Louvain School
of Management

Working paper
The Bounding Discrete Phase-Type Method
06/07

Jean-Sébastien TANCREZ & Pierre SEMAL

Academic Year 2005 - 2006

Louvain School of Management

THE BOUNDING DISCRETE PHASE–TYPE METHOD

JEAN–SÉBASTIEN TANCREZ[†] [‡] AND PIERRE SEMAL[†]

Abstract. Models of production systems have always been essential. They are needed at a strategic level in order to guide the design of production systems but also at an operational level when, for example, the daily load and staffing have to be chosen.

Models can be classified into three categories: analytical, simulative and approximate. In this paper, we propose an approximation approach that works as follows. Each arrival or service distribution is discretized using the same time step. The evolution of the production system can then be described by a Markov chain. The performances of the production system can then be estimated from the analysis of the Markov chain.

The way the discretization is carried on determines the properties of the results. In this paper, we investigate the "grouping at the end" discretization method and, in order to fix ideas, in the context of production lines. In this case, upper and lower bounds on the throughput can be derived. Furthermore, the distance between these bounds is proved to be related to the time step used in the discretization. They are thus refinable and their precision can be evaluated a priori. All these results are proved using the concept of critical path of a production run.

Beside the conceptual contribution of this paper, the method has been successfully applied to a line with three stations in which three buffer spaces have to be allocated. Nevertheless, the complexity and solution aspects will require further attention before making the method eligible for real large scale problems.

Key words. Production line, Discretization, Bounds, Critical path.

AMS subject classifications. 60J20, 60K25, 90B22, 90B30

1. Introduction.

Production systems follow various types of organization, among which the job–shop and the flow line are most typical [5, 12]. In this paper, we focus on production lines although we strongly believe the approach and the method presented here can be applied to any production system. The details of the production line we study are described below in §1.1.

Models of production systems have always been essential. They are needed during the design phases, when the following elements have to be selected: the type of production equipments and their power, the layout of the different stations and the mechanisms of synchronization (equipment, transfer lot size, buffer size, ...). At a more operational level, models are also needed to provide support for daily production decisions like the load, the sequence of jobs or the necessary staffing, and for customer oriented decisions like accepting a new job or promising some delivery time. Refer to [4] and [2] for an overview of manufacturing systems models. A systematic review of the literature for models of production lines is given below in §1.2.

The model presented in this paper is approximate in the sense that it simplifies the system to be studied, without changing its general structure, in order to facilitate the evaluation of its performance. The model has two peculiar features: it provides not only an approximation of the performance of interest but bounds on it, and these lower and upper bounds can be tightened. This will be illustrated by determining bounds on the throughput of a production line, which are valid both in the transient and in steady–state. The approach seems to be applicable to other performances like the buffer utilization, the job flow time or the work in progress.

[†] Université Catholique de Louvain, Place des Doyens, 1, 1348 Louvain–la–Neuve, Belgium, {tancrez,semal@poms.ucl.ac.be}.

[‡] Facultés Universitaires Catholiques de Mons.

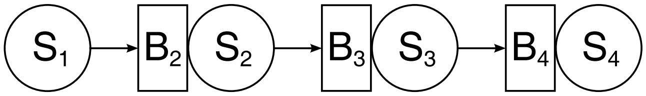

FIG. 1. *Production line including four stations and three buffers.*

Practically, the paper is organized as follows. Here below, §1.1 details the type of production lines we focus on and §1.2 provides a short literature review of models used for such lines. Section 2 and 3 are the heart of the paper. Section 2 describes the BDPH method and §3 states its properties. Section 4 provides an example. Finally, §5 focuses on possible extensions of the method.

1.1. Production Lines. This paper focuses on production lines with asynchronous part transfer, in other terms tandem queueing systems. As shown on Figure 1, production lines have a very special structure. It is a linear network of m service stations $(S_1, S_2, \ldots, S_m)$ separated by $m-1$ buffers storages $(B_2, B_3, \ldots, B_m)$. The manufacturing of one item consists in the sequential processing of this item by the stations S_1 to S_m. The item enters the system at station S_1 and leaves at S_m. After its processing by a station, let us say S_i, the unit is stored in the buffer B_{i+1} if a space is available. There it waits until the next station, S_{i+1}, finishes its job on the previous item and gets rid of it. At this moment, the processing of the unit on station S_{i+1} starts. This is repeated until the item gets its last processing at station S_m and then leaves the system.

Such lines experience productivity losses due to blocking and starving. First, a station is said to be blocked when it cannot get rid of an item because the next buffer is full. Second, a station is to be starved when it cannot begin to work on a new item because the previous buffer is empty. Increasing buffer sizes allows to limit these productivity losses.

Here is the list of assumptions we make on the lines we analyze, in order to fix ideas. None of these assumptions is restrictive.

- *General finite service time distributions.* The service times are generally distributed but finite. Successive processing times are independent and identically distributed. The finiteness assumption is not restrictive since it is always the case in practice.
- *Finite buffer sizes.* We do not make any assumption on the buffers sizes except their finiteness.
- *Infinite arrival.* With this assumption, the first station is never starved. This assumption can be relaxed by using an initial station that models the arrival process.
- *Infinite demand.* With this assumption, the last station is never blocked. Again, this assumption can be relaxed by using a last station that models the real demand or the real storage space.
- *Blocking after service.* This means that if the next station is full, a job is blocked in its current station after having been processed. A blocking before service policy can be modeled by adapting the buffer size.

