Mohammadi, M., Dauzère-Pérès, S., & Yugma, C. (2019). Performance evaluation of single and multi-class production systems using an approximating queuing network. *International Journal of Production Research*, *57*(5), 1497-1523. doi:10.1080/00207543.2018.1492163

Performance evaluation of single and multi-class production systems using an approximating queuing network

Mehrdad MOHAMMADI*[a], Stéphane DAUZERE-PERES[b,c], Claude YUGMA[b]

[a]IMT Atlantique, Lab-STICC, UBL, F-29238 Brest, France

[b]Mines Saint-Etienne, Univ. Clermont Auvergne, CNRS, UMR 6158 LIMOS, CMP, Department of Manufacturing Sciences and Logistics, Gardanne, France

[c]Department of Accounting, Auditing and Business Analytics, BI Norwegian Business School, Oslo, Norway

Abstract

Performance evaluation, and in particular cycle time estimation, is critical to optimize production plans in high-tech manufacturing industries. This paper develops a new aggregation model based on queuing network, so-called queue-based aggregation (QAG) model, to estimate the cycle time in a production system. Multiple workstations in serial and job-shop configurations are aggregated into a single-step workstation. The parameters of the aggregated workstation are approximated based on the parameters of the original workstations. Numerical experiments indicate that the proposed QAG model is computationally efficient and yields fairly accurate results when compared to other aggregation, approaches in the literature.

Keywords: Performance evaluation; Serial and job-shop production systems; Open queue network; Cycle time estimation; Approximating queuing network

* Corresponding author: Mehrdad MOHAMMADI (mehrdad.mohammadi@imt-atlantique.fr). Tel: +33 (0)2 29 00 10 30.

Performance evaluation of single and multi-class production systems using an approximating queuing network

Abstract

Performance evaluation, and in particular cycle time estimation, is critical to optimize production plans in high-tech manufacturing industries. This paper develops a new aggregation model based on queuing network, so-called queue-based aggregation (QAG) model, to estimate the cycle time in a production system. Multiple workstations in serial and job-shop configurations are aggregated into a single-step workstation. The parameters of the aggregated workstation are approximated based on the parameters of the original workstations. Numerical experiments indicate that the proposed QAG model is computationally efficient and yields fairly accurate results when compared to other aggregation approaches in the literature.

Keywords: Performance evaluation; Serial and job-shop production systems; Open queue network; Cycle time estimation; Approximating queuing network

1. Introduction and background

High-tech manufacturing industries are under increasing pressure to offer a wide variety of products to their customers with shorter lead times and at minimal cost. This has rendered production planning more and more important and has created remarkable interest in improving factory efficiency. When planning operations in workstations, there is always a trade-off between productivity and responsiveness (i.e., respecting due dates). A workstation refers to a unique equipment or a set of equipment that performs similar operations and might share the same input buffer and production resources. Workstation productivity is defined as the number of products processed per time unit, which is also referred to as throughput.

Comprehensively, high-tech manufacturing industries are looking for high workstation productivity because of the intensive capital investment. However, high workstation productivity results in large cycle times and less responsiveness. The cycle time is defined as the sum of the process time and the waiting time at the workstation for each product [1]. With large cycle times, many products might not meet their due date. The responsiveness is defined as the capability of the workstation to meet due dates. The increased level of complexity combined with the price pressure requires a scientific approach to evaluate the balance between productivity and due dates.

An accurate prediction of the cycle time distribution as a function of the throughput is then essential. A prediction model has to incorporate the workstation behavior such as integrated processing (i.e., processing multiple products at the same time in the various process clusters/chambers), predicted/unpredicted breakdowns and dispatching rules [1], [2]. It is also desired that the proposed prediction model requires little development and maintenance effort and the model evaluations are computationally cheap [2], [3].

For predicting the cycle time in high-tech manufacturing, there are two common categories of models including 1) “Simple” analytical models (i.e., models that require only a few easily-used and easily-estimated parameters) and 2) (discrete-event) Simulation models. Commonly-used simple analytical models are closed form G/G/m queueing models [2]–[12]. Despite the usefulness and easiness of queueing models, they suffer from the lack of accuracy for complex workstations/equipment. However, some modifications have been proposed by Morrison and Martin [2], [7], [8] for predicting the cycle time of cluster tools in semiconductor manufacturing. Simulation models, on the other hand, are alternatively used to accurately represent the processing of workstations [13]–[15]. Simulation modeling allows all relevant factory floor details to be taken into account [16], [17]. This necessitates the collection of all the required input data related to the various process parameters. Consequently, an enough accurate simulation model becomes computationally very expensive and requires significant development time [3]. Incorporating too many inevitable details makes simulation modeling impractical for quick throughput estimation.

One way to abstract a detailed simulation model is to perform simulation runs according to a design of experiments, and use the responses to generate a metamodel [18], [19]. One of the drawbacks of this approach is that different types of changes in the workstation require re-running the simulation model.

Another approach to providing an abstraction of a detailed simulation model is aggregation [2]. Aggregation can be done by simplifying the complex components/assumptions of the system as [20], and [21] replaced non-bottleneck workstations by a constant delay, but they did not use the simplified model for cycle time distribution prediction. Using delay distributions, Rose [22] aggregated all workstations except the bottleneck station; however, for certain scenarios, this aggregation model fails to predict the cycle time distributions accurately.

Hopp and Spearman [23], [24] and Jacobs et al. [25] proposed an algorithm to integrate the process time distributions and outage delays in the workstation. This aggregate process time is referred to the Effective Process Time (EPT). Hopp and Spearman [23] defined the EPT as "the process time seen by a product at a workstation from a logistical point of view", wherein the mean and the variance of the EPT are approximated from the raw process time and the preemptive and non-preemptive outages. Finally, the authors used the approximated mean and variance of the EPT in a closed-form G/G/m queuing model to estimate the mean cycle time. Since data for the shop floor details with different distributions may not always be available, Jacobs et al. [26] developed an algorithm to approximate the EPT distribution parameters directly from the products' arrival and departure times at the workstation.

In a workstation with integrated equipment, the EPT distribution parameters depend on the processing load or Work-In-Process (WIP), since multiple products may simultaneously be in the process [1]. On the other hand, the outage delays (e.g., preventive maintenance) due to machine idleness can also cause the WIP-dependency of the EPT distribution parameters [27]. To cope with this issue, Kock et al. [3] proposed a G/G/m-based aggregate simulation model with a WIP-dependent EPT-distribution to predict the mean cycle time. Veeger et al. [28] argued that although the aggregate model of [3] can predict the mean cycle time as a function of the throughput for workstations, the model does not necessarily lead to accurate cycle time estimation, due to the First-Come-First-Served (FCFS) rule in the aggregate model.

Similar to Kock et al. [3], Veeger et al. [1] developed an aggregate model using a WIP-dependent EPT distribution that takes into account the order in which products are processed. Accordingly, each product that arrives in the aggregate model has a probability of overtaking a number of other products already in the system. This number is determined by a WIP-dependent overtaking distribution measured from the arrival and departure times of the products. They applied the proposed aggregate model to estimate cycle time distributions for semiconductor workstations. The authors first built a detailed simulation model of the workstation and the aggregate model is then trained using arrival and departure data measured at the simulated workstation model. Numerous replications (i.e., 10^5 runs) were conducted in a specific level of throughput rate (e.g., 80%) for the simulated model to obtain significantly enough arrival and departure events to get a better EPT estimation. Once the WIP-based EPT distribution is estimated, the aggregated simulation model is built to estimate the cycle time of the workstation. The first drawback of this study is that the approach for predicting cycle time distributions is computationally expensive due to the need for creating detailed and aggregated simulation models with numerous replications. Secondly, the detailed and aggregate simulation models are run for a specific throughput rate as well as for given details of the workstation (e.g., number of parallel servers, number of processing steps etc.). When adding more details or changing the structure of the workstation (e.g., increasing the number of processing stations), the EPT-distribution might change, and we need to re-create and re-run the detailed and the aggregate simulation models.

Besides the above-mentioned aggregation methods to estimate the workstation's cycle time, Morrison [29] developed flow line models to estimate the cycle time of clustered photolithography tools in semiconductor manufacturing. In the flow line model, the cycle time, the start time and the process time of the products are calculated recursively from the completion time of the products. The flow line models of [29] have been only proposed for a single-class workstation/equipment while considering multiple products needs to reconstruct the models and take into account the inter-product correlations. In another work, Morrison and Martin [2] considered the processing workstations as general G/G/m queues and proposed a closed deterministic formulation for approximating the cycle time of the workstations subject to server failures and cycle time offsets (e.g., events such as travel and hold upstream of a workstation). The authors then developed flow line models for clustered photolithography tools and conducted simulations to assess the quality of the models. Although this queue-based approximation model is computationally cheaper than EPT-based aggregation models but similar to [29], the main drawback of the queue-based approximation model of [2] is twofold. First, the model works only for single-class workstations. Second, the model is unable to approximate the cycle time of integrated workstations including serial/non-serial configurations of the processing steps.

Jarrahi and Abdul-Kader [30] developed an analytical method to measure the performance, namely the Total Cycle Time, of a multi-product unreliable production line with finite buffers between workstations. The proposed approximation generalizes the processing times to ease the variation of product types in a multi-product system. A decomposition method, which considers generally distributed processing times as well as random failure and repair, is presented to approximate the production rate of a multi-product production line. A GI/G/1/N queuing model is used to obtain parameters such as blocking and starvation probabilities that are needed for the approximation. Jarrahi and Abdul-Kader [30] concluded that by increasing buffer capacity, the accuracy and performance of the line are improved up to a certain point, and thereafter, increasing buffer size will not make a significant improvement. Another result obtained from the numerical study is the improvement in the accuracy of the approximation when the number of product types increases. Jarrahi and Abdul-Kader [30] explain that increasing the number of product types makes the studied system closer to a system with generally distributed processing times, and since the concept of general distribution is used for tackling the variety in the product types, the observed improvement in the approximation is well explained.

Sharma and Jain [31] assessed the performance of nine dispatching rules in a stochastic dynamic job shop. The dispatching rules are considering the following performance measures: cycle time, mean flow time, maximum flow time, mean tardiness, maximum tardiness, number of tardy jobs, total setups and mean setup time. A discrete event simulation model of a stochastic dynamic job shop manufacturing system is developed for investigation purpose. Nine dispatching rules from the literature are incorporated into the simulation model. MacGregor [32] presented a new methodology for modeling exponential closed finite queueing networks and their corresponding material handling systems. A queue-based decomposition approach using state-dependent queues was used to capture the buffer of finite M/M/1/K queues. In this study, each M/M/1/K queue is replaced with a coupled state dependent queue plus an M/M/1 queue. Finally, an extended Mean Value Analysis (MVA) algorithm was employed to demonstrate the integration of the state dependent queues for the buffers in the approach. Baumann and Sandmann [33] presented a computational analysis of steady-state performance measures for multi-server tandem queues with Markovian arrival process, finite buffers, and phase-type distributed service times at both queueing nodes, where losses and blocking can occur. Their approach is based on the structured modeling of such tandem queues as level-dependent quasi-birth-and-death processes and using suitable computationally efficient matrix-analytic algorithms. This approach is conceptually exact, that is, the modeling does not introduce any approximation such as, e.g., decomposition approaches and the computational analysis using the matrix-analytic algorithm is exact up to numerical errors.

Sattler [34] and Wu [35] proposed simple queuing models to approximate the cycle time of a single class in a semiconductor fab and consequently to improve the productivity of the production system. The proposed approximation models were used for a single machine production system with a general distribution of arrival and service times. Yang et al. [36] developed a nonlinear regression meta-model supported by queueing theory to represent the underlying Cycle time-Throughput (CT-TH) curve related to a manufacturing simulation model. To estimate the model efficiently, simulation experiments were built up sequentially using a multistage procedure.

Inspired by the underlying structure of tandem queues, Wu and McGinnis [37] derived a single-class approximate model to characterize the system performance. Their model decomposes the system queue time and variability into bottleneck and non-bottleneck parts while capturing the dependence among workstations. Their model is iteratively solved to approximate the cycle time of the production system. When comparing their model with prior approximation approaches, the new model not only is more accurate but also requires less information.

In an interesting study, Bandi et al. [38] proposed an alternative approach for studying queues based on robust optimization. The authors modeled the uncertainty in the arrivals and services via polyhedral uncertainty sets which are inspired by the limit laws of probability. Using the generalized central limit theorem, their framework helps to model heavy-tailed behavior characterized by bursts of rapidly occurring arrivals and long service times. By considering a worst-case approach, they obtained closed-form upper bounds on the system time in a single-class FCFS queue for both single-server and multi-server queues. Their approximated bounds are nearly tight for heavy-traffic systems operating under steady state.

In a different application, Nazzal and McGinnis [39] proposed a queue-based analytical approach to model the performance of a simple closed loop Automated Material Handling System, which is typical in supporting a 300 mm wafer fab bay. Due to the significant impact of vehicle blocking, a straightforward queueing network model which treats the material handling system as a central server can be inaccurate. Hence, Nazzal and McGinnis [39] proposed an alternative model that estimates the material handling system performance considering the possibility of vehicle blocking. In another work, Nazzal [40] modeled a multi-vehicle material handling system as a closed-loop queueing network with finite buffers and general service times, where the vehicles represent the jobs in the network. In this work, the vehicles' residence times on track segments (servers) depend on the number of jobs (vehicles) in circulation. A new iterative approximation algorithm is developed that estimates throughput capacity and decomposes the network consisting of S servers into S separate G/G/1 systems. Tu et al. [41] studied the Automated Material Handling

System capacity determination model in order to maintain the originally designed optimal production throughput or cycle time of products. A GI/G/m queuing model is applied based on the First-Come-First-Serve (FCFS) dispatching rule for the Automated Material Handling System to determine the required number of vehicles. In this model, products should be transported to the specific workstation before the workstation finishes the existing process; therefore, sufficient WIP in front of this specific workstation should be kept.

This paper develops a new aggregation model based on queueing network, so-called queue-based aggregation (QAG) model, to estimate the cycle time of serial and non-serial (i.e., job shop) configured workstations. Queueing networks with finite/infinite buffers are useful for modeling and analyzing discrete event systems such as manufacturing systems, computer systems, and communication systems [42]. In the case of manufacturing systems, serial production systems can be modeled as a tandem configuration of queueing networks while job-shop production systems can be modeled as an arbitrary configuration of queueing networks.

Large numbers of studies have been conducted on the performance evaluation of serial and arbitrarily configured queuing networks to approximate the queue length as well as the waiting time at each node/workstation of the network [5], [42]–[46] and references therein. To the best of our knowledge, almost all of these works have used a decomposition-based approach to decompose the network into its components and have evaluated each component separately to estimate the queue length and the waiting time of that component. None of them proposed an approximation model for the whole network. Accordingly, this paper aims at proposing an approximation model for the entire network (i.e., manufacturing system) by aggregating multiple workstations into a single-step workstation. The parameters of the aggregated workstation are approximated based on the parameters of the original workstations. This paper not only helps to provide an abstraction of detailed simulation models for complex manufacturing systems but also proposes an accurate and timely efficient estimation of the cycle time of serial or arbitrarily configured manufacturing systems.

