variation of a starvation period for $A_{i+1}$, we first introduce the following concept. Suppose that random variable $Y$ can be described as the time until absorption in a Markov process with $k_Y$ non-absorbing states and one absorbing state. The $k_Y \times k_Y$ matrix of transitions between the non-absorbing states is denoted by $\Gamma_Y$. Consequently, $(I - \Gamma_Y)1$ is the (column) vector of transitions from the non-absorbing states to the absorbing state. The Markov process does not necessarily start in the first (or any) state with probability 1. Instead, we define $\alpha_Y$ as the starting probability vector, the $j$th element of which is the probability of starting in state $j \leq k_Y$. The sum of the elements in $\alpha_Y$ should be 1, which implies that the process starts in the non-absorbing states. The $n$th moment of the time until absorption is given by (see e.g. Latouche and Ramaswami (1999))
\[
E(Y^n) = (-1)^n n! \alpha_Y \Gamma_Y^{-n} 1.
\]

Using this, we can calculate the rate $\lambda_Y = 1/E(Y)$ and coefficient of variation $c_Y = (E(Y^2) - E^2(Y))/E^2(Y)$ of the random variable $Y$. In this way we can describe the duration of a starvation period of $A_{i+1}$. This period ends whenever $A_i$ jumps back up from either a down-state or a starved-state. Thus, we can model the duration of a starvation period as a Markov process with the down-states and starved-states of $A_i$ being the non-absorbing states and the set of up-states being the absorbing state. The transition matrix $\Gamma_{SU}^{(i+1)}$ is given by
\[
\Gamma_{SU}^{(i+1)} = \left( \begin{array}{cc} Q_{A,d,d}^{(i)} & 0 \\ 0 & Q_{A,ut}^{(i)} \end{array} \right).
\]

Furthermore, we have to determine the probabilities of starting the starvation period in either of the down- and starved-states of $A_i$. The starting probability vector $\alpha_{SU}^{(i+1)}$ is given by
\[
\alpha_{SU}^{(i+1)} = \xi_{SU}^{(i+1)} \left( \begin{array}{c} F_{d,u}^{(i)}(0) \\ F_{st,u}^{(i)}(0) \end{array} \right)'s_D^{(i)} + \left( P_{u,u}^{(i)}(0)1 \right)' \left( Q_{A,u}^{(i)}Q_{A,u}^{(i)} \right),
\]
where $s_{SU}^{(i+1)}$ is a normalization constant, chosen such that the sum of $\alpha_{SU}^{(i+1)}$ is equal to one. Note the similarity between this expression and (1).

Next, we determine the average speed at which $A_{i+1}$ produces whenever it is up. If the maximum speed $s_{i+1}$ of $M_{i+1}$ is lower than the average speed $s_A^{(i)}$ of $A_i$, then $A_{i+1}$ can always produce at its maximum speed whenever up and $s_A^{(i+1)} = s_{i+1}$. If this is not the case, then $A_{i+1}$ has to adjust its speed to $s_A^{(i)}$ when $B_i$ is empty and $A_i$ is up. The (conditional) fraction of time that the case is given by $p_{u,u}^{(i)}(0)/\left(\pi_{u,u}^{(i)} + \pi_{d,u}^{(i)} - p_{d,u}^{(i)}(0) + \pi_{st,u}^{(i)} - p_{st,u}^{(i)}(0)\right)$. The resulting expression for $s_A^{(i+1)}$ can be obtained as
\[
s_A^{(i+1)} = \left\{ \begin{array}{ll} s_{i+1} & \text{if } s_{i+1} \leq s_A^{(i)} \\ s_{i+1} - \pi_{u,u}^{(i)} + \pi_{d,u}^{(i)} - p_{d,u}^{(i)}(0) + \pi_{st,u}^{(i)} - p_{st,u}^{(i)}(0)(s_{i+1} - s_A^{(i)}) & \text{if } s_{i+1} > s_A^{(i)} \end{array} \right.
\]

We can obtain expressions for departure server $D_{i-1}$ of subsystem $L_{i-1}$ in a symmetrical way.
5. Results

In this section we investigate the quality of the proposed method. We compare our method to the approximation in Bierbooms et al. (2010). This method uses exponential distributions for transitions from and to breakdown times, starvation times, and blocking times. Thus, it neglects the second moment or coefficient of variation of these transitions.

We test the performance of our method on a large test set, in which we vary the values of seven input parameters: the number of machines in the line, mean uptimes, mean downtimes, squared coefficients of variation ($c^2$) of uptimes, $c^2$'s of downtimes, machine speed configuration, and buffer sizes. Table 1 lists the different settings for each input parameter. By making all combinations of these setting, we obtain a test set of $4 \times 4 \times 4 \times 4 \times 8 \times 3 \times 4 = 49,152$ cases. As can be seen in Table 1, cases are included with imbalance in mean up- and downtimes, squared coefficients of variation of up- and downtimes, and machine speeds. For the speed setting $\{15,...,10,...,15\}$, speeds decrease linearly in the first part of the production line, and speeds increase linearly in the second part of the production line. For instance, in a 6-machine production line the machine speeds would be $\{15,12.5,10,10,12.5,15\}$.