1.2. Models for Production Lines. Several good comprehensive reviews of models for production lines are available (see for example [2], [4], [17], [7], [10], and [16]). In order to situate precisely our method in the state of the art, let us review the main approaches used for the evaluation of production line performance. As already said, there are three kinds of models: exact analytical models, approximate analytical models and simulations.

Exact analytical models are the richest since they allow a direct and exact understanding of the influence of a decision variable on the performance of interest. Unfortunately, most production systems are too complex to be modeled analytically. In our case, exact models can be used for simple production lines only. Three methods are significant:

- *Closed–form models.* Closed–form results are available for very simple configurations, e.g. two stations lines and exponentially (or sometimes Erlang) distributed service times.
- *State models.* These models build continuous (rarely discrete) Markov chains to analyze lines with exponential or phase–type distributions. Based on the identified state space, a transition matrix is derived and the stationary equations are solved numerically to obtain the steady–state probabilities. The main difficulty lies in the explosion of the state space size. A first paper [13] dates from 1967.
- *Holding time models.* Introduced by Muth, this method aims to be computationally more efficient than state models for tandem queues without intermediate buffers. It considers the sequence of holding times (blocking time added to processing time) for successive jobs at each station, and constructs recursive relationships. For an overview, see [15].

Simulation is at the other end of the continuum. It does not rely on any assumption and is therefore very general. Almost all systems can be simulated. The weakness of the simulation approach mainly lies in its development cost. An example of simulation for production lines is given in [8].

Approximate analytical models are in between: the system to be analyzed is simplified in order to be analytically modeled. This keeps the development costs low. However, the uncertainty about the results is the weakness of the approach. The method presented in this paper is approximate but tries to give certainties about the results, by proving bounds. Here are the two main approximate methods:

- *Decomposition.* The idea is to decompose a system into smaller subsystems. Solving more, but much easier, subproblems, allows to approximately analyze the global system much more quickly. A set of equations that determines the unknown parameters and the links between subsystems is first derived. An iterative procedure is then used to solve the equations. This method has initially been created for exponentially distributed production lines, see [9] and [3] for examples or [6] for a good review. Some authors then extended it to lines with phase–type processing times, see [1] for continuous PH and [11] for discrete PH.
- *Expansion.* The Generalized Expansion Method (GEM) is also based on the idea of decomposing the system, but it adds the concept of an artificial node that registers the blocked jobs. For a description and an example of application, see [14].

2. The Bounding Discrete Phase–Type Method. The method presented in this paper works as follows. First, each distribution is discretized using the same time step. The evolution of the production system can then be described by a Markov chain whose states describe the stages of the different service centers and the current utilizations of the various storage areas (buffers). The performance of the production system can then be estimated from the analysis of the Markov chain. The production

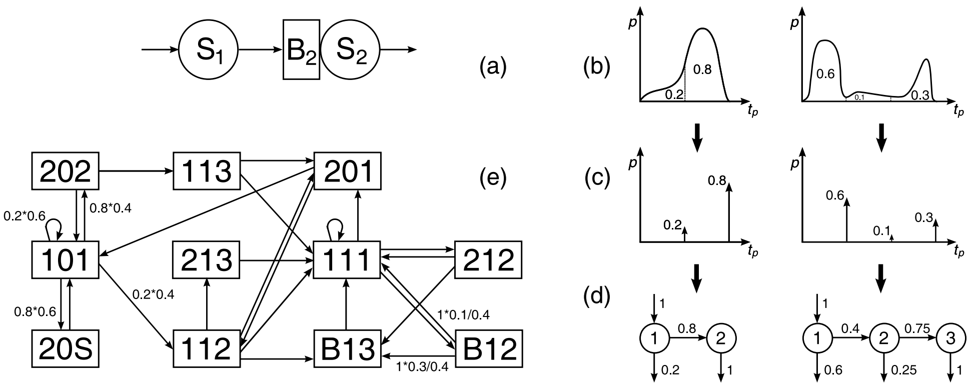

FIG. 2. *Stages of the BDPH method applied to a two station line: discretization of the original service time distributions by "grouping at the end", PH representation and Markov Chain.*

rate and the buffer utilization, for example, can be derived from the steady–state probabilities. Transient characteristics can also be determined.

The originality comes from the discretization method we use. Using a discretization step τ, we transform the original distributions into a discrete one by concentrating the probability mass distributed in the interval $[k\tau, (k+1)\tau]$ on a single value. The choice of this value is open. Here, we chose a "grouping at the end" principle, that is, the probability mass is carried forward on the point $(k+1)\tau$. This discretization creates a bias, as each job length is increased. However, it has the advantage to keep an intelligible link with the original distributions.

The method is best illustrated on an example. Let us look at a simple line made of two stations separated by a buffer of size one depicted on Figure 2.a. Figure 2.b shows the exact service time distributions for the two servers. The discretized distributions by "grouping at the end" are shown in Figure 2.c and their phase–type representation in Figure 2.d.