The paper is organized as follows. Section 2 develops the proposed QAG models. Section 3 provides two experiments to validate the correctness and the performance of the proposed QAG models in comparison to classical methods of the literature. Sensitivity analysis is performed in Section 3. Finally, we conclude the paper in Section 4.

2. Proposed Queue-based Aggregation Model – QAG Model

This section develops new aggregation models based on queuing networks to accurately and efficiently estimate the cycle time in complex manufacturing systems. For this aim, manufacturing systems are described in the form of single/multiple-class open queuing networks, wherein the nodes in the network represent the processing workstations of the manufacturing system. A queueing network is said to be open when external flow units (products) can enter the network at every node, and the internal flow can leave the network from any node. A queueing network is called closed when flow units can neither enter nor leave the network. Accordingly, the number of flow units in a closed queue network is constant. The queueing network can also be called as a single class and multiclass network depending on whether the queueing network serves single or multiple types of products [43].

It is obvious that queueing networks consisting of several service workstations are more suitable for representing the structure of many manufacturing systems with a wide range of resources than simple queue models with only a single service station, as in [2]. It is noteworthy that an underlying assumption in a queueing network is that at least two workstations are connected to each other.

In the following, two types of manufacturing systems are studied, i.e., parallel-series and job-shop systems and the relevant QAG models are proposed. Sections 2.1 and 2.2 propose the QAG models for the parallel-series manufacturing system and the job-shop manufacturing system, respectively. Hereafter, let *PS*, *JS*, *APS* and *AJS* respectively denote Parallel-Series, Job-Shop, Aggregated Parallel-Series and Aggregated Job-Shop.

2.1. QAG model for a single-class PS system

This section develops a QAG model for a single-class *PS* system and provides closed approximation formulations to estimate the cycle time.

2.1.1. Notations

Necessary notations for developing the QAG model of a single-class *PS* manufacturing system are listed below. All stochastic parameters are average values.

- Original system notations

Sets:

$\mathbb{S}$	Set of processing steps, $\lvert\mathbb{S}\rvert = N$
M	Number of parallel production lines in the *PS* system
i	Index of processing steps; $i \in \mathbb{S}$

Parameters:

λ_i	Arrival rate of the products at processing step *i*
μ_i	Service rate at processing step *i*
μ_B	Service rate at bottleneck processing step *B*, $B \in \mathbb{S}$ (the bottleneck processing step *B* is expressed as the processing step with minimum service rate or longest service time).
$\sigma_{A,i}^2$	The variance of the inter-arrival time at processing step *i*
$\sigma_{S,i}^2$	The variance of the service time at processing step *i*
c_i	Number of servers at processing step *i*
ρ_i	Utilization rate at processing step *i*
$c_{A,i}^2$	Squared Coefficient of Variation (SCV) of the inter-arrival time at processing step *i*
$c_{S,i}^2$	SCV of the service time at processing step *i*
$c_{D,i}^2$	SCV of the inter-departure time at processing step *i*
$\tau_{A,i}$	Inter-arrival time at processing step *i* ($\tau_{a,i} = 1/\lambda_i$)
$\tau_{S,i}$	Service time at processing step *i* ($\tau_{s,i} = 1/\mu_i$)

Variables:

W_i	Waiting time at processing step *i*
CT_i	Cycle time at processing step *i*
L_i	Work-In-Process (WIP) at processing step *i*
W^{PS}	Waiting time for the *PS* system
CT^{PS}	The cycle time of the *PS* system
L^{PS}	Total WIP in the *PS* system

- Aggregated system notations

Parameter:

λ^{APS}	Arrival rate of the products to *APS* system

Variables:

μ^{APS}	Service rate of *APS* system
$\sigma_S^{2,APS}$	The variance of the service time at *APS* system
ρ^{APS}	The utilization rate of *APS* system
$c_A^{2,APS}$	SCV of the inter-arrival time at *APS* system
$c_S^{2,APS}$	SCV of the service time at *APS* system
$c_D^{2,APS}$	SCV of the inter-departure time at *APS* system
τ^{APS}	The service time of *APS* system
W^{APS}	Waiting time of *APS* system
CT^{APS}	The cycle time of *APS* system

2.1.2. Characterization of the PS and APS systems

Figure 1 shows a *PS* manufacturing system with *M* identical parallel production lines, wherein each production line consists of *N* processing steps in series. The manufacturing system has a specific arrival rate of a single product, and this arrival is equally divided between production lines. Accordingly, we consider that each line has an equal arrival rate which is identical to λ_1. Each processing step *i* is modeled as a G/G/c queue wherein the inter-arrival times between flow units are given by a random variable with general distribution and mean $1/\lambda_i$. The service times are imposed by a random variable with general distribution and mean $1/\mu_i$. All inter-arrival and service times are independent. Before each processing step *i*, there exists an infinite buffer and no bulk of flow and the flow units are served in a first-come first-served (FCFS) manner. Each processing step *i* contains *c* parallel servers and each server processes only one unit at a time and devotes all of its resources to complete the transaction. If a server is idle, it will immediately start to process an available unit from the queue.

The utilization rate at processing step *i* is represented as $\rho_i = \lambda_i / c_i \mu_i$ (i.e., $\rho_i = \tau_{S,i} / c_i \tau_{A,i}$). The SCV of the inter-arrival time and service time at processing step *i* are explained as $c_{A,i}^2 = \sigma_{A,i}^2 / \tau_{A,i}^2$ and $c_{S,i}^2 = \sigma_{S,i}^2 / \tau_{S,i}^2$, respectively [2]. The key performance metrics (KPIs) at every processing step *i* are the average waiting time W_i, and the average cycle time CT_i. The cycle time is defined as the mean time that a unit spends in the queue and for receiving services [2]. Accordingly, the KPIs for the entire *PS* manufacturing system are determined as the cycle time, C^{PS}, and the waiting time, W^{PS}, of the entire *PS* system. These KPIs are illustrated in Figure 2 for each processing step and the entire *PS* system.

To be precise about the serial production line in Figure 2, a list of basic assumptions is provided as follows.

Assumption 1. The network is *open* rather than closed. Products come from outside, receive service from all nodes subsequently, and eventually leave the production line.

Assumption 2. There are *no storage capacity constraints*. There is no limit on the WIP in the entire production line, and each processing step has unlimited waiting space. This paper deals with mass production systems such as semiconductor manufacturing systems. In such systems, there are enough storage areas to store all products. Once new places in the finite buffer of a machine become available, products are transferred from the storage areas to the buffer. Without loss of generality, it is thus possible to consider that the buffer sizes are actually infinite. Therefore, the proposed model is suitable for mass production systems with no storage capacity limitations.

Assumption 3. There can be *any number* of servers at each processing step. The servers are independent, and each of them serves a single product at a time.

Assumption 4. Processing steps are stochastically independent.

Assumption 5. Products are served according to a *first-come*, *first-serve* policy.

Assumption 6. There is only one class of products.

Assumption 7. Products are not created and combined at the processing steps, e.g., one arrival causes only one departure.

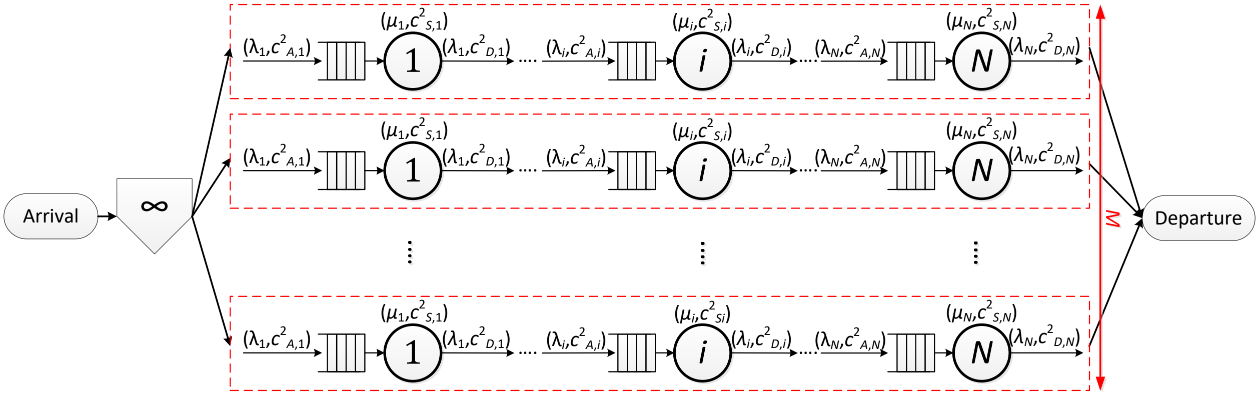

Figure 1. A *PS* manufacturing system as an open queue network

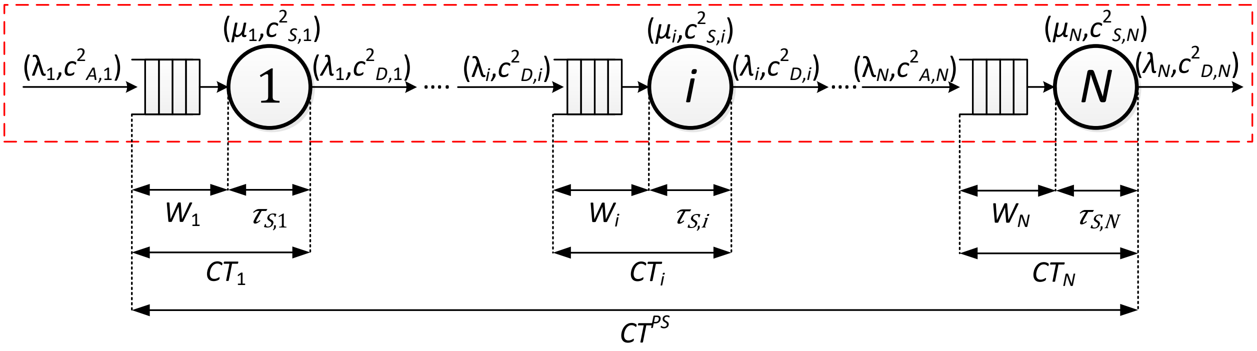

Figure 2. A serial production line with corresponding KPIs

The main purpose of illustrating the serial production line as the serial queue network of Figure 2 is to represent all the arrival processes and service-time distributions by a limited number of parameters. The waiting time at each processing step is then calculated by an approximating formulation that depends only on these parameters.

The most popular approximation scheme for serial queue networks is the Queueing Network Analyzer (QNA) developed by Whitt [47]. Whitt [47] developed the QNA approach based on the above-mentioned assumptions, Marshall's equation [48] and Kingman's heavy-traffic approximation [49], [50]. After considering each processing step as a G/G/c queue, the QNA approach [6] provides the approximation Eq. (1) for calculating the waiting time of processing step *i*.

$$W_i \cong \left(\frac{c_{A,i}^2 + c_{S,i}^2}{2}\right)\frac{\rho_i{}^{\sqrt{2(c_i+1)}-1}}{c_i(1-\rho_i)} \times \tau_{S,i} \qquad \forall i \tag{1}$$

where the utilization rate of processing step *i*, ρ_i is described as $\tau_{S,i}/c_i\tau_{A,i}$ $(=\lambda_i/c_i\mu_i)$ and a valid equality is $\lambda_i = \lambda_1$ for $i = 2, \dots, N$. A better approximation has been proposed for calculating W_i by Kramer-Langenbach-Belz [50] as Eq. (2).

$$W_i \cong \left(\frac{c_{A,i}^2 + c_{S,i}^2}{2}\right)\frac{P_i\tau_{S,i}}{c_i(1-\rho_i)} \times \mathbb{G}_i \qquad \forall i \tag{2}$$

where $\mathbb{G}_i$ is calculated as:

$$\mathbb{G}_i = \begin{cases} \exp\left(-\frac{2}{3} \times \frac{1-\rho_i}{P_i} \times \frac{\left(1-c_{A,i}^2\right)^2}{c_{A,i}^2 + c_{S,i}^2}\right), & 0 \le c_{A,i} \le 1 \\ \exp\left(-(1-\rho_i) \times \frac{c_{A,i}^2 - 1}{c_{A,i}^2 + c_{S,i}^2}\right), & c_{A,i} > 1 \end{cases} \tag{3}$$

where $P_i = \frac{\rho_i{}^{c_i+\rho_i}}{2}$ if $\rho_i > 0.7$ and $P_i = \rho_i{}^{\frac{c_i+1}{2}}$ if $\rho_i < 0.7$. The SCV of the inter-departure time at processing step *i* is characterized as Eq. (4) using Marshall's equation [40]:

$$c_{D,i}^2 = 1 + (1-\rho_i^2)\left(c_{A,i}^2 - 1\right) + \frac{\rho_i^2}{\sqrt{c_i}}\left(c_{S,i}^2 - 1\right) \qquad \forall i \tag{4}$$

Eq. (4) is exact and yields $c_{D,i}^2 = 1$ for M/M/c and M/G/c. The third term in Eq. (4) approaches 0 as c_i increases, reflecting the way multiple servers tend to act as a superposition operation. Iteratively applying Eq. (4) and knowing that $c_{A,i}^2 = c_{D,i-1}^2$, we obtain the recursive Eq. (5) for $c_{D,i}^2$.

$$c_{D,i}^2 = 1 + (1-\rho_i^2)\left(c_{D,i-1}^2 - 1\right) + \frac{\rho_i^2}{\sqrt{c_i}}\left(c_{S,i}^2 - 1\right) \qquad \forall i \ge 2 \tag{5}$$

The closed-form version of Eqs. (4) and (5) is provided as Eqs. (6) and (7) below:

$$c_{D,i}^2 = 1 + \theta_{i,1}\left(c_{A,1}^2 - 1\right) + \sum_{k=1}^{i} \theta_{i,k+1}\frac{\rho_k^2}{\sqrt{c_k}}\left(c_{S,k}^2 - 1\right) \qquad \forall i \tag{6}$$

$$\theta_{i,k} = \prod_{j=k}^{j=i}\left(1-\rho_j^2\right) \qquad \forall i,k; 1 \le k \le i, \theta_{i,i+1} = 1 \tag{7}$$

As a consequence of Eq. (4), $c_{D,i}^2$ is a convex combination of $c_{A,1}^2$ and $c_{S,j}^2$, $1 \leq j \leq i$. The weight of $c_{S,j}^2$ is increasing in ρ_j and decreasing in ρ_k, $1 \leq k \leq j \leq i$ and $k \neq j$. Whitt [51] pointed out that the performance of Eq. (4) in approximating the SCV of inter-departure time, deteriorates in the presence of high variability, especially in the arrival process, and the method tends to perform poorly when the service time is deterministic or nearly deterministic.

The cycle time CT_i of processing step *i* is the sum of waiting time, W_i, and the mean process time $\tau_{S,i}$. Eq. (8) presents the cycle time KPI for each processing step *i*. Applying Little's law, the WIP L_i at processing step *i* is calculated as:

$$CT_i = W_i + \tau_{S,i} \qquad \forall i \tag{8}$$

$$L_i = \lambda_i CT_i \qquad \forall i \tag{9}$$

Hereafter, the aim of this section is to convert the serial production line of Figure 2 into an aggregated single-step processing step as reflected in Figure 3, wherein the *N* serial processing steps are aggregated into a single processing step with corresponding parameters and KPIs. The parameters of the aggregated processing step in Figure 3 are approximated based on the parameters of the original serial processing steps of Figure 2.