<table>
<thead>
<tr>
<th>Input parameter</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of machines</td>
<td>4, 8, 12, 16</td>
</tr>
<tr>
<td>Mean uptimes</td>
<td>{10,10,10,...}, {10,5,10,5,...}, {20,20,20,...}, {20,10,20,...}</td>
</tr>
<tr>
<td>Mean downtimes</td>
<td>{1,1,1,...}, {1,0.5,1,0.5,...}, {2,2,2,...}, {2,1,2,...}</td>
</tr>
<tr>
<td>Squared coefficient of variation of uptimes</td>
<td>{0.5,0.5,0.5,...}, {0.5,1,0.5,...}, {2,2,2,...}, {4,4,4,...}</td>
</tr>
<tr>
<td>Squared coefficient of variation of downtimes</td>
<td>{0.5,0.5,0.5,...}, {0.5,1,0.5,...}, {2,2,2,...}, {4,4,4,...}</td>
</tr>
<tr>
<td>Machine speeds</td>
<td>{10,10,10,...}, {10,15,10,...}, {15,10,15,...}, {15,...,10,...,15}</td>
</tr>
<tr>
<td>Buffer size</td>
<td>{1,1,1,...}, {10,10,10,...}, {25,25,25,...}, {50,50,50,...}</td>
</tr>
</tbody>
</table>

We test the performance of our method by comparing the approximated throughput and mean total buffer content to the same quantities obtained from a simulation model. The 95% confidence intervals in this simulation have a width of at most 0.5%. In Tables 2-8, we show the average relative errors of our approximation (column "Model PHT") and the approximation in Bierbooms et al. (2010) (column "Model EXP"). Each table row provides the average relative errors over all cases in which the input parameters have the values as specified in the first column. For instance, the first row of Table 2 gives the average errors over all 12,288 cases with four machines in the production line.

<table>
<thead>
<tr>
<th>Line length</th>
<th>Error (%) in the throughput</th>
<th>Error (%) in mean buffer content</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Model PHT</td>
<td>Model EXP</td>
</tr>
<tr>
<td>4</td>
<td>0.57</td>
<td>2.00</td>
</tr>
<tr>
<td>8</td>
<td>1.03</td>
<td>4.37</td>
</tr>
<tr>
<td>12</td>
<td>1.65</td>
<td>6.30</td>
</tr>
<tr>
<td>16</td>
<td>2.18</td>
<td>7.91</td>
</tr>
</tbody>
</table>

In Table 2 we see that our model is able to give reliable throughput estimates for production lines of up to 16 machines. The error in mean buffer content even decreases for longer production lines. Table 3 shows that our approximation is less sensitive to the squared coefficient of variation of uptimes than the model in Bierbooms et al. (2010). In Table 4 it appears that our model is nearly insensitive to the squared coefficient of variation of downtimes, a feature which is definitely not shared by the model in Bierbooms et al. (2010). Therefore we can conclude that our model is able to handle with big and small variations in up- and downtimes.
Table 3 Results for production lines with different squared coefficients of variation of uptimes

<table>
<thead>
<tr>
<th>$C^2$ of uptimes</th>
<th>Error (%) in the throughput</th>
<th>Error (%) in avg buffer content</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Model PHT</td>
<td>Model EXP</td>
</tr>
<tr>
<td>0.5,0.5,0.5,0.5,...</td>
<td>1.09</td>
<td>4.06</td>
</tr>
<tr>
<td>0.5,1,0.5,1,...</td>
<td>1.12</td>
<td>4.23</td>
</tr>
<tr>
<td>1,1,1,...</td>
<td>1.36</td>
<td>4.59</td>
</tr>
<tr>
<td>1,2,1,2,...</td>
<td>1.29</td>
<td>4.83</td>
</tr>
<tr>
<td>2,2,2,2,...</td>
<td>1.44</td>
<td>5.26</td>
</tr>
<tr>
<td>2,4,2,4,...</td>
<td>1.37</td>
<td>5.62</td>
</tr>
<tr>
<td>4,4,4,4,...</td>
<td>1.62</td>
<td>6.12</td>
</tr>
<tr>
<td>4,8,4,8,...</td>
<td>1.57</td>
<td>6.45</td>
</tr>
</tbody>
</table>

Table 4 Results for production lines with different squared coefficients of variation of downtimes

<table>
<thead>
<tr>
<th>$C^2$ of downtimes</th>
<th>Error (%) in the throughput</th>
<th>Error (%) in avg buffer content</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Model PHT</td>
<td>Model EXP</td>
</tr>
<tr>
<td>0.5,0.5,0.5,0.5,...</td>
<td>1.32</td>
<td>1.51</td>
</tr>
<tr>
<td>0.5,1,0.5,1,...</td>
<td>1.37</td>
<td>1.63</td>
</tr>
<tr>
<td>1,1,1,...</td>
<td>1.42</td>
<td>2.32</td>
</tr>
<tr>
<td>1,2,1,2,...</td>
<td>1.49</td>
<td>3.86</td>
</tr>
<tr>
<td>2,2,2,2,...</td>
<td>1.30</td>
<td>5.13</td>
</tr>
<tr>
<td>2,4,2,4,...</td>
<td>1.28</td>
<td>6.92</td>
</tr>
<tr>
<td>4,4,4,4,...</td>
<td>1.28</td>
<td>8.81</td>
</tr>
<tr>
<td>4,8,4,8,...</td>
<td>1.38</td>
<td>10.98</td>
</tr>
</tbody>
</table>

References