The behavior of the complete system can now be modeled by a Markov Chain, i.e., using a state model. The Markov chain given in Figure 2.e lists all the possible recurrent states of the system (the first symbol refers to the first station, the second to the buffer and the third to the second station) and the possible transitions between these states. Each station can be starved (S), blocked (B) or in some stage of service (1 means, for example, that the station already spent one time step working on the current job). Each buffer is described by its utilization (0 or 1 with a buffer of size one). For example, state B12 means that the first station is blocked, that the buffer is full and that the second station already worked during two time steps on the current job. Two transitions are possible from B12, depending if the second station continues to work on the same job or ends. In the first case, the new state will be B13. The probability of this transition is 0.3/0.4, the probability that the processing time is greater or equal to three knowing that it is greater or equal to two. It is easily deduced from the discrete distribution given in Figure 2.c. In the second case, the second station ends his job, picks up the next item in the buffer and begin to work on it. The first station can thus get rid of its blocking item and begins a new job. The new state is thus 111.

Transient performances, like the throughput at some time t, can be derived from the matrix of transition probabilities. The steady–state performances can be computed from the steady–state probabilities derived from the Markov chain. Steady–

state productivity, work in progress or buffer utilization, for example, can be approximated. The size of the Markov chain[1] and the complexity problems are not addressed in this paper. They require attention in the future.

The method is approximate. However, it has the advantage to offer some theoretical control. Indeed, it is proven here below that the time needed to achieve any production target will be overestimated by the method. Furthermore, the amount by which this time is overestimated can also be bounded. Finally, these bounds can be tightened, by reducing the time step τ.

The method is named "Bounding Discrete Phase–type" (BDPH) since it relies on a discrete phase type approximation of the various distributions of the system to be studied and since it leads to bounds.

3. Properties. In this section, we try to better understand the behavior of the production line and the effect of the discretization by "grouping at the end". We indeed have the intuition that the BDPH method leads to a pessimistic estimation of the productivity. In the next two subsections, we will lay the foundations that allow this result to be formally proved. These foundations rely on the concept of critical path.

3.1. Structural Properties. Let us consider a random infinite real time production run. To construct such a run, we only need a sequence of random processing times drawn according to the original service distributions. The jobs then find their places in the run according to the structure of the production system. We simply denote this infinite real time production run by r. We have:

$$\{\, l^r(W_{i,k})\} \overset{\Delta}{\longmapsto} r,$$

where $W_{i,k}$ denotes the job k at station i and $l^r(W_{i,k})$ the time it takes in this particular run r.

If the original service times $l^r(W_{i,k})$ are discretized, giving the discretized service times $\overline{l^r(W_{i,k})}$, we get the discretized production run, denoted $\overline{r}$:

$$\{\, l^r(W_{i,k})\} \overset{\text{disc.}}{\longmapsto} \{\, \overline{l^r(W_{i,k})}\} \overset{\Delta}{\longmapsto} \overline{r}.$$

When using the "grouping at the end" discretization, we have, $\forall\, r, i, k$:

$$\overline{l^r(W_{i,k})} \triangleq \left\lceil \frac{l^r(W_{i,k})}{\tau} \right\rceil \tau, \tag{1}$$

with, by construction, the following property, $\forall\, r, i, k$:

$$\overline{l^r(W_{i,k})} - \tau \leq l^r(W_{i,k}) \leq \overline{l^r(W_{i,k})}. \tag{2}$$

Before stating a first result, we define the moment a job is started and the moment it is ended. Obviously, a job length, in a particular run r, is given by the difference between these moments:

$$l^r(W_{i,k}) = t^r_{end}\,(W_{i,k}) - t^r_{start}\,(W_{i,k}).$$

Since the production system has a definite structure, i.e., a line in our case, these moments are subject to structural constraints, stated in the following lemma.

[1] The size of the Markov chain is, in first approximation, proportional to $(a+2)^m(b+1)^{m-1}$, with m the number of machines, a the number of steps and b the buffer size.

Lemma 1 (Structural properties of a production line). *Given a production line including a buffer of size b_i before each station i, the jobs verify the following inequalities, $\forall\, r, k$:*

$$
\begin{array}{lrll}
(3) & t^r_{start}\,(W_{i,k}) \geq & t^r_{end}\,(W_{i-1,k}) & \forall\, i \geq 2 \\
(4) & \geq & t^r_{end}\,(W_{i,k-1}) & \forall\, i \\
(5) & \geq & t^r_{end}\left(W_{\begin{array}{ll} i+1, & k-b_{i+1}-2 \\ i+2, & k-b_{i+1}-b_{i+2}-3 \\ i+3, & k-b_{i+1}-b_{i+2}-b_{i+3}-4 \\ \vdots & \vdots \end{array}}\right) & \begin{array}{l} \forall\, i \leq m-1 \\ \forall\, i \leq m-2 \\ \forall\, i \leq m-3 \\ \vdots \end{array}
\end{array}
$$

with at least one equality.