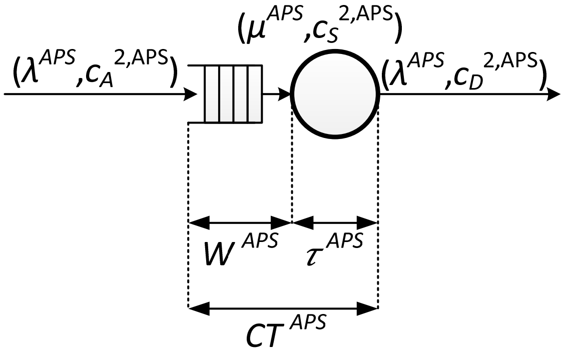

Figure 3. The single-step aggregated processing step

2.1.3. Approximation for the APS system

This section aims at approximating the variables of the aggregated single-step processing step based on the parameters of the original serial processing steps. From Figure 2, the average cycle time of the entire serial production line, CT^{PS}, can be expressed as the sum of the cycle times at all processing steps as Eq. (10):

$$CT^{PS} = \sum_{i=1}^{N} CT_i \tag{10}$$

Eq. (10) can be rewritten as:

$$CT^{PS} = \sum_{i=1}^{N} \left(W_i + \tau_{S,i}\right) \tag{11}$$

The idea of this paper is that the average service time in the aggregate system is the total time that each product spends in the system after departing the queue of the first processing step (i.e., *i*=1). This time is equal to the sum of the service times for all processing steps and the in-between waiting times. By "in-between" waiting time, we mean the sum of the waiting times of processing step *i* $\forall i = 2, \ldots, N$. Accordingly, we express the average service time in the aggregate system as Eq. (12). Note that, unlike the literature [2]–[5], [29], [39], [40], the average service time in Eq. (12) is WIP-dependent. When the WIP increases, the waiting time increases and the service time of the *APS* system increases as well.

$$\tau^{APS} = \sum_{i=2}^{N} W_i + \sum_{i=1}^{N} \tau_{S,i} \tag{12}$$

An underlying assumption in heavy traffic condition for serial queue networks is that the waiting time of the network is dominated by the waiting time at the bottleneck processing step [4]–[6], [47], [48], [50]–[52]. Knowing that $\lambda_i = \lambda_1$ for $i = 2, \dots, N$, the bottleneck processing step *B* is expressed as the processing step with minimum service rate or longest service time, i.e., $\mu^{APS} = \mu_B = \min\limits_{i \in \mathbb{S}} \mu_i$. Accordingly, the waiting time of the aggregated system W^{APS} can be calculated through Eqs. (13) and (14).

$$W^{APS} \approx \frac{\rho^{APS}/\mu^{APS}}{1-\rho^{APS}} \times \frac{c_A^{2,APS} + c_S^{2,APS}}{2} \times \mathbb{G}_{APS} \tag{13}$$

$$\mathbb{G}_{APS} = \begin{cases} \exp\left(-\frac{2}{3} \times \frac{1-\rho^{APS}}{\rho^{APS}} \times \frac{\left(1-c_A^{2,APS}\right)^2}{c_A^{2,APS} + c_S^{2,APS}}\right), & 0 \le c_A^{APS} \le 1 \\ \exp\left(-(1-\rho^{APS}) \times \frac{c_A^{2,APS} - 1}{c_A^{2,APS} + c_S^{2,APS}}\right), & c_A^{APS} > 1 \end{cases} \tag{14}$$

where $c_A^{2,APS} = {c_{A,1}}^2$ and $\rho^{APS} = \lambda^{APS}/\mu_B$ and $\lambda^{APS} = \lambda_1$. To calculate $c_S^{2,APS}$, we first need to calculate the variance of the service time in the aggregated system, $\sigma_S^{2,APS}$, as Eq. (15), wherein Var(*X*) is the variance of variable *X*.

$$\sigma_S^{2,APS} = \mathrm{Var}\left(\sum_{i=2}^{N} W_i + \sum_{i=1}^{N} \tau_{S,i}\right) \tag{15}$$

Since the waiting times, as well as the service times of the processing steps, are identical and independent, Eq. (15) can be re-written as Eqs. (16) and (17).

$$\sigma_S^{2,APS} = \textstyle\sum_{i=2}^{N} \mathrm{Var}(W_i) + \sum_{i=1}^{N} \mathrm{Var}\left(\tau_{S,i}\right) \tag{16}$$

$$\sigma_S^{2,APS} = \sum_{i=2}^{N} \mathrm{Var}(W_i) + \sum_{i=1}^{N} \sigma_{S,i}^2 \tag{17}$$

Applying the Kingman-Kollerstrom approximation [53], the cumulative distribution function of waiting time for a G/G/c queue can be well approximated by Eq. (18).

$$F_{W_i}(x) = 1 - \exp\left(-\frac{2}{c_{A,i}^2 + c_{S,i}^2} \frac{c_i(1-\rho_i)}{\rho_i \tau_{S,i} \mathbb{G}_i} x\right) \tag{18}$$

Accordingly, the variance of the waiting time at processing step *i* can be expressed as:

$$\mathrm{Var}(W_i) = \frac{\left(c_{A,i}^2 + c_{S,i}^2\right)^2}{4} \frac{\rho_i^2 \tau_{S,i}^2}{c_i^2 (1-\rho_i)^2} \times \mathbb{G}_i^2 \tag{19}$$

Finally, the SCV of the service time for the aggregated system is approximated as:

$$c_S^{2,APS} \approx \frac{\sum_{i=2}^{N} \left[\frac{\left(c_{A,i}^2 + c_{S,i}^2\right)^2}{4} \frac{\rho_i^2 \tau_{S,i}^2}{c_i^2 (1-\rho_i)^2} \times \mathbb{G}_i^2\right] + \sum_{i=1}^{N} \sigma_{S,i}^2}{\sum_{i=2}^{N} W_i + \sum_{i=1}^{N} \tau_{S,i}} \tag{20}$$

Similar to Eq. (8), the average cycle time of the aggregated system is approximated as:

$$CT^{APS} = W^{APS} + \tau^{APS} \tag{21}$$

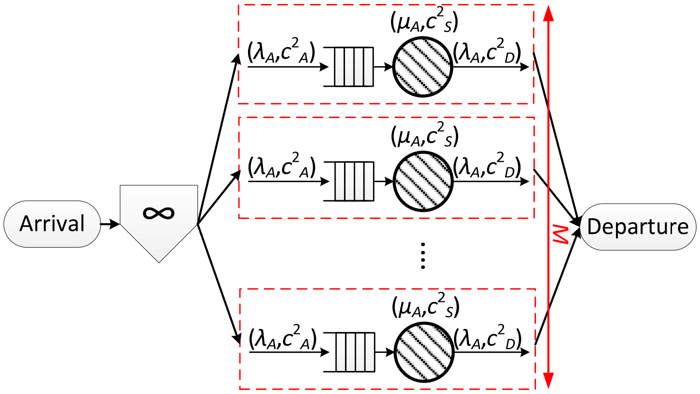

Figure 4. The aggregated parallel manufacturing system

After aggregating each serial production line (Figure 2) into a single-step processing step (Figure 3), the *PS* manufacturing system of Figure 1 is reduced to a "simple" aggregated parallel manufacturing system as shown in Figure 4. Finally, the KPIs of the *PS* manufacturing system (i.e., waiting time, cycle time and the WIP) are approximated as Eqs. (22) to (24). Instead of Eq. (22) for calculating the waiting time, one may use other well-known approximation formulations in the literature [2], [47], [50]–[55].

$$W^{PS} \cong \left(\frac{c_A^{2,APS}+c_S^{2,APS}}{2}\right)\frac{\rho^{APS}\tau^{APS}}{M(1-\rho^{APS})} \times \mathbb{G}_{APS} \tag{22}$$

$$CT^{PS} = W^{PS} + \tau^{APS} \tag{23}$$

$$L^{PS} = \lambda^{APS} CT^{PS} \tag{24}$$

2.2. QAG model for a multi-class *JS* manufacturing system

This section develops a QAG model for a multi-class *JS* manufacturing system and provides closed approximation formulations to estimate the cycle time.

2.2.1. Notations

The notations of Section 2.1.1 are modified and updated to take into account different types of products.

Sets:

$\mathbb{S}$ Set of processing steps, $|\mathbb{S}| = N$

P Set of products

i Index of processing steps; $i \in P$

p Index of products; $p \in P$

E_p Processing route of product p ($E_p = \{e_{1p}, e_{2p}, \dots, e_{ip}, \dots, e_{Np}\}$).

R_i Set of products with processing step *i* in their process.

Parameters:

$\lambda_{A,pi}$ Arrival rate of product *p* to processing step i

μ_{pi} Service rate of product *p* at processing step i

μ_{pB} Service rate of the bottleneck processing step *B* corresponding to product *p*

$\tau_{A,pi}$ Inter-arrival time of product p at processing step i

$\tau_{S,pi}$ The service time of product p at processing step i

c_i Number of servers at processing step i

$\sigma^2_{A,pi}$ The variance of the inter-arrival time of product *p* at processing step i

$\sigma^2_{S,pi}$ The variance of the service time of product *p* at processing step i

$c^2_{A,pi}$ SCV of the inter-arrival time of product *p* at processing step i

$c^2_{S,pi}$ SCV of the service time of product *p* at processing step i

ρ_{pi} The utilization rate of product *p* at processing step i

Variables:

- Original products

$\lambda_{D,pi}$ Departure rate of product p from processing step i

$c^2_{D,pi}$ SCV of the inter-departure time of product *p* at processing step i

W_{pi} Waiting time of product p in the queue at processing step i

L_{pi} WIP of product *p* in the queue at processing step i

CT_{pi} The cycle time of product *p* in the queue at processing step i

- Aggregated product

$\lambda_{A,\mathbb{P}i}$ Arrival rate of the aggregated product to processing step i in the *JS* system

$\lambda_{D,\mathbb{P}i}$ Departure rate of the aggregated product from processing step i in the *JS* system

$\mu_{\mathbb{P}i}$ Service rate of the aggregated product at processing step i in the *JS* system

$\rho_{\mathbb{P}i}$ The utilization rate of the aggregated product at processing step i in the *JS* system ($\rho_{\mathbb{P}i} = \sum_{p\in R_i} \rho_{pi} = \lambda_{A,\mathbb{P}i}/c_i\mu_{\mathbb{P}i}$)

$c^2_{A,\mathbb{P}i}$ SCV of the inter-arrival time of the aggregated product at processing step i in the *JS* system

$c^2_{S,\mathbb{P}i}$ SCV of the service time of the aggregated product at processing step i in the *JS* system

$c^2_{D,\mathbb{P}i}$ SCV of the inter-departure time of the aggregated product at processing step i in the *JS* system

$\tau_{A,\mathbb{P}i}$ Inter-arrival time of the aggregated product at processing step $\mathbb{S}_i$ ($\tau_{a,\mathbb{P}i} = 1/\lambda_{\mathbb{P}i}$)

$\tau_{S,\mathbb{P}i}$ The service time of the aggregated product at processing step i in the *JS* system ($\tau_{s,\mathbb{P}i} = 1/\mu_{i\mathbb{P}}$)

$W_{\mathbb{P}i}$ Waiting time for the aggregated product at processing step i in the *JS* system

$L_{\mathbb{P}i}$ WIP of the aggregated product at processing step i in the *JS* system

$CT_{\mathbb{P}i}$ The cycle time of the aggregated product at processing step i in the *JS* system

- Aggregated job-shop system

$\lambda^{AJS}_{A,p}$ Arrival rate of product p to the *AJS* system

$\lambda^{AJS}_{D,p}$ Departure rate of product p from the *AJS* system

μ^{AJS}_{p} Service rate of product p at the *AJS* system

ρ^{AJS}_{p} The utilization rate of product p at the *AJS* system

$c^{2,AJS}_{A,p}$ SCV of the inter-arrival time of product p at the *AJS* system

$c^{2,AJS}_{S,p}$ SCV of the service time of product p at the *AJS* system

$c^{2,AJS}_{D,p}$ The coefficient of variation of the inter-departure time of product p at the *AJS* system

$\tau^{AJS}_{A,p}$ Inter-arrival time of product p at the *AJS* system

$\tau^{AJS}_{S,p}$ The service time of product p at the *AJS* system

W^{AJS}_{p} Waiting time for product p in the queue at the *AJS* system

L^{AJS}_{p} Queue length of product p in the *AJS* system

CT^{AJS}_{p} The cycle time of product p in the *AJS* system

2.2.2. Characterization of the *JS* system

Figure 5 shows an example of a multi-class *JS* manufacturing system containing six processing steps and three products. Each product is routed through a set of processing steps, and the processing routes of the products are different. For instance, the processing route of the products are $E_1 = \{1,2,4\}$, $E_2 = \{4,5,3\}$, and $E_3 = \{5,6,1\}$. In addition, we have $R_1 = \{1,3\}$, $R_2 = \{1\}$, $R_3 = \{2\}$, $R_4 = \{1,2\}$, $R_5 = \{2,3\}$, and $R_6 = \{3\}$. Figure 6 depicts the product-focused structure of the multi-class *JS* system and shows the serial processing route for each product.

Each processing step *i* is modeled as a G/G/c queue wherein the inter-arrival times between flow units of each product *p* are given by a random variable with general distribution and mean $1/\lambda_{A,pi}$. The service times are imposed by a random variable with general distribution and mean $1/\mu_{pi}$. Before each processing step *i*, there exists an infinite buffer and the products are served in a first-come first-served (FCFS) policy. Each processing step *i* contains *c* parallel servers and each server processes only one unit of each product at a time and devotes all of its resources to complete the transaction. The utilization rate of product *p* at processing step *i* is $\rho_{pi} = \lambda_{A,pi}/c_i\mu_{pi}$ (i.e., $\rho_{pi} = \tau_{S,pi}/c_i\tau_{A,pi}$). The SCV of the inter-arrival time and service time of product *p* at processing step $\mathbb{S}_i$ are $c^2_{A,pi} = \sigma^2_{A,pi}/\tau^2_{A,pi}$ and $c^2_{S,pi} = \sigma^2_{S,pi}/\tau^2_{S,pi}$, respectively.

The fork-joint analysis [43] is adopted to evaluate the performance of each processing step separately. In this analysis, the arrival products are aggregated into a single product (called as aggregated product $\mathbb{P}$) and the performance of the processing step is evaluated for the aggregated product. The arrival, service and departure parameters for the aggregated product are approximated based on the parameters of the original products at each processing step. Hereafter, the characterization of the aggregated product and the inter-departure expressions are approximated. Figure 7 illustrates a decomposed processing step *i* that processes a given set of products R_i.

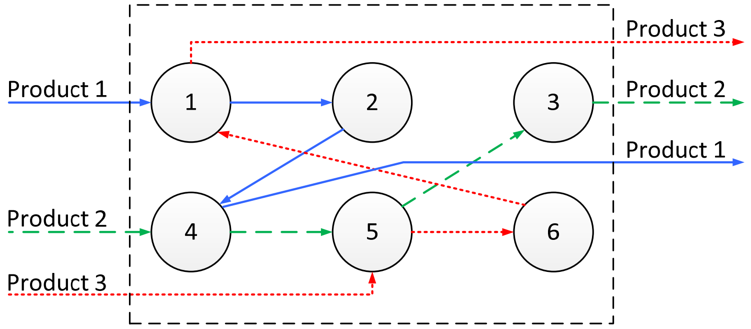

Figure 5. Process-focused structure of a multi-class *JS* manufacturing system

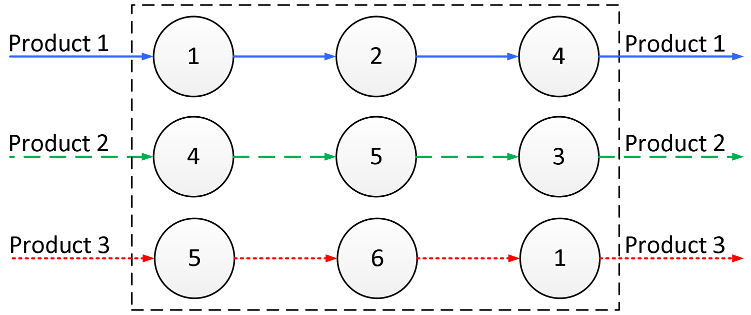

Figure 6. Product-focused structure of a multi-class *JS* manufacturing system

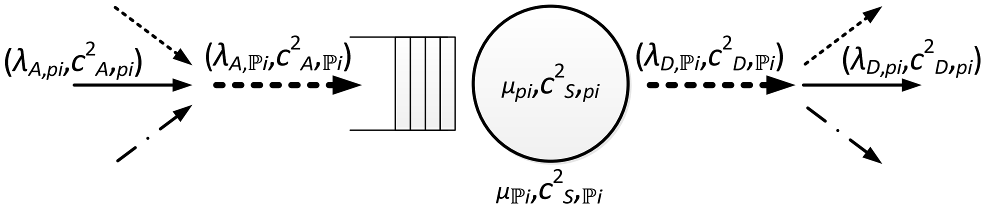
$(\lambda_{A,pi},c^2_{A,pi})$
$(\lambda_{A,\mathbb{P}i},c^2_{A,\mathbb{P}i})$
$\mu_{pi},c^2_{S,pi}$
$(\lambda_{D,\mathbb{P}i},c^2_{D,\mathbb{P}i})$
$(\lambda_{D,pi},c^2_{D,pi})$
$\mu_{\mathbb{P}i},c^2_{S,\mathbb{P}i}$

Figure 7. Decomposed processing step *i* and aggregated product

Based on the notations in Section 2.2.1, Eqs. (25) to (28) have been proposed to estimate the parameter of the aggregate product [47], [56].

$$\lambda_{A,\mathbb{P}i} = \sum_{p\in R_i} \lambda_{A,pi} \qquad \forall i \tag{25}$$

$$\tau_{S,\mathbb{P}i} = \sum_{p\in R_i} \left(\frac{\lambda_{A,pi}}{\lambda_{A,\mathbb{P}i}}\right)\tau_{S,pi} \qquad \forall i \tag{26}$$

$$c^2_{A,\mathbb{P}i} = \theta_i\delta_i^2 + (1-\theta_i) \qquad \forall i \tag{27}$$

$$c_{S,\mathbb{P}i}^2 = \frac{\left\{\sum_{p \in R_i} \left(\frac{\lambda_{A,pi}}{\lambda_{A,\mathbb{P}i}}\right) \left(c_{S,pi}^2 + 1\right)\left(\tau_{S,pi}\right)^2\right\} - \left(\tau_{S,\mathbb{P}i}\right)^2}{\left(\tau_{S,\mathbb{P}i}\right)^2} \qquad \forall i \tag{28}$$

where $\theta_i = [1 + 4(1 - \rho_{\mathbb{P}i})^2(v_i - 1)]^{-1}$, $\delta_i^2 = \sum_p \left(\frac{\lambda_{A,pi}}{\lambda_{A,\mathbb{P}i}}\right) c_{A,pi}^2$ and ${v_i}^{-1} = \sum_p \left(\frac{\lambda_{A,pi}}{\lambda_{A,\mathbb{P}i}}\right)^2$. Based on the estimated input parameters of the aggregated product provided in Eqs. (25) to (28), the average departure rate, $\lambda_{D,\mathbb{P}i}$, and SCV of the inter-departure time, $c_{D,\mathbb{P}i}^2$, of the aggregated product at processing step *i* are provided below, where:

$c_{A,p,e_{ip}}^2 = c_{D,p,e_{ip}-1}^2, \forall e_{ip} \geq 2.$

$$\lambda_{D,\mathbb{P}i} = \lambda_{A,\mathbb{P}i} \qquad \forall i \tag{29}$$

$$c_{D,\mathbb{P}i}^2 = (1 - \rho_{\mathbb{P}i}^2)c_{A,\mathbb{P}i}^2 + \rho_{\mathbb{P}i}^2 c_{S,\mathbb{P}i}^2 \qquad \forall i \tag{30}$$

Finally, the average departure rate, $\lambda_{D,pi}$, and SCV of the inter-departure time, $c_{D,pi}^2$, of product *p* at processing step *i* can be written as:

$$\lambda_{D,pi} = \lambda_{A,pi} \qquad \forall i \tag{31}$$

$$\begin{aligned} c_{D,pi}^2 = {} & {\rho_{pi}}^2 c_{S,pi}^2 + \left(1 - 2\rho_{pi}\rho_{\mathbb{P}i} + {\rho_{pi}}^2\right)c_{A,pi}^2 \\ & + \left(\frac{\lambda_{A,pi}}{\lambda_{A,\mathbb{P}i}}\right) \sum_{\substack{r \neq p \\ p \in R_i}} \frac{\lambda_{A,\mathbb{P}i}\rho_{ri}^2}{\lambda_{A,ri}} \left(c_{A,pi}^2 + c_{S,pi}^2\right) \qquad \forall i \end{aligned} \tag{32}$$

After calculating the parameters of the aggregated product at processing step *i*, the waiting time of product *p* in the queue at processing step *i*, W_{pi}, the WIP of product *p* at processing step *i*, L_{pi}, and the WIP of the aggregated product at processing step *i*, $L_{\mathbb{P}i}$, are estimated as Eqs. (33) to (35), respectively.

$$W_{pi} \approx \frac{P_{\mathbb{P}i}\tau_{S,\mathbb{P}i}}{c_i(1 - \rho_{\mathbb{P}i})} \times \frac{c_{A,\mathbb{P}i}^2 + c_{S,\mathbb{P}i}^2}{2} \times G_{\mathbb{P}i} \qquad \forall i, p \in R_i \tag{33}$$

$$L_{pi} = \lambda_{A,pi}\left(W_{pi} + \tau_{S,pi}\right) \qquad \forall i, p \in R_i \tag{34}$$

$$L_{\mathbb{P}i} = \sum_{p \in R_i} L_{pi} \qquad \forall i \tag{35}$$

where $P_{\mathbb{P}i} = \frac{{\rho_{\mathbb{P}i}}^{c_i} + \rho_{\mathbb{P}i}}{2}$ if $\rho_{\mathbb{P}i} > 0.7$, $P_{\mathbb{P}i} = {\rho_{\mathbb{P}i}}^{\frac{c_i+1}{2}}$ if $\rho_{\mathbb{P}i} < 0.7$ and $G_{KLB,\mathbb{P}i}$ is calculated as:

$$G_{\mathbb{P}i} = \begin{cases} \exp\left(-\frac{2}{3} \times \frac{1 - \rho_{\mathbb{P}i}}{P_{\mathbb{P}i}} \times \frac{\left(1 - c_{A,\mathbb{P}i}^2\right)^2}{c_{A,\mathbb{P}i}^2 + c_{S,\mathbb{P}i}^2}\right), & 0 \leq c_{A,\mathbb{P}i} \leq 1 \\ \exp\left(-(1 - \rho_{\mathbb{P}i}) \times \frac{c_{A,\mathbb{P}i}^2 - 1}{c_{A,\mathbb{P}i}^2 + c_{S,\mathbb{P}i}^2}\right), & c_{A,\mathbb{P}i} > 1 \end{cases} \qquad \forall i \tag{36}$$

2.2.3. Characterization of the AJS system

According to Eq. (33), it is clear that the mean waiting time at processing step *i* is the same for all products *R_i*. Because the processing route of each product is known (see Figure 6), E_p, the *JS* system can be aggregated into a single-step *AJS* system for each product *p* utilizing the approximation formulation proposed in Section 2.1.3. The *AJS* system for each product *p* is illustrated in Figure 8 with the corresponding parameters, which are approximated using Eqs. (37) to (47).

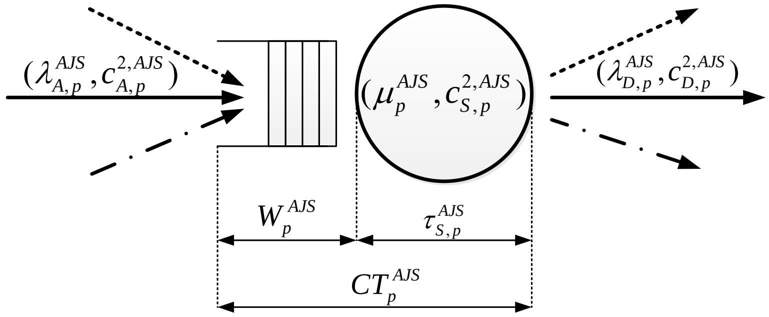
$(\lambda_{A,p}^{AJS}, c_{A,p}^{2,AJS})$
$(\mu_p^{AJS}, c_{S,p}^{2,AJS})$
$(\lambda_{D,p}^{AJS}, c_{D,p}^{2,AJS})$
W_p^{AJS}
$\tau_{S,p}^{AJS}$
CT_p^{AJS}

Figure 8. *AJS* system

$$\lambda_{A,p}^{AJS} = \lambda_{A,pi} \qquad \forall p, i = e_{p1} \tag{37}$$

$$c_{A,p}^{2,AJS} = c_{A,pi}^{2} \qquad \forall p, i = e_{p1} \tag{38}$$

$$c_{D,p}^{2,AJS} = c_{D,pi}^{2} \qquad \forall p, i = e_{Np} \tag{39}$$

$$c_{S,p}^{2,AJS} \approx \frac{\sum_{\substack{i\in E_p \\ i\geq e_{p2}}} \left[\frac{\left(c_{A,p}^{2,AJS} + c_{S,p}^{2,AJS}\right)^2}{4} \frac{\left(\rho_p^{AJS}\right)^2 \tau_{S,p}^{AJS}}{\left(1-\rho_p^{AJS}\right)^2} \times G_p^{AJS}\right] + \sum_{i\in E_p} \sum_{r\in R_i} \left(\frac{\lambda_{A,ri}}{\lambda_{A,\mathbb{P}i}}\right) \sigma_{S,ri}^2}{\sum_{\substack{i\in E_p \\ i\geq e_{p2}}} W_{pi} + \sum_{i\in E_p} \tau_{S,p}^{AJS}} \qquad \forall p \tag{40}$$

$$\mu_p^{AJS} = \min_{i\in E_p} \frac{1}{\tau_{S,\mathbb{P}i}} \qquad \forall p \tag{41}$$

$$\tau_{S,p}^{AJS} = \sum_{\substack{i\in E_p \\ i\geq e_{p2}}} W_{pi} + \sum_{i\in E_p} \tau_{S,\mathbb{P}i} \qquad \forall p \tag{42}$$

$$\rho_p^{AJS} = \frac{\lambda_{A,p}^{AJS}}{\mu_p^{AJS}} \qquad \forall p \tag{43}$$

$$W_p^{AJS} \approx \frac{\rho_p^{AJS}/\mu_p^{AJS}}{1-\rho_p} \times \frac{c_{A,p}^{2,AJS} + c_{S,p}^{2,AJS}}{2} \times G_p^{AJS} \qquad \forall p \tag{44}$$

$$G_p^{AJS} = \begin{cases} \exp\left(-\frac{2}{3} \times \frac{1-\rho_p^{AJS}}{\rho_p^{AJS}} \times \frac{\left(1-c_{A,p}^{2,AJS}\right)^2}{c_{A,p}^{2,AJS} + c_{S,p}^{2,AJS}}\right), & 0 \leq c_{A,p}^{AJS} \leq 1 \\ \exp\left(-\left(1-\rho_p^{AJS}\right) \times \frac{c_{A,p}^{2,AJS} - 1}{c_{A,p}^{2,AJS} + c_{S,p}^{2,AJS}}\right), & c_{A,p}^{AJS} > 1 \end{cases} \qquad \forall p \tag{45}$$

$$CT_p^{AJS} = W_p^{AJS} + \tau_{S,p}^{AJS} \qquad \forall p \tag{46}$$

$$L_p^{AJS} = \lambda_{A,p}^{AJS} CT_p^{AJS} \qquad \forall p \tag{47}$$

3. Numerical results

Various experiments have been conducted to demonstrate the validity and the numerical accuracy of the proposed aggregated models. The QAG models were implemented in MATLAB 2014 and executed on an Intel Pentium IV PC. The QAG models are executed in less than a second for each scenario. The results from the analytical aggregated models were compared with those obtained from a simulation model built using AnyLogic (www.anylogic.com) and executed on the same PC. The simulation experiments for the smallest scenario took at least 4 hours on average. The simulation results were recorded based on 20 independent simulation runs with a 95% confidence interval. This ensured that the standard deviation of the throughput value from different replications is within ±0.5% of the mean. Each run represented the production of at least 20,000 units of product. The statistics corresponding to the first 2,000 units were neglected to account for the transient start-up effects.

In all the simulation runs, the random parameters of the models such as inter-arrival times and service times were generated using a Gamma distribution with given mean and SCV. The KPIs recorded in all experiments were the average waiting time and the average cycle time at the processing steps and (if applicable) for each product. To determine the numerical accuracy of the proposed QAG models, percentage errors between the QAG model results and the simulation results (SIM) of the KPIs are computed. The percentage error in each KPI (i.e., waiting time and cycle time) is calculated as Eq. (48). In the following, the results from two sets of experiments are reported.

$$\Delta_{KPI} = \frac{KPI^{(SIM)} - KPI^{(QAG)}}{KPI^{(QAG)}} \times 100 \tag{48}$$

3.1. Experiment 1: A single-class PS system

In this experiment, the performance of a single-class PS system is analyzed that contains M = 4 parallel production lines and N = 3 processing steps in each line. Once a unit arrives at the system, it selects the first free production line, and each unit visits processing steps 1, 2 and 3 sequentially. Figure 9 shows the single-class PS system in this experiment with the given service rate for each processing step. Other parameters (i.e., arrival-rate and SCVs) are set through different scenarios labeled 1-24. Table 1 shows the settings and the corresponding results for each scenario of the QAG model, the detailed simulation (SIM) model and the EPT approach [28]. In all the scenarios, μ_1 = 5, μ_2 = 4 and μ_3 = 6. In addition, one server (c_i = 1) has been considered for all processing steps in all scenarios. Finally, we consider $\lambda_{\max}$ = 5 for each scenario. Accordingly, the throughput (utilization) rate can be calculated as $\lambda/\lambda_{\max}$ for each scenario.