Intuitively, the proof works as follows. Because of the line structure, a job k can only be started on station i: (3) if its processing on the previous station $i-1$ is ended, (4) if the processing of the previous job $k-1$ in station i is ended and (5) if this previous job $k-1$ is not blocked in station i by some unfinished jobs downstream. Furthermore, since there is no reason to wait once all these conditions are satisfied, $t^r_{start}\,(W_{i,k})$ will be given by the maximum of the right hand sides of Lemma 1. The formal proof is given in the Appendix.

Note that the inequalities of Lemma 1 are valid for any run: the job $W_{i,k}$ cannot be started before all the jobs on the right hand sides are finished. These are just static structural properties of the line, independent of the run. However, which precise job end will trigger the start of job $W_{i,k}$ depends on the processing times and thus on the particular run we consider.

3.2. Productivity and Critical Path. In this subsection, we are interested in the time needed to produce p units in a particular run r. This time is given by $t^r_{end}\,(W_{m,p})$ if we fix, without loss of generality, that the run has been started at time 0[2].

For the determination of $t^r_{end}\,(W_{m,p})$, it is first clear that not all the events of the run r are relevant. From Lemma 1, we see that $t^r_{end}\,(W_{m,p})$ only depends on job p or on previous jobs (by any station). We can thus restrict our attention to the following part of the run r, called its p-part and denoted r_p:

$$\{\, l^r(W_{i,k}) \mid 1 \leq i \leq m, 1 \leq k \leq p\} \overset{\Delta}{\longmapsto} r_p.$$

The length of r_p, denoted $l(r_p)$, equals $t^r_{end}\,(W_{m,p})$, i.e., the time needed to produce p units in r. We thus focus on determining the length of this part of the run.

Second, the structural properties given in Lemma 1 allow us to introduce a useful concept. We define the *critical path* of r_p, $cp(r_p)$, as the sequence of jobs that covers r_p. It can be built quite easily. Starting with the last job that leaves the system, job $W_{m,p}$, we can look which job end, in this precise run, has triggered its start, in other words which inequality of Lemma 1 is satisfied at equality[3]. Repeating this process, we can proceed backward in time until the start time of the run. It is obvious by Lemma 1 that every run r_p has at least one critical path.

[2] The start time may correspond to various situations. In most cases, it will be given by the start of the first job on the first workstation, $t^r_{start}\,(W_{1,1})$. However, arbitrary loaded lines at start time may also be considered. For simplicity, we will just assume that the run is started with no job being partially processed.

[3] It can happen that several equalities are satisfied at the same time. In this case, one of them is chosen arbitrarily (let us say the one corresponding to the preceding state in the same station, see Figure 5).

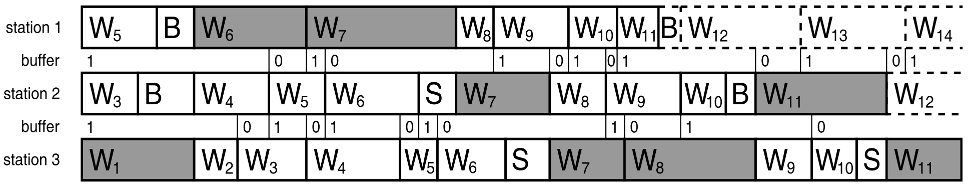

FIG. 3. *Gantt chart of the 11–part of a run on a production line with three stations and buffers of size one. The critical path is given in gray.*

This critical path $cp(r_p)$ has some nice properties. It is a set of jobs $W_{i,k}$ which covers the time of this part of the production run r_p without overlap and without gap. In other words, the length of r_p equals the length of its critical path, denoted $l(cp(r_p))$. It is given by:

$$(6) \qquad l(r_p) = t^r_{end}(W_{m,p}) = l(cp(r_p)) = \sum_{W_{i,k} \in cp(r_p)} l^r(W_{i,k}) = \sum_{j \in cp(r_p)} l^r(j).$$

As already said, the inequalities of Lemma 1 are valid for any run. The absence of overlap in the critical path $cp(r_p)$ is thus a property independent of the considered run. Therefore, in another run, the sequence of jobs $cp(r_p)$ will just be one non–overlapping path (maybe with gaps) whose total length is shorter or equal to the length of the p–part of this other run.

The notion of critical part can be best illustrated on the Gantt chart (see Figure 3) associated to a particular run. In this chart, the time goes from left to right. The state of a station at a given time is represented either by a letter (B for blocked, S for starved) or by the job currently processed. The state of a buffer is represented by the number of jobs waiting inside. For the run depicted, the line is fully loaded at time 0, with one job waiting in each buffer. The critical path (in gray) of the 11–part is made of the following backward sequence:

$$cp(r_{11}) = \{\, W_{3,11}, W_{2,11}, W_{3,8}, W_{3,7}, W_{2,7}, W_{1,7}, W_{1,6}, W_{1,3}\}$$

It can be checked that each couple of successive jobs satisfies one of the inequalities of Lemma 1 at equality.

The concept of critical path offers a useful tool to understand what is happening in a production system. Its ability to cover the time allows to express the time to produce p units in a run r as the sum of job lengths. When an intelligible transformation is operated on the job lengths, i.e., on their distribution, it allows to relate this transformation and its effect on the global length of a production run. We believe the concept of critical path can be generalized to other production systems and used for other transformations of the service distributions.