Table 1. Results of Experiment 1

Sc. #	$[c_A^2, c_{S,1}^2, c_{S,2}^2, c_{S,3}^2]$	λ	Waiting Time					Cycle Time				
			Value (E-05)			Δ (%)		Value			Δ (%)	
			QAG	EPT	SIM	QAG	EPT	QAG	EPT	SIM	QAG	EPT
1	[.05,.05,.05,0.5]	1.00	0.423	0.423	0.424	.02	.02	0.528	0.528	0.529	.03	.03
2	[.5,.5,.5,.5]	1.00	3.148	3.141	3.151	.06	.09	0.616	0.615	0.617	.07	.10
3	[.75,1,.75,1]	1.00	433.1	432. 6	433.8	.07	.10	0.637	0.636	0.638	.07	.12
4	[1.5,2,2,1.5]	1.00	2796	2795	2798	.08	.11	1.160	1.158	1.161	.09	.13
5	[.05,.05,.05,0.5]	1.50	5.565	5.564	5.571	.09	.10	0.515	0.514	0.516	.09	.10
6	[.5,.5,.5,.5]	1.50	45.91	45.11	45.95	.09	.10	0.618	0.616	0.619	.10	.10
7	[.75,1,.75,1]	1.50	1037	1036	1038	.04	.09	0.678	0.677	0.679	.06	.13
8	[1.5,2,2,1.5]	1.50	5269	5267	5274	.11	.13	1.676	1.675	1.679	.15	.16
9	[.05,.05,.05,0.5]	2.00	84.60	84.56	84.68	.09	.14	0.579	0.578	0.580	.09	.15
10	[.5,.5,.5,.5]	2.00	231	230	232	.12	.16	0.625	0.623	0.626	.13	.18
11	[.75,1,.75,1]	2.00	2097	2096	2100	.11	.19	0.756	0.754	0.757	.14	.21
12	[1.5,2,2,1.5]	2.00	9478	9473	9494	.17	.22	2.530	2.527	2.535	.19	.25
13	[.05,.05,.05,0.5]	2.50	98.65	98.53	98.75	.10	.22	0.609	0.608	0.610	.11	.25
14	[.5,.5,.5,.5]	2.50	786	784	787	.15	.29	0.646	0.643	0.647	.16	.31
15	[.75,1,.75,1]	2.50	4163	4155	4168	.13	.31	0.894	0.891	0.896	.16	.34
16	[1.5,2,2,1.5]	2.50	17645	17606	17674	.17	.38	4.013	4.004	4.021	.21	.42
17	[.05,.05,.05,0.5]	3.00	157	156	158	.18	1.1	0.651	0.643	0.652	.20	1.3
18	[.5,.5,.5,.5]	3.00	2072	2049	2077	.29	1.3	0.703	0.693	0.705	.33	1.6
19	[.75,1,.75,1]	3.00	8152	8044	8175	.28	1.6	1.168	1.150	1.173	.38	1.9
20	[1.5,2,2,1.5]	3.00	33493	32968	33607	.34	1.9	6.871	6.753	6.898	.39	2.1
21	[.05,.05,.05,0.5]	3.50	547	537	549	.27	2.2	0.746	0.731	0.748	.30	2.3
22	[.5,.5,.5,.5]	3.50	7854	7658	7879	.32	2.8	0.912	0.889	0.916	.42	2.9
23	[.75,1,.75,1]	3.50	24720	23972	24816	.39	3.4	2.003	1.939	2.012	.44	3.6
24	[1.5,2,2,1.5]	3.50	99468	95100	99896	.43	4.8	15.112	14.401	15.184	.48	5.1
25	[.05,.05,.05,0.5]	3.75	3820	3523	3846	.75	9.23	0.886	0.810	0.895	.98	10.5
26	[.5,.5,.5,.5]	3.75	20134	17904	20416	1.4	12.3	1.348	1.200	1.373	1.9	12.6
27	[.75,1,.75,1]	3.75	59290	51952	60061	1.3	13.5	3.641	3.199	3.729	2.4	14.2
28	[1.5,2,2,1.5]	3.75	237566	202265	241367	1.6	16.2	31.538	26.656	32.389	2.7	17.7
29	[.05,.05,.05,0.5]	3.95	52220	44150	52795	1.1	19.5	2.316	1.937	2.359	1.84	21.8
30	[.5,.5,.5,.5]	3.95	119964	92447	122123	1.8	24.3	4.848	3.692	5.004	3.2	26.2
31	[.75,1,.75,1]	3.95	339439	255073	346567	2.1	26.4	16.640	12.400	17.173	3.2	27.8
32	[1.5,2,2,1.5]	3.95	1357998	987947	1397380	2.9	29.3	165.130	115.698	171.405	3.8	32.5

The results in Table 1 provides insights regarding the impact of different input parameters such as SCV of the inter-arrival time (c_A^2), SCV of the processing times (i.e., $c_{S,1}^2, c_{S,2}^2, c_{S,3}^2$) and arrival rates (λ) on the KPIs such as the waiting time (*W*) and the cycle time (*CT*) of the *APS* system. As seen in Table 1, the estimates of waiting times and cycle times, for the QAG model, are within 2.9% and 3.8% of the simulation estimates. These values are 10.6% and 11.1% for the EPT approach. In all the scenarios, the proposed QAG model provides better estimation (i.e., closer values to the simulation estimations) for both the waiting time and the cycle time in comparison to the EPT approach. These results show the advantage of the proposed QAG model compared to the EPT approach. The results also provide several design and performance insights for the *APS* system. Comparing results for scenarios 1 and 2 (1 and 3 or 2 and 3), respectively, it is observed that an increase in the SCVs results in an increase in the waiting time as well as the cycle time for the *APS* system. On the other hand, the increase in the arrival rate λ leads to a significant increase of the waiting time and the cycle time as long as the arrival rates λ are smaller than the minimum service rate among processing steps 1, 2 and 3. This increase in the waiting time and the cycle time is accompanied by a corresponding increase in the queue lengths or WIP inventory at processing steps 1, 2 and 3 and particularly processing step 2 that is the bottleneck step since it has the lowest service rate (μ=4). It is worth mentioning that the EPT approach requires at least 6 hours for running each scenario while the computational time for the proposed QAG model is smaller than one second.

Similar insights are obtained by comparing the results of other scenarios when the corresponding parameters are increased. Results of the final scenarios (e.g., Sc. #20 to Sc. #24) clearly indicate the impact of arrival rates on the performance measures, wherein the high rate of arrivals results in a significant increase of the waiting time and the cycle time.

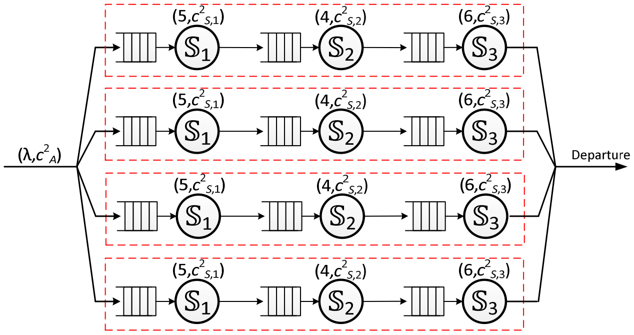

Figure 9. Experiment 1: A 3-step single-class *PS* system

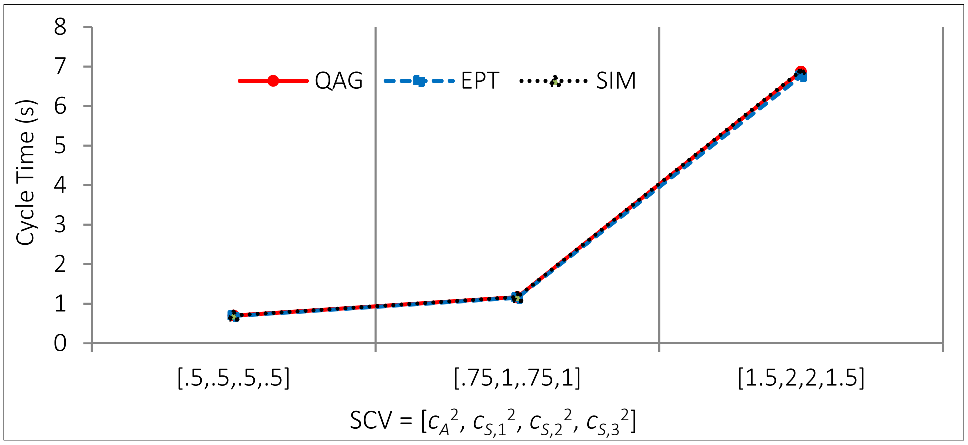

Figure 10. SCV vs. Cycle time (λ = 2)

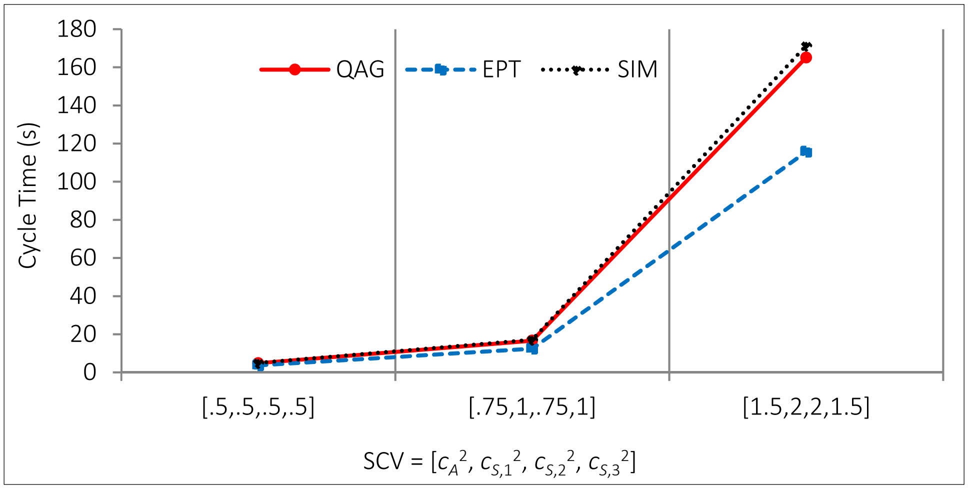

Figure 11. SCV vs. Cycle time (λ = 3.95)

Figures 10 and 11 illustrate the impact of SCV on the cycle time of the APS system. It is observed that an increase of the SCVs leads to an increase of the cycle time of the APS system. Comparing Figures 10 and 11 shows that the larger the values of the arrival rate λ, the more sensitive the cycle time of the APS system to an increase of the SCVs. In addition, the performance of the QAG model is superior to EPT even with large values of the SCVs.

Figures 12 and 13 illustrate the impact of λ on the waiting time and the cycle time of the APS system, respectively. It is observed that an increase of λ leads to a significant increase in the cycle time of the APS system. The QAG model yields more accurate estimates of the waiting time and the cycle time of the APS system in comparison to the EPT approach.

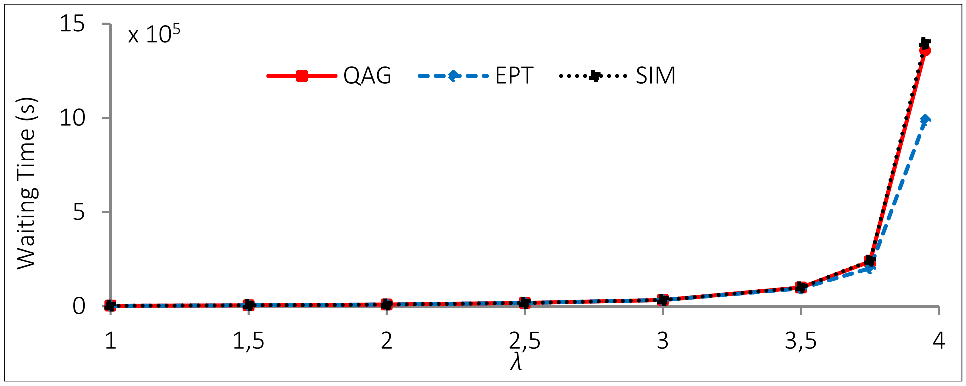

Figure 12. Arrival rate λ vs. Waiting time: SCV = [1.5,2,2,1.5]

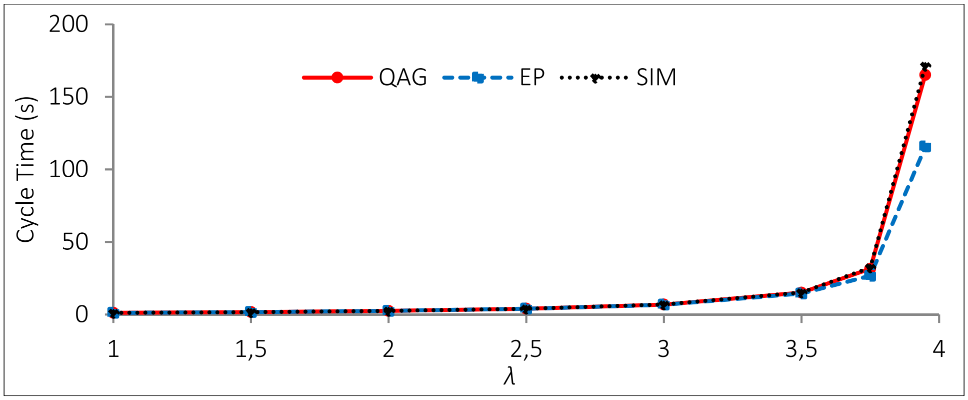

Figure 13. Arrival rate λ vs. Cycle time: SCV = [1.5,2,2,1.5]

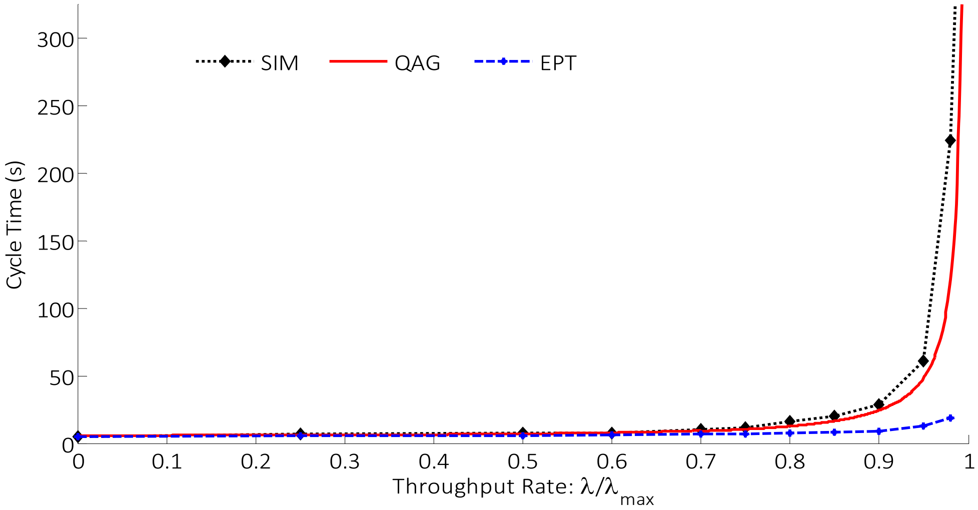

Figure 14. CT-TH curve

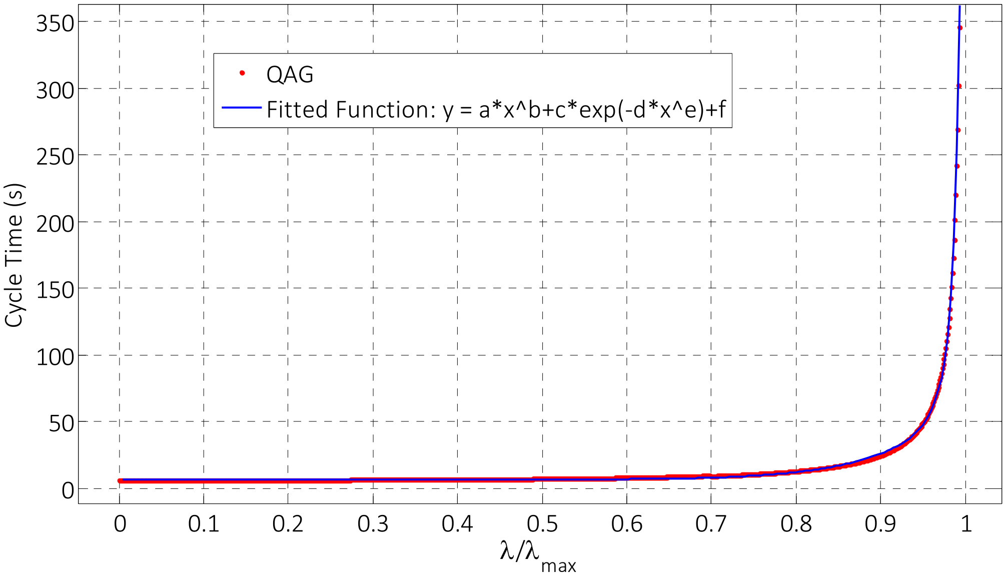

Figure 15. Fitted function to the CT-TH curve of the QAG model

Table 2. Fitted function on the CT-TH curve: Experiment 1

General Model	Approximation		Goodness of fit
	Coefficient	95% confidence bounds	
$y = \boldsymbol{a}x^{\boldsymbol{b}} + \boldsymbol{c}\exp(-\boldsymbol{d}x^{\boldsymbol{e}}) + \boldsymbol{f}$	*a* = 56.05 *b* = 10.4 *c* = 25.83 *d* = -3.471 *e* = 49.02 *f* = -19.2	(52.71, 59.39) (10.05, 10.75) (5.284, 46.38) (-4.202, -2.739) (36.48, 61.57) (-39.74, 1.338)	SSE: 881.1 R-square: 0.9988 Adjusted R-square: 0.9988 RMSE: 0.9448

Figure 14 depicts the performance curve CT-TH of the APS system wherein the cycle time diagram is illustrated versus the throughput rate. It can be seen that the proposed QAG model provides enough accurate estimation of the cycle time compared to the EPT approach. Figure 15 shows the fitted function on the CT-TH curve of the QAG model results. The parameters of the fitted function have been approximated as Table 2. It can be concluded that the cycle time is the sum of two polynomial and exponential functions of the throughput rate.

3.2. Experiment 2: A multiple-class JS system

In this experiment, the performance of the multiple-class *JS* system of Figure 16 is analyzed. This system is a cluster tool that contains *N* = 6 identical processing steps that serve *P* = 3 different products (P_1 to P_3). The products enter the equipment from the *In-port* and leave the equipment by the *Out-port* once their process finishes. The processing routes E_p of the products are: $E_1 = \{\mathbb{S}_1, \mathbb{S}_3, \mathbb{S}_5, \mathbb{S}_6\}$, $E_2 = \{\mathbb{S}_1, \mathbb{S}_2, \mathbb{S}_4, \mathbb{S}_6\}$ and $E_3 = \{\mathbb{S}_3, \mathbb{S}_4, \mathbb{S}_5\}$. Accordingly, the set of products that utilize each processing step are: $R_1 = \{P_1, P_2\}$, $R_2 = \{P_2\}$, $R_3 = \{P_1, P_3\}$, $R_4 = \{P_2, P_3\}$, $R_5 = \{P_1, P_3\}$ and $R_6 = \{P_1, P_2\}$. The parameters are set through different scenarios labeled 1-20. Table 4 shows the settings and the corresponding results for each scenario of the QAG and detailed simulation (SIM) models. In all the scenarios, the service rates are determined based on Table 3. The results for the 20 scenarios of Table 4 provide insights regarding the impact of different input parameters such as arrival-rate and SCVs on the cycle time of each product.

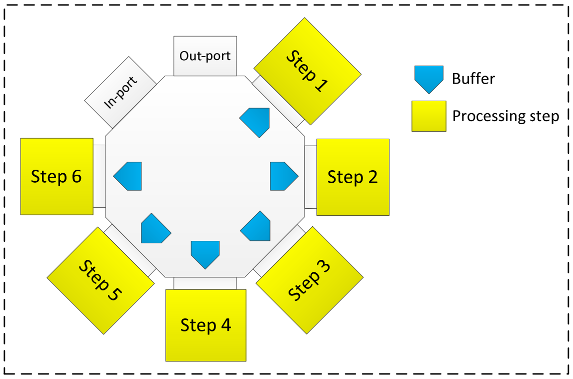

Figure 16. Sample multiple-class JS system

Table 3. Service rate interval at each processing step and for each product

Product	Service rates interval (LB, UB)*					
	1	2	3	4	5	6
P_1	(0.5, 5)	-	(0.3, 3)	-	(0.3, 2)	(0.5, 4)
P_2	(1, 3)	(0.3, 3)	-	(0.5, 2)	-	(0.4, 4)
P_3	-	-	(0.2, 2)	(0.5, 5)	(0.2, 3)	-

*LB: Lower bound, UB: Upper bound

As seen in Table 4, the estimates of cycle times for the QAG model are at most of 3.62% of the simulation estimates. The results provide several performance insights for the *AJS* system. By comparing the result of each pair of scenarios 1, 5, 9, 13 and 17, it is observed that an increase in the SCVs results in an increase of the cycle time for the *AJS* system. On the other hand, an increase in the arrival rates leads to a significant increase in the cycle time as long as the ρ_{pi} remains lower than 1. Similar to Experiment 1, this increase of the cycle time is accompanied by a corresponding increase of the queue lengths or WIP inventory of the products in the processing steps.

Figures 17 and 18 illustrate the impact of SCV on the cycle time of each product. It is observed that an increase of the SCVs leads to an increase in the cycle time of all the products although product P_2 is more sensitive. Comparing Figures 17 and 18 shows that the larger the arrival rates, the more sensitive is the cycle time of *AJS* system to an increase of the SCVs. Figures 19 and 20 illustrate the impact of arrival rates on the cycle time of different products. It is observed that an increase in the arrival rates leads to a significant increase in the cycle time of all the products. The effect of the arrival rate increase is propagated for larger values of the SCVs.

Table 4. Results of Experiment 2

Sc. #	SCVs Interval [IB, UB]	Arrival Rates interval [IB, UB]	Cycle Time Estimation								
			QAG			SIM			Δ (%)		
			P_1	P_2	P_3	P_1	P_2	P_3	P_1	P_2	P_3
1	[0.2, 0.4]	[0.1, 0.7]	1.383	1.377	0.997	1.384	1.378	0.998	0.01	0.02	0.01
2	[0.2, 0.4]	[0.7, 1.2]	1.998	2.057	1.326	1.999	2.058	1.327	0.08	0.07	0.08
3	[0.2, 0.4]	[1.2, 1.6]	50.02	17.33	13.96	50.08	17.36	13.99	0.12	0.15	0.18
4	[0.2, 0.4]	[1.5, 2]	273.5	1378.7	1241.8	275.1	1387.4	1248.1	0.52	0.63	0.51
5	[0.5, 1]	[0.1, 0.7]	1.548	1.540	1.080	1.549	1.541	1.081	0.02	0.03	0.02
6	[0.5, 1]	[0.7, 1.2]	2.748	2.900	1.694	2.752	2.905	1.697	0.13	0.16	0.18
7	[0.5, 1]	[1.2, 1.6]	124.33	42.78	32.86	124.86	43.01	33.02	0.43	0.54	0.49
8	[0.5, 1]	[1.5, 2]	686.3	3577.3	3041.8	695.5	3626.6	3085.3	1.35	1.38	1.43
9	[1, 1.5]	[0.1, 0.7]	1.917	1.906	1.266	1.918	1.907	1.267	0.08	0.07	0.08
10	[1, 1.5]	[0.7, 1.2]	4.62	5.02	2.58	4.63	5.03	2.589	0.26	0.29	0.37
11	[1, 1.5]	[1.2, 1.6]	352.8	121.5	88.2	355.8	122.6	88.9	0.87	0.91	0.78
12	[1, 1.5]	[1.5, 2]	1962	10454	8340	1991	10610	8475.6	1.52	1.49	1.62
13	[1.5, 2]	[0.1, 0.7]	2.97	2.94	1.78	2.98	2.95	1.784	0.25	0.24	0.24
14	[1.5, 2]	[0.7, 1.2]	11.09	12.27	5.44	11.16	12.35	5.478	0.59	0.62	0.71
15	[1.5, 2]	[1.2, 1.6]	1337	461	316	1353	467	320.7	1.21	1.32	1.41
16	[1.5, 2]	[1.5, 2]	7465	40449	30294	7624	41294	30951	2.12	2.09	2.17
17	[2, 3]	[0.1, 0.7]	3.18	3.15	1.88	3.21	3.18	1.901	0.91	0.86	1.09
18	[2, 3]	[0.7, 1.2]	12.57	13.93	6.08	12.71	14.11	6.151	1.12	1.22	1.18
19	[2, 3]	[1.2, 1.6]	1584	547	373	1619	560	381.1	2.18	2.39	2.28
20	[2, 3]	[1.5, 2]	8848	48014	35733	9159	49660	37026	3.51	3.43	3.62

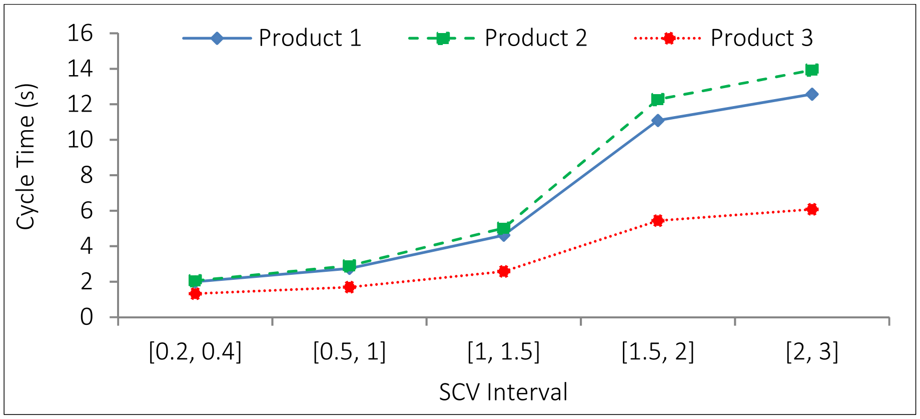

Figure 17. SCVs vs. Cycle Time: $\lambda \in [0.7, 1.2]$

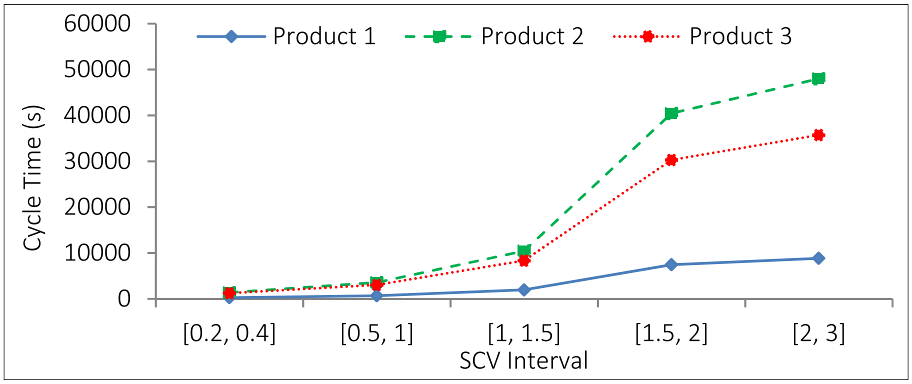

Figure 18. SCVs vs. Cycle Time: $\lambda \in [1.5, 2]$

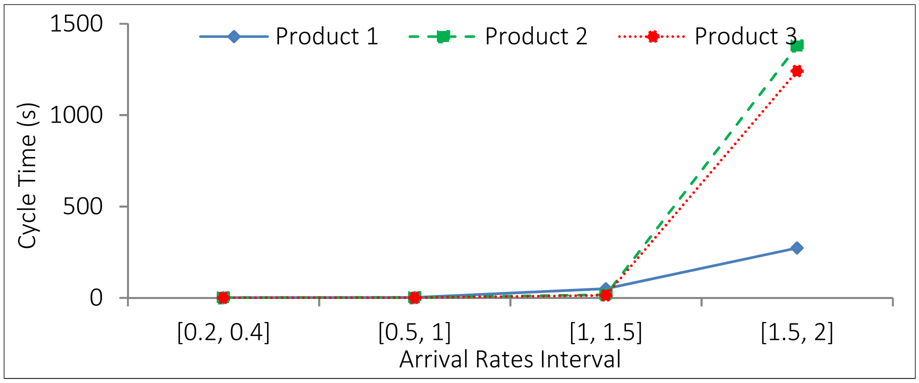

Figure 19. Arrival Rates vs. Cycle Time: SCV ∈ [0.2, 0.4]

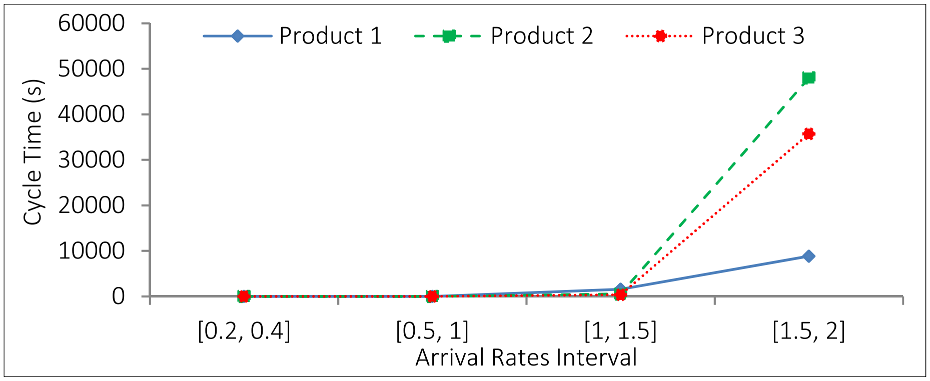

Figure 20. Arrival Rates vs. Cycle Time: SCV ∈ [2, 3]