In the case of production lines and discretization by "grouping at the end", the critical path leads us to the following result, where the p–part of $\overline{r}$ is defined similarly :

$$\{\, \overline{l^r(W_{i,k})} \mid 1 \le i \le m, 1 \le k \le p\} \overset{\Delta}{\longmapsto} \overline{r}_p.$$

LEMMA 2. *The time an m–station line takes to produce p units in a random real time production run r can be bounded as follows:*

$$l(\overline{r}_p) - \tau(p + m - 1) \le l(r_p) \le l(\overline{r}_p).$$

Proof. Using equations (6) and (2) and the fact that $cp(r_p)$ is just a non–overlapping path in the discretized production run (smaller than the critical path, $cp(\overline{r}_p)$), we can write:

$$l(r_p) = \sum_{j \in cp(r_p)} l^r(j) \leq \sum_{j \in cp(r_p)} \overline{l^r(j)} \leq \sum_{j \in cp(\overline{r}_p)} \overline{l^r(j)} = l(\overline{r}_p),$$

that states the right inequality of the lemma. For the left inequality, using the same equations and the fact that $cp(\overline{r}_p)$ is non–overlapping in the original run, we get:

$$l(\overline{r}_p) - \tau |cp(\overline{r}_p)| = \sum_{j \in cp(\overline{r}_p)} (\overline{l^r(j)} - \tau) \leq \sum_{j \in cp(\overline{r}_p)} l^r(j) \leq \sum_{j \in cp(r_p)} l^r(j) = l(r_p),$$

where $|cp(\overline{r}_p)|$ denotes the cardinality of the critical path $cp(\overline{r}_p)$, i.e., the number of jobs making it up. The proof ends by showing that this cardinality is smaller than $p + m - 1$ (see Lemma 8 in the Appendix). □

Lemma 2 provides a major result. Considering any random production run, we have upper and lower bounds on the time it would take to produce a given production. Unfortunately, this result cannot yet be directly used since it refers to a given random production run. For the results to be useful, we need to be able to say something about an average production run. This point is tackled in the next subsection.

3.3. Bounds on the Throughput. Let us first consider the mean time T_P necessary to reach a given production P. By definition,

$$T_P = \int f(r_P) l(r_P) dr_P,$$

where $f(r_P)$ is the density function of the p–part of the production runs. This time can be bounded as follows.

Theorem 3. *The mean time T_P an m–station line takes to produce P units can be bounded on the basis of the information computed by the BDPH method. If $\overline{T}_P$ is the mean time to produce P units using the "grouping at the end" discretized times, we have:*

$$\overline{T}_P - \tau(P + m - 1) \leq T_P \leq \overline{T}_P.$$

The proof detailed in the Appendix relies on Lemma 2 and on the fact that the probabilities of r_P and $\overline{r}_P$ are the same since they are both derived from the same run r.

If we are interested in a fixed time instead of a fixed production, bounds can quite easily be derived from the previous theorem.

Theorem 4. *The mean production P_T produced by an m–station line during a fixed time T can be bounded on the basis of the information computed by the BDPH method. If $\overline{P}_T$ is the mean production during time T using the"grouping at the end" discretized times, and $\overline{P}^*$ is the mean production during discrete time $\overline{T}^*$, where chosen minimal such that $\overline{T}^* - \tau(\overline{P}^* + m - 1) \geq T$, we have:*

$$\overline{P}_T \leq P_T \leq \overline{P}^*.$$

The formal proof given in the Appendix relies on the following simple argument. Since the average time to produce a given production is longer with discretized times than with the original times, the quantity produced in a given time is smaller with discretized times than with the original times.

When focusing on the steady–state productivity, the results get even simpler. Indeed, in this case, simple bounds are derived from Theorem 3. The average time between the completion of two units, in steady-state, is called the cycle time c where $c = lim_{P\to\infty} T_P/P$. If $\overline{c}$ denotes the cycle time measured using the "grouping at the end" discretized times (BDPH method), we have the following result.

Corollary 5 (BDPH bounds for a production line). *When it measures the productivity of a production line in steady–state, the BDPH method is pessimistic, i.e., the cycle time is overvalued. Moreover, the error is smaller than the discretization time step:*

$$\overline{c} - \tau \leq c \leq \overline{c}. \tag{7}$$

The proof is given in the Appendix.

Theorems 3 and 4 and Corollary 5 show that our method allows to bound the productivity, in transient or in steady–state, from below and from above. Moreover, these bounds become tighter and converge to the exact productivity when the discretization step is decreased.

These results also show another feature of the method: the accuracy of the bounds is directly related to the selected discretization step. Of course every accuracy improvement will require additional computational efforts caused by the increase of the state space size.

The BDPH bounds lead to two simple approximations of the cycle time. More precise approximations are goals for future research. Inequality (7) leads quite intuitively to a first approximation.

Approximation 6. *The cycle time of a line can be approximated by:*

$$c \approx \overline{c} - \frac{\tau}{2}.$$

This approximation can be seen as an approximation of the cycle time we would obtain by grouping "at the middle", i.e., by concentrating the probability mass at the middle of the step instead of at the end. More rigorously, it can be seen as a converging approximation of the following better approximation.