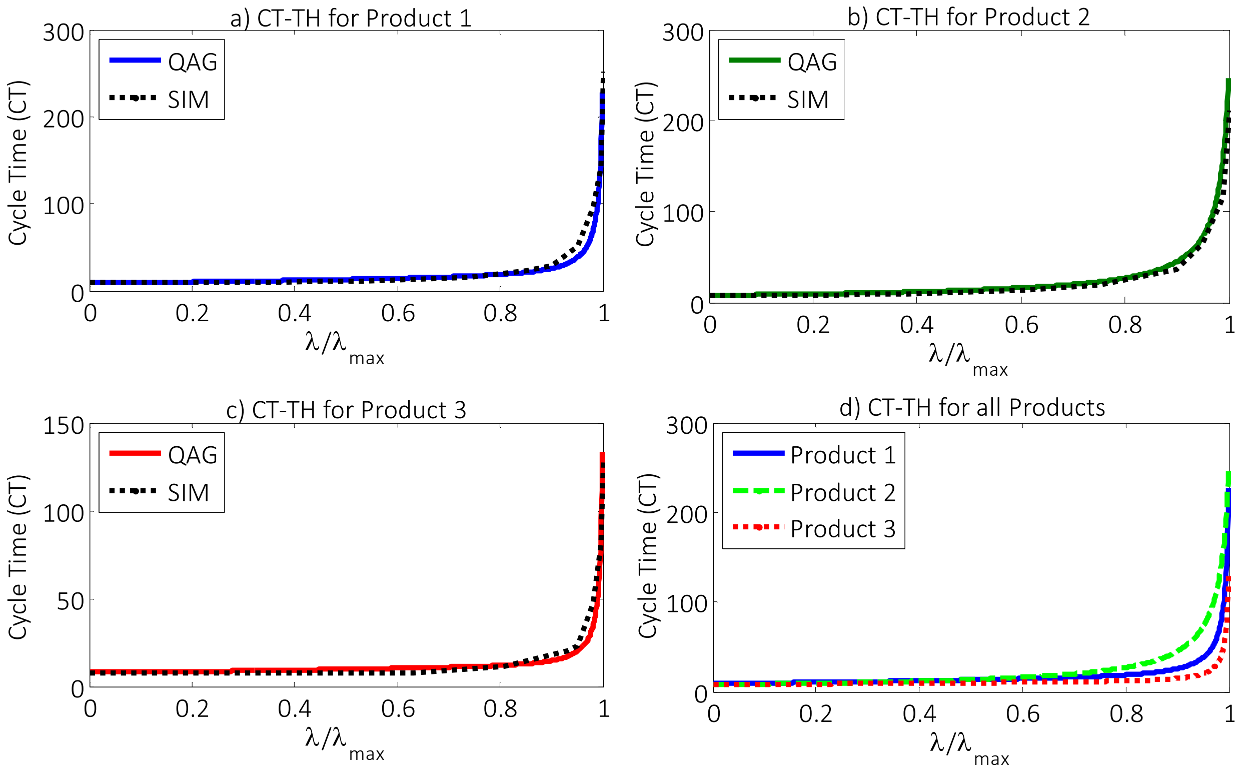

Figure 21. CT-TH curves for all products

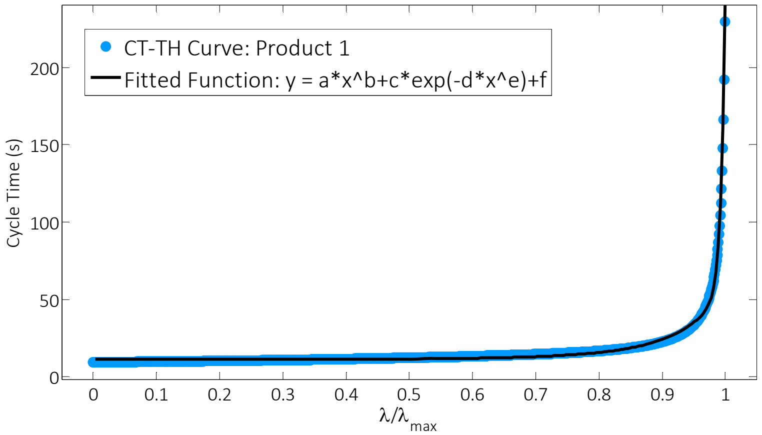

Figure 22. Fitted function to the CT-TH curve of the QAG model: Product 1

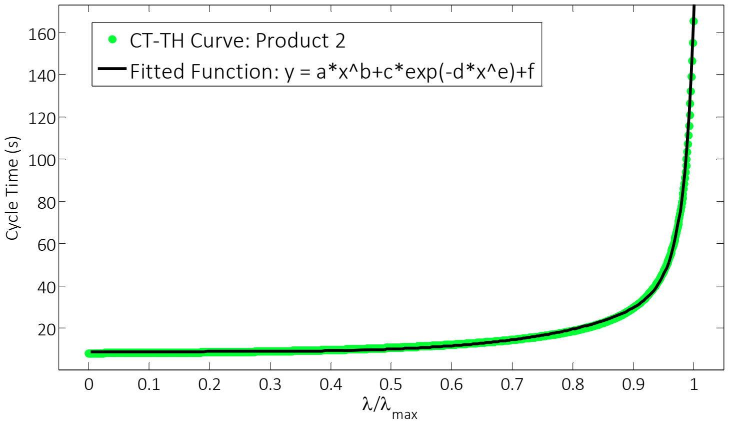

Figure 23. Fitted function to the CT-TH curve of the QAG model: Product 2

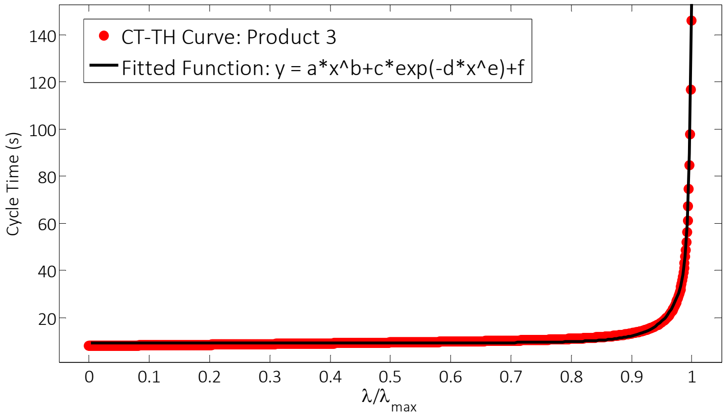

Figure 24. Fitted function to the CT-TH curve of the QAG model: Product 3

Figures 21a to 21c depict the CT-TH curve for each product in the AJS system. It can be seen that the proposed QAG model provides enough accurate and close estimation of the cycle time compared to the simulation model. It can be seen that the proposed QAG model has a quite accurate estimation of the cycle time for throughput rates lower than 80%. Figure 21d shows the CT-TH curves for all the products, and it can be observed that product P_2 has the largest cycle time in most cases.

Similar to Figure 15 in Experiment 1, Figures 22 to 24 show the fitted function on the CT-TH curves for products 1 to 3, respectively. Table 5 provides the approximated parameters of the fitted functions for each product. The cycle time for each product is the sum of two identical polynomial and exponential functions of the throughput rate.

Table 5. Fitted function on the CT-TH curves of all products: Experiment 2

Product	General Model	Approximation		Goodness of fit
		Coefficient	95% confidence bounds	
Product 1	$y = ax^b + c\exp(-dx^e) + f$	a = 26.28	(24.74, 27.83)	**SSE**: 1627
		b = 6.621	(6.281, 6.962)	**R-square**: 0.9935
		c = 11.36	(-5.477, 28.2)	**Adjusted R-square**: 0.9935
		d = -2.944	(-4.344, -1.544)	**RMSE**: 1.279
		e = 48.22	(26.55, 69.9)	
		f = -0.4142	(-17.23, 16.41)	
Product 2		a = 27.55	(26.12, 28.98)	**SSE**: 372.8
		b = 4.392	(4.239, 4.544)	**R-square**: 0.9988
		c = 11.95	(2.128, 21.77)	**Adjusted R-square**: 0.9988
		d = -2.469	(-3.228, -1.711)	**RMSE**: 0.6124
		e = 21.91	(16.08, 27.73)	
		f = -3.015	(-12.82, 6.79)	
Product 3		a = 126.2	(123.9, 128.5)	**SSE**: 550.6
		b = 222.4	(214.9, 230)	**R-square**: 0.9924
		c = 0.7435	(-1.351, 2.838)	**Adjusted R-square**: 0.9924
		d = -3.738	(-6.425, -1.05)	**RMSE**: 0.7443
		e = 6.368	(1.296, 11.44)	
		f = 8.368	(6.298, 10.44)	

Sensitivity analysis

This section analyzes the sensitivity between products. This sensitivity is studied by fixing the throughput of a product and investigating the effect of increasing the arrival rate of other products on the fixed-throughput product. This analysis is also conducted by fixing the throughput rate of two products. The processing time of each product in each processing step is considered as Table 6. Table 7 provides six scenarios wherein the throughput rate of the different set of products is fixed. For instance, in scenario 1, the throughput rates of products P_1 and P_2 are considered equal to 75% and sensitivity is performed on product P_2.

Table 6. The processing time of each product at each processing step

Product	Processing time (s)					
	1	2	3	4	5	6
P_1	2	-	2.5	-	3	1
P_2	1	3	-	2	-	2.5
P_3	-	-	3	2	4	-

Table 7. Scenarios for sensitivity analysis

Scenario #	Throughput rate			Results
	P_1	P_2	P_3	
1	75%	75%	varies from 0 to 1	Figure 24a
2	75%	varies from 0 to 1	75%	Figure 24b
3	varies from 0 to 1	75%	75%	Figure 24c
4	75%	varies from 0 to 1	varies from 0 to 1	Figure 24d
5	varies from 0 to 1	75%	varies from 0 to 1	Figure 24e
6	varies from 0 to 1	varies from 0 to 1	75%	Figure 24f

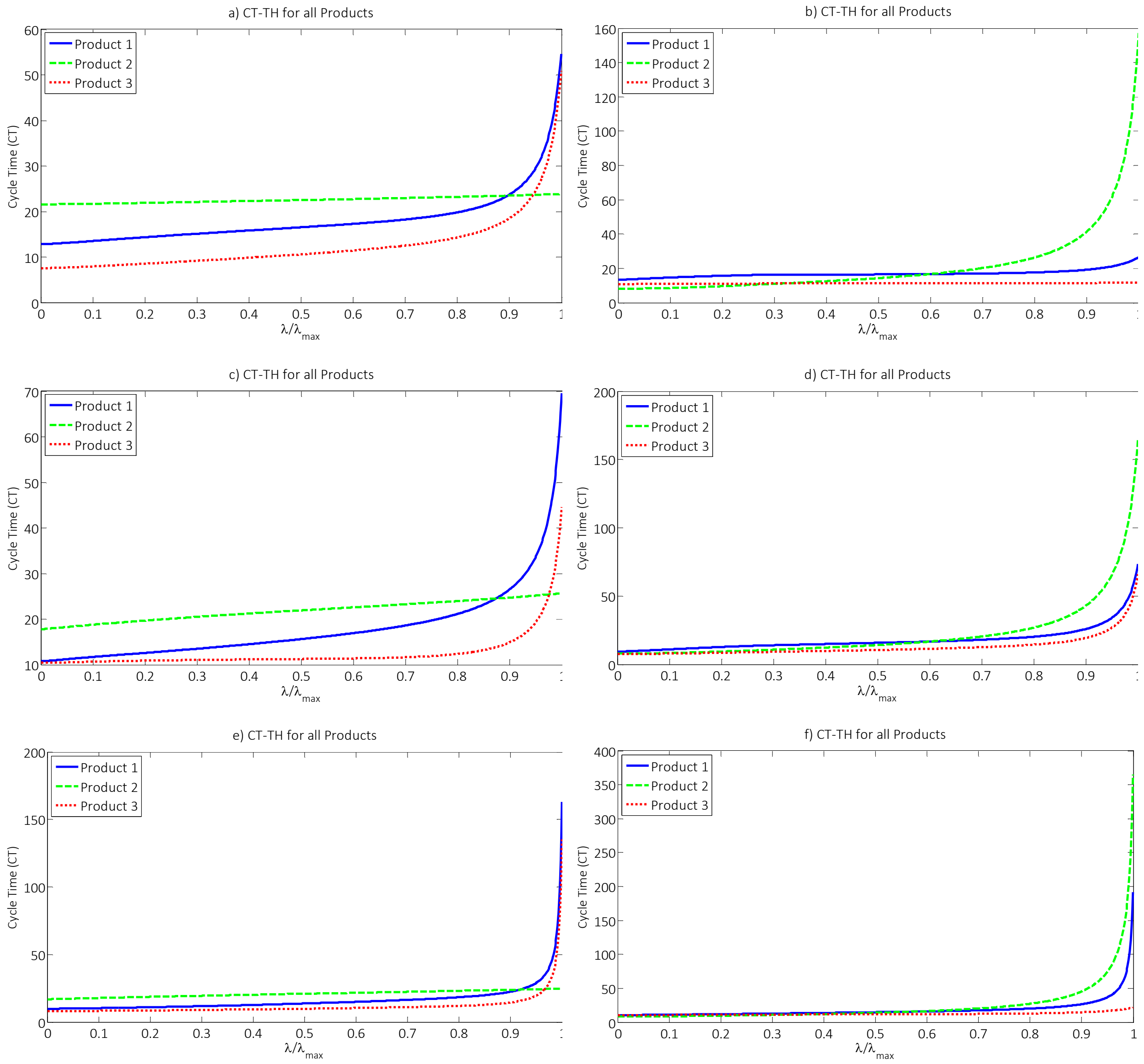

Figure 24. Sensitivity analysis

Looking at Figure 24a (Scenario 1 in Table 7), one can see that, when fixing the throughput rates of products P_1 and P_2, increasing the throughput rate of product P_3 only affects the cycle time of product P_1. The underlying reason is that products P_1 and P_3 share the same bottleneck processing step 5. Accordingly, since the throughput rate of products P_3 increases, the WIP in processing step 5 increases and consequently the waiting time of product P_1 in processing step 5 is increasing as well, which results in a larger cycle time for product P_1. On the other hand, the cycle time of product P_2 is not much affected by the throughput of product P_3 since product P_2 has processing step 2 as a bottleneck. This is the same explanation for Figure 24c corresponding to scenario 3 such that products P_1 and P_3 are interdependent but looking at both Figures 24a and 24c, one can see that product P_1 is more influenced by product P_3 but not the opposite. According to Table 6, it can be seen that products P_1 and P_3 have the largest processing times in both processing steps 3 and 5, while product P_3 has larger values. Accordingly, increasing the throughput rate of product P_3 creates longer queues (higher WIP) at processing steps 3 and 5 compared to product P_1. Therefore, we can conclude that the cycle times of products P_1 and P_3 are more sensitive to the throughput of product P_3 than product P_1. Figure 24b shows that when the throughput rates of products P_1 and P_3 are fixed, increasing the throughput rate of product P_2 does not affect the cycle time of products P_1 and P_3 since the bottleneck processing step corresponding to product P_2 is different than those of products P_1 and P_3. Figure 24d shows that increasing the throughput rates of both products P_1 and P_3 does not affect much the cycle time of product P_2. This again illustrates the independence between products P_1 and P_3 and product P_2.