Approximation 7. *The cycle time of a line can be approximated, $\forall\, i$, by:*

$$c \approx \overline{c} - e_W(i),$$

with $e_W(i) = E[\overline{l^r(W_{i,k})}] - E[l^r(W_{i,k})]$, the discretization bias on the service distribution of station i.

This result comes from the fact that the cycle time can be divided up into two components: the processing time and the blocking/starving time. A good approximation of the cycle time can thus be obtained by removing the known discretization bias on the service time distribution. When the discretization step τ decreases, the bias tends to $\tau/2$ and Approximation 7 tends to Approximation 6. Note that both approximations converge to the exact cycle time as they are between the converging bounds.

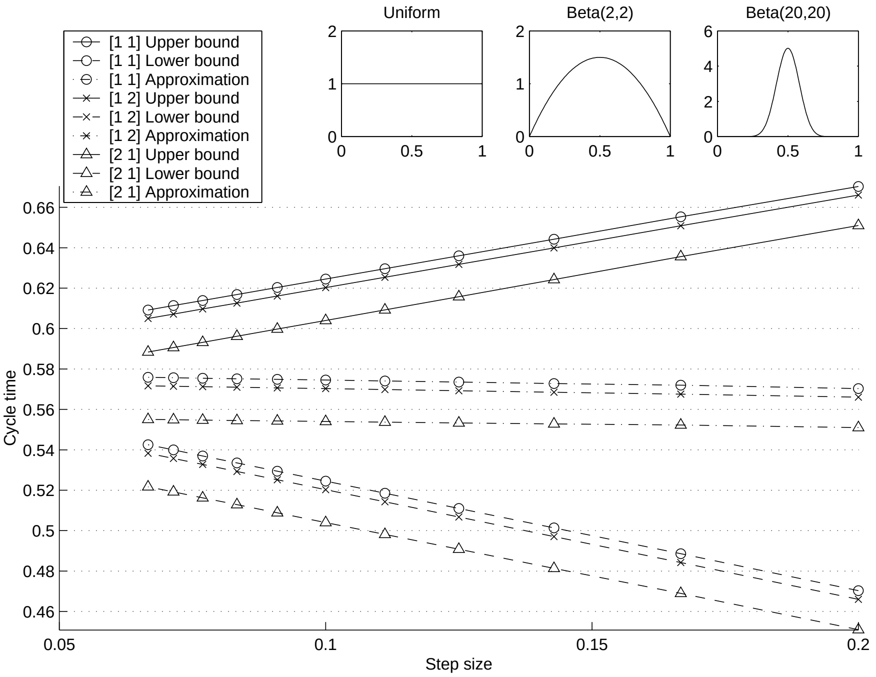

FIG. 4. *Bounds and approximations on the cycle time for a three station line with various buffers configurations. The upper part shows the service distributions, with first station on the left.*

The cycle time in discretized time, $\overline{c}$, can easily be computed from the steady–state probabilities given by the BDPH method. Let us define p_1 as the steady–state probability that a station, the last one for example, is in first stage of service. As every job on this station pass through this first stage during one time step, p_1 equals one step length divided by the cycle time. We get: $\overline{c} = \tau/p_1$.

4. An Example. In this section, we briefly show how the bounding discrete phase–type method performs on a simple example. More realistic examples are not in the scope of this paper.

Let us consider a three station line with the processing times depicted on top of Figure 4. Different buffer configurations are being studied : [1 1], [1 2] and [2 1]. The minimization of the cycle time is the objective here. The chart of Figure 4 gives the bounds on the cycle time for the original buffer configuration and for the two configurations with one more buffer space. The Approximation 7 is also given (with $i = 3$). We see that the bounds and the approximation regularly converge when the step size decreases. Moreover, the accuracy of the approximation can be assessed by comparing it to a simulation result. The [1 1] buffer configuration leads to a cycle time of 0.5786, with a 99% confidence interval given by $[0.5780, 0.5792]$. In this case, the approximation makes an error of 1% with seven steps and 0.47% with fifteen steps. The other buffer configurations lead to similar accuracy.

As expected, the configuration with a buffer of size two in first position turns out to be better, as the beginning of the line is more variable. The benefit in term of productivity can be estimated. Moreover, as the BDPH method offers a quite complete modeling, other performances (like the work–in–progress, i.e., the average number of items in the system, and the buffer occupancy for example) can be estimated.

5. Conclusion and Future Work. In this paper, we presented a new method, called BDPH method, to determine bounds on the performance of a production line. The method relies on a discrete phase–type approximation of the service time distributions with a "grouping at the end" approach. The study of the critical path of a part of a production run allowed the main results to be stated.

In this paper, we sticked to simple production lines and to simple performance in order to present rigorous results. It is clear that the BDPH approach calls for various extensions and future research.

The method has been used to compute bounds on the throughput and on the productivity of a production line. These bounds are valid both in the transient and in the steady–state. It should now be investigated how other performances of interest like the buffer utilization or the average job flow time can be bounded by the BDPH method.

A second direction for future research is related to the "approximation" methods. Indeed, on the basis of the stated BDPH bounds, very obvious approximation methods have been derived (Approximation 6 and 7). A more thorough analysis of the critical path could open the door to more subtle approximation approaches.

In terms of production systems, more complex organization can be studied. Indeed, Lemma 1 states the structural conditions of a line system. In case of an assembly tree, i.e., a set of production lines converging to a unique final workstation, these structural conditions can be very easily updated so that similar conclusions can be drawn. For job–shops, the door is still open. However, the notion of critical path will still constitute the heart of the proofs.

Finally, a large field of research is related to the solution methods to be implemented in order to solve the generated discrete time Markov Chain. It is clear that iterative methods [18] that take advantage of the sparsity of the transition matrix constitute promising ways in that respect. Moreover, the decomposition methods presented in §1.2 offer another way to accelerate the solution.

Appendix. Here are the formal proofs of the results presented in the paper.

Proof of Lemma 1. We first show the three inequalities, simply giving their practical significance:

(3) A station i cannot begin to work on an item k before the preceding station $i-1$ has done his job on this item k.

(4) A station i cannot begin to work on an item k before it has done his job on item $k-1$.

(5) To begin to work on an item k, station i has to finish its job on item $k-1$ (inequality (4)) *and* get rid of it. There is space in the next buffer if station $i+1$ already began to work on item $k-b_{i+1}-1$:

$$t^r_{start}\left(W_{i,k}\right) \geq t^r_{start}\left(W_{i+1,k-b_{i+1}-1}\right).$$

Combining this and inequality (4), we get inequalities (5).

The fact that one of the inequalities is always satisfied at equality comes from the fact that a job always begins due to the end of another job. As each possible state preceding $W_{i,k}$ on the same station corresponds to one of the inequalities (satisfied at equality), the inequalities give all the possibilities:

(3) Before $W_{i,k}$, the station was starved, (3) is thus satisfied at equality: $W_{i,k}$ begins when the previous station pass the item on (see Figure 5.b).

(4) The station was already working previously, (4) is thus satisfied at equality: the station begin to work on item k directly when it ends on $k-1$ (see Figure 5.a).

(5) The station was blocked, it had thus to pass the blocking item on, what is possible when the start of a job in a next station vacate a space in the buffers between. According to the number of blocked states one after the other, it corresponds to one of the inequalities (5) satisfied at equality (see Figure 5.c). □

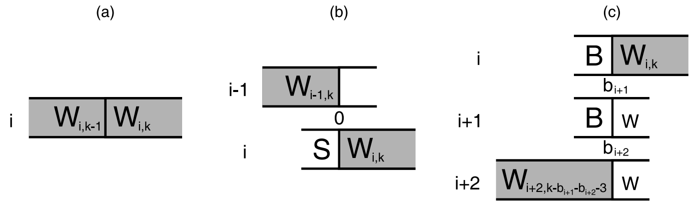

FIG. 5. *Predecessors of $W_{i,k}$ in the critical path, according to the preceding state in station i and corresponding to inequalities (4), (3) and (5).*

Lemma 8 and Proof. Here, we give the result needed in proof of Lemma 2.

LEMMA 8. *The number of jobs in the critical path is smaller than the number of productions plus the number of workstations in the line, m, minus one:*

$$|cp(r_p)| \leq p + m - 1.$$

Proof. The construction of the critical path begins on job $W_{m,p}$. To count the number of jobs in $cp(r_p)$, we relate them to the underlying items. Let us call δ the difference between item indexes of the first and the last job of $cp(r_p)$ plus one. We study $d = |cp(r_p)| - \delta$ which counts the number of items counted twice minus those omitted when the critical path is constructed. Three possibilities exist, corresponding to the inequalities of Lemma 1.

(3) The predecessor is a job on the same item, by the previous station (see Figure 5.b). The same item is thus counted twice.

(4) The predecessor is a job by the same station, on the previous item (see Figure 5.a). No item is thus omitted or counted twice.

(5) The predecessor is on a next station (see Figure 5.c). Depending on the number of stations skipped, the number of omitted items follow from the inequality (5).

The value of d is thus maximal when the critical path jumps upward a lot without jumping downward. We thus get $d \leq m - 1$ and $|cp(r_p)| \leq \delta + m - 1$. As the first and last items of the critical path are part of the production, as each item between, we have $\delta \leq p$ and the lemma is proved. □

Proof of Theorem 3. To get the mean time to produce P, we consider each possible P–part r_P of the possible runs and weight its length by its probability, giving:

$$T_P = \int f(r_P) l(r_P) dr_P,$$

where $f(r_P)$ is the density function of the P–parts. Aiming to use Lemma 2, we have to relate r_P to its discrete correspondent, $\overline{r}_P$ (which also produce P). Let us note $\gamma(\overline{r}_P)$ the set of continuous P–parts (in infinite number) which have the same discrete correspondent $\overline{r}_P$. We can decompose the previous integral:

$$T_P = \sum_{\overline{r}_P} \int_{r_P \in \gamma(\overline{r}_P)} f(r_P) l(r_P) dr_P. \tag{8}$$

By Lemma 2, we get:

$$T_P \leq \sum_{\overline{r}_P} l(\overline{r}_P) \int_{r_P \in \gamma(\overline{r}_P)} f(r_P) dr_P.$$