4. Conclusion

This paper proposed a new aggregation model based on queueing network by modeling each workstation as a GI/G/m queue and then aggregating multiple workstations into a single-step workstation. The parameters of the aggregated workstation are approximated based on the parameters of the original workstations.

Two production systems were studied in this paper: A single-class parallel-series production system and a multi-class job-shop production system. Comprehensive numerical experiments were conducted to show the validity and the performance of the proposed aggregation models compared to the recently proposed simulation-based models of the literature. Numerical experiments indicate that the proposed aggregation model is computationally efficient and yields fairly accurate results when compared to the literature. After proving the validity of the proposed aggregation models, valuable insights were provided by analyzing the sensitivity of the multi-class job-shop production system to the input parameters. Having the aggregation models enables us to quickly analyze the performance of the system and to make decisions.

Some interesting future research directions are provided below:

- The proposed aggregation models can be improved by incorporating other shop floor factors such as workstation breakdowns, processing setups, and scheduled maintenance periods. Incorporating these factors will make the aggregation models closer to real settings.
- The performance of the QAG model could be compared or coupled with other approximation methods for multi-class manufacturing systems. As an example, the robust approximation approach of Bandi et al. [38] can be utilized in the QAG model to provide better performance, in particular for high throughput rates.
- One way of modeling processing steps with finite buffer is using a GI/G/c/K queuing model. In this case, any product arriving at the full buffer is ignored and leaves the system. However, in semiconductor manufacturing systems, products are stored in additional buffers and never leaves the system. Once a new place in the finite buffer becomes available, one product is transferred to the buffer. Without loss of generality, we can consider that buffer sizes are actually infinite. Therefore, the proposed QAG model is suitable for mass production systems where products are small enough and there is no storage limitation. On the other hand, proposing a QAG model for finite buffer workstations with storage limitations can be interesting.
- Another important component that can contribute to cycle time is the material handling system. It performs as an interconnector for workstations and should deliver the right amount of materials to the right place, at the right time and at a minimal cost. The material handling system may impact the performance of production systems, mainly by affecting the WIP, i.e. an efficient material handling system reduces the WIP and the cycle time is then shortened due to the decreased waiting time. In wafer manufacturing facilities, the material handling system contributes between 15% and 70% of the total operating cost and around 20% of the product cycle time [39]. Hence, in such settings, it is important to consider material handling when developing cycle time prediction models. Different factors could be considered such as processing several products at the same time, predicted/unpredicted breakdowns of workstations and of the material handling system, vehicle blockage and dispatching rules.

Acknowledgments

This research was partially supported by the French national research and cooperative development program Nano 2017. The work in this paper was performed while the first author was a post-doctoral fellow at "*Ecole des Mines de Saint-Etienne*".

References

[1] C. P. L. Veeger, L. F. P. Etman, E. Lefeber, I. J. B. F. Adan, J. van Herk, and J. E. Rooda, "Predicting Cycle Time Distributions for Integrated Processing Workstations: An Aggregate Modeling Approach," *IEEE Trans. Semicond. Manuf.*, vol. 24, no. 2, pp. 223–236, May 2011.

[2] J. R. Morrison and D. P. Martin, "Cycle Time Approximations for the G/G/m Queue Subject to Server Failures and Cycle Time Offsets with Applications," in *The 17th Annual SEMI/IEEE ASMC 2006 Conference*, 2006, pp. 322–326.

[3] A. A. A. Kock, L. F. P. Etman, and J. E. Rooda, "Effective process times for multi-server flowlines with finite buffers," *IIE Trans.*, vol. 40, no. 3, pp. 177–186, Jan. 2008.

[4] D. P. Martin, “Capacity and cycle time-throughput understanding system (CAC-TUS) an analysis tool to determine the components of capacity and cycle time in a semiconductor manufacturing line,” in *10th Annual IEEE/SEMI. Advanced Semiconductor Manufacturing Conference and Workshop. ASMC 99 Proceedings (Cat. No.99CH36295)*, 1999, pp. 127–131.

[5] M. Manitz, “Analysis of assembly/disassembly queueing networks with blocking after service and general service times,” *Ann. Oper. Res.*, vol. 226, no. 1, pp. 417–441, Mar. 2015.

[6] Whitt Ward, “APPROXIMATIONS FOR THE GI/G/m QUEUE,” *Prod. Oper. Manag.*, vol. 2, no. 2, pp. 114–161, Jan. 2009.

[7] J. R. Morrison and D. P. Martin, “Performance evaluation of photolithography cluster tools,” *Spectr.*, vol. 29, no. 3, pp. 375–389, Jul. 2007.

[8] J. R. Morrison and D. P. Martin, “Practical Extensions to Cycle Time Approximations for the G/G/m-Queue With Applications,” *IEEE Trans. Autom. Sci. Eng.*, vol. 4, no. 4, pp. 523–532, Oct. 2007.

[9] B. Vahdani and M. Mohammadi, “A bi-objective interval-stochastic robust optimization model for designing closed loop supply chain network with multi-priority queuing system,” *Int. J. Prod. Econ.*, vol. 170, pp. 67–87, Dec. 2015.

[10] Y. Dallery and Y. Frein, “On Decomposition Methods for Tandem Queueing Networks with Blocking,” *Oper. Res.*, vol. 41, no. 2, pp. 386–399, 1993.

[11] M. K. Govil and M. C. Fu, “Queueing theory in manufacturing: A survey,” *J. Manuf. Syst.*, vol. 18, no. 3, pp. 214–240, Jan. 1999.

[12] H. TEMPELMEIER and M. BÜRGER, “Performance evaluation of unbalanced flow lines with general distributed processing times, failures and imperfect production,” *IIE Trans.*, vol. 33, no. 4, pp. 293–302, Apr. 2001.

[13] A. Arisha and P. Young, “Intelligent simulation-based lot scheduling of photolithography toolsets in a wafer fabrication facility,” in *Proceedings of the 2004 Winter Simulation Conference, 2004.*, 2004, vol. 2, pp. 1935–1942 vol.2.

[14] N. Nayani and M. Mollaghasemi, “Validation and verification of the simulation model of a photolithography process in semiconductor manufacturing,” in *1998 Winter Simulation Conference. Proceedings (Cat. No.98CH36274)*, 1998, vol. 2, pp. 1017–1022 vol.2.

[15] N. G. Pierce and M. J. Drevna, “Development of generic simulation models to evaluate wafer fabrication cluster tools,” *Proc. 24th Conf. Winter Simul.*, pp. 874–878, Dec. 1992.

[16] M. Thürer, M. Stevenson, C. Silva, and T. Qu, “Drum-buffer-rope and workload control in High-variety flow and job shops with bottlenecks: An assessment by simulation,” *Int. J. Prod. Econ.*, vol. 188, pp. 116–127, Jun. 2017.

[17] M. Calle, P. L. González-R, J. M. Leon, H. Pierreval, and D. Canca, “Integrated management of inventory and production systems based on floating decoupling point and real-time information: A simulation-based analysis,” *Int. J. Prod. Econ.*, vol. 181, pp. 48–57, Nov. 2016.

[18] F. Yang, B. E. Ankenman, and B. L. Nelson, “Estimating Cycle Time Percentile Curves for Manufacturing Systems via Simulation,” *Inf. J. Comput.*, vol. 20, no. 4, pp. 628–643, Jul. 2008.

[19] E. J. Chen, “Metamodels for Estimating Quantiles of Systems with One Controllable Parameter,” *SIMULATION*, vol. 85, no. 5, pp. 307–317, May 2009.

[20] R. J. Brooks and A. M. Tobias, “Simplification in the simulation of manufacturing systems,” *Int. J. Prod. Res.*, vol. 38, no. 5, pp. 1009–1027, Mar. 2000.

[21] R. T. Johnson, J. W. Fowler, and G. T. Mackulak, “A discrete event simulation model simplification technique,” in *Proceedings of the Winter Simulation Conference, 2005.*, 2005, pp. 5 pp.-.

[22] O. Rose, “Why do simple wafer fab models fail in certain scenarios?,” in *2000 Winter Simulation Conference Proceedings (Cat. No.00CH37165)*, 2000, vol. 2, pp. 1481–1490 vol.2.

[23] W. J. Hopp and M. L. Spearman, *Factory Physics: Foundations of Manufacturing Management*. Irwin, 1996.

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

[25] J. H. Jacobs, L. F. P. Etman, J. E. Rooda, and E. J. J. V. Campen, “Quantifying operational time variability: the missing parameter for cycle time reduction,” in *2001 IEEE/SEMI Advanced Semiconductor Manufacturing Conference (IEEE Cat. No.01CH37160)*, 2001, pp. 1–10.

[26] J. H. Jacobs, L. F. P. Etman, E. J. J. van Campen, and J. E. Rooda, “Characterization of operational time variability using effective process times,” *IEEE Trans. Semicond. Manuf.*, vol. 16, no. 3, pp. 511–520, Aug. 2003.

[27] K. Wu and K. Hui, “The Determination and Indetermination of Service Times in Manufacturing Systems,” *IEEE Trans. Semicond. Manuf.*, vol. 21, no. 1, pp. 72–82, Feb. 2008.

[28] C. P. L. Veeger, L. F. P. Etman, J. van Herk, and J. E. Rooda, “Generating Cycle Time-Throughput Curves Using Effective Process Time Based Aggregate Modeling,” *IEEE Trans. Semicond. Manuf.*, vol. 23, no. 4, pp. 517–526, Nov. 2010.

[29] J. R. Morrison, “Equipment models for fab level production simulation: Practical features and computational tractability,” in *Proceedings of the 2009 Winter Simulation Conference (WSC)*, 2009, pp. 1581–1591.

[30] F. Jarrahi and W. Abdul-Kader, “Performance evaluation of a multi-product production line: An approximation method,” *Appl. Math. Model.*, vol. 39, no. 13, pp. 3619–3636, Jul. 2015.

[31] M. J. Sharma and S. J. Yu, “Stepwise regression data envelopment analysis for variable reduction,” *Appl. Math. Comput.*, vol. 253, pp. 126–134, Feb. 2015.

[32] J. MacGregor Smith, “Queue decomposition & finite closed queueing network models,” *Comput. Oper. Res.*, vol. 53, pp. 176–193, Jan. 2015.

[33] H. Baumann and W. Sandmann, “Multi-server tandem queue with Markovian arrival process, phase-type service times, and finite buffers,” *Eur. J. Oper. Res.*, vol. 256, no. 1, pp. 187–195, Jan. 2017.

[34] L. Sattler, “Using queueing curve approximations in a fab to determine productivity improvements,” in *IEEE/SEMI 1996 Advanced Semiconductor Manufacturing Conference and Workshop. Theme-Innovative Approaches to Growth in the Semiconductor Industry. ASMC 96 Proceedings*, 1996, pp. 140–145.

[35] K. Wu, “An examination of variability and its basic properties for a factory,” *IEEE Trans. Semicond. Manuf.*, vol. 18, no. 1, pp. 214–221, Feb. 2005.

[36] F. Yang, B. Ankenman, and B. L. Nelson, “Efficient generation of cycle time-throughput curves through simulation and metamodeling,” *Nav. Res. Logist. NRL*, vol. 54, no. 1, pp. 78–93.

[37] K. Wu and L. McGinnis, “Performance evaluation for general queueing networks in manufacturing systems: Characterizing the trade-off between queue time and utilization,” *Eur. J. Oper. Res.*, vol. 221, no. 2, pp. 328–339, Sep. 2012.

[38] C. Bandi, D. Bertsimas, and N. Youssef, “Robust Queueing Theory,” *Oper. Res.*, vol. 63, no. 3, pp. 676–700, Apr. 2015.

[39] D. Nazzal and L. F. McGinnis, “Analytical approach to estimating AMHS performance in 300 mm fabs,” *Int. J. Prod. Res.*, vol. 45, no. 3, pp. 571–590, Feb. 2007.

[40] D. Nazzal, “A closed queueing network approach to analyzing multi-vehicle material handling systems,” *IIE Trans.*, vol. 43, no. 10, pp. 721–738, Oct. 2011.

[41] Y.-M. Tu, C.-W. L. P. D, and A. H. I. Lee, “AMHS capacity determination model for wafer fabrication based on production performance optimization,” *Int. J. Prod. Res.*, vol. 51, no. 18, pp. 5520–5535, Sep. 2013.

[42] H. S. Lee, A. Bouhchouch, Y. Dallery, and Y. Frein, “Performance evaluation of open queueing networks with arbitrary configuration and finite buffers,” *Ann. Oper. Res.*, vol. 79, no. 0, pp. 181–206, Jan. 1998.

[43] K. Satyam and A. Krishnamurthy, “Performance evaluation of a multi-product system under CONWIP control,” *IIE Trans.*, vol. 40, no. 3, pp. 252–264, Jan. 2008.

[44] K. Wu and L. McGinnis, “Interpolation approximations for queues in series,” *IIE Trans.*, vol. 45, no. 3, pp. 273–290, Mar. 2013.

[45] K. Satyam, A. Krishnamurthy, and M. Kamath, “Solving general multi-class closed queuing networks using parametric decomposition,” *Comput. Oper. Res.*, vol. 40, no. 7, pp. 1777–1789, Jul. 2013.

[46] R. Schelasin, “Using static capacity modeling and queuing theory equations to predict factory cycle time performance in semiconductor manufacturing,” in *Proceedings of the 2011 Winter Simulation Conference (WSC)*, 2011, pp. 2040–2049.

[47] W. Whitt, “The Queueing Network Analyzer,” *Bell Syst. Tech. J.*, vol. 62, no. 9, pp. 2779–2815, Nov. 1983.

[48] K. T. Marshall and R. V. Evans, “Some Inequalities in Queuing,” *Oper. Res.*, vol. 16, no. 3, pp. 651–668, 1968.

[49] J. Kingman, “The heavy traffic approximation in the theory of queues,” *Symposium on Congestion Theory*, University of North Carolina Press, Chapel Hill, NC, 1965.

[50] W. Kramer and L.-B. Manfred, “Approximate formulae for general single systems with single and batch arrivals,” *Proc. 8th Int. Teletraffic Congr.*, vol. 2, no. 3, pp. 235-1-235–8, 1976.

[51] W. Whitt, “The Best Order for Queues in Series,” *Manag. Sci.*, vol. 31, no. 4, pp. 475–487, Apr. 1985.

[52] D. P. Heyman, “A diffusion model approximation for the GI/G/1 queue in heavy traffic,” *Bell Syst. Tech. J.*, vol. 54, no. 9, pp. 1637–1646, Nov. 1975.

[53] J. Köllerström, “Heavy Traffic Theory for Queues with Several Servers. I,” *J. Appl. Probab.*, vol. 11, no. 3, pp. 544–552, 1974.

[54] O. J. Boxma, J. W. Cohen, and N. Huffels, “Approximations of the Mean Waiting Time in an M/G/s Queueing System,” *Oper. Res.*, vol. 27, no. 6, pp. 1115–1127, 1979.

[55] T. Kimura, “Approximations for the delay probability in the M/G/s queue,” *Math. Comput. Model.*, vol. 22, no. 10, pp. 157–165, Nov. 1995.

[56] G. R. Bitran and D. Tirupati, “Multiproduct Queueing Networks with Deterministic Routing: Decomposition Approach and the Notion of Interference,” *Manag. Sci.*, vol. 34, no. 1, pp. 75–100, Jan. 1988.