As the discretization by "grouping at the end" simply concentrates the probability masses in intervals, the integrals in the last equation give the probabilities of the P–parts in discrete time. We get:

$$T_P \leq \sum_{\overline{r}_P} l(\overline{r}_P) P[\overline{r}_P] = \overline{T}_P.$$

The way to the lower bound is very similar. By Lemma 2, (8) becomes:

$$T_P \geq \sum_{\overline{r}_P} \Big(l(\overline{r}_P) - \tau(P + m - 1) \Big) \int_{r_P \in \gamma(\overline{r}_P)} f(r_P) dr_P.$$

For the same reasons as previously and as $\sum_{\overline{r}_P} \int_{r_P \in \gamma(\overline{r}_P)} f(r_P) dr_P = 1$, we get the lower bound. □

Proof of Theorem 4. The lower bound, $\overline{P}_T \leq P_T$, follows from the upper bound in Theorem 3, $T_P \leq \overline{T}_P$. As the mean time to produce a given production is longer in discretized time, the mean quantity produced in a given time T is smaller in discretized time.

Similarly, the upper bound, $P_T \leq \overline{P}^*$, comes from the lower bound in Theorem 3. The time $\overline{T}^*$ to produce $\overline{P}^*$ respects $\overline{T}^* - \tau(\overline{P}^* + m - 1) \leq T^*$, where T^* is the mean time, in continuous time, to produce $\overline{P}^*$. By definition of $\overline{P}^*$, we get $T \leq T^*$. Consequently, the production in T, P_T, is smaller than the one in T^*, $\overline{P}^*$. □

Proof of Corollary 5. Let us divide the equation of Theorem 3 by the production P, in steady–state case (when $P \to \infty$). First, the last term becomes the cycle time computed by the BDPH method. Second, the middle term becomes the exact cycle time. Finally, the first term simplifies to the cycle time in discretized time minus the step length, as the term $(m-1)$ disappears when the time becomes infinite. □

Acknowledgments. The work of J.S. Tancrez has been supported by the European Social Fund and the Walloon Region of Belgium and is made in collaboration with Fontainunion SA. Authors are also grateful to Prof. P. Chevalier for his helpful comments on earlier versions of the work.

REFERENCES

[1] T. Altiok, *Approximate analysis of queues in series with phase-type service times and blocking*, Oper. Res., 37 (1989), pp. 601–610.

[2] R.G. Askin and C.R. Standridge, *Modeling and analysis of manufacturing systems*, John Wiley & Sons, Inc., New York, 1993.

[3] A. Brandwajn and Y.L. Jow, *An approximation method for tandem queues with blocking*, Oper. Res., 36 (1988), pp. 73–83.

[4] J.A. Buzacott and J.G. Shanthikumar, *Stochastic models of manufacturing systems*, Prentice-Hall, Englewood Cliffs, New Jersey, 1993.

[5] R.B. Chase, N.J. Aquilano, and F.R. Jacobs, *Operations management for competitive advantage*, McGraw-Hill/Irwin, New York, 2001.

[6] Y. Dallery and Y. Frein, *On decomposition methods for tandem queueing networks with blocking*, Oper. Res., 41 (1993), pp. 386–399.

[7] Y. Dallery and S.B. Gershwin, *Manufacturing flow line systems: a review of models and analytical results*, Queueing Syst., 12.

[8] C. Dinçer and B. Deler, *On the distribution of throughput of transfer lines*, Journal of the Operational Research Society, 52 (2000), pp. 1170–1178.

[9] S.B. Gershwin, *An efficient decomposition method for the approximate evaluation of tandem queues with finite storage space and blocking*, Oper. Res., 35 (1987), pp. 291–305.

[10] M.K. Govil and M.C. Fu, *Queueing theory in manufacturing : A survey*, Journal of Manufacturing Systems, 18 (1999), pp. 214–240.

[11] L. Gun and A.M. Makowski, *An approximation method for general tandem queueing sytems subject to blocking*, src technical report 87-209-r1, Electrical Engineering Departement and Systems Research Center, University of Maryland, 1987.

[12] T. Hill, *Manufacturing strategy*, Irwin, Burr Ridge, Illinois, 1994.

[13] F.S. Hillier and R.W. Boling, *Finite queues in series with exponential or erlang service times—a numerical approach*, Oper. Res., 15 (1967), pp. 286–303.

[14] L. Kerbache and J. MacGregor Smith, *Multi-objective routing within large scale facilities using open finite queueing networks*, European J. Oper. Res., 121 (2000), pp. 105–123.

[15] H.T. Papadopoulos, *The throughput of multistation production lines with no intermediate buffers*, Oper. Res., 43 (1995), pp. 712–715.

[16] H.T. Papadopoulos and C. Heavey, *Queueing theory in manufacturing systems analysis and design : A classification of models for production and transfer lines*, European J. Oper. Res., 92 (1996), pp. 1–27.

[17] H.G. Perros, *Queueing networks with blocking : exact and approximate solutions*, Oxford University Press, 1994.

[18] W.J. Stewart, *Introduction to the numerical solution of Markov chains*, Princeton University Press, 1994.