

This is the peer reviewed version of the following article:

The two-machine one-buffer continuous time model with restart policy / Gebennini, Elisa; Grassi, Andrea; Fantuzzi, Cesare. - In: ANNALS OF OPERATIONS RESEARCH. - ISSN 0254-5330. - STAMPA. - 231:1(2015), pp. 33-64. [10.1007/s10479-013-1373-9]

*Terms of use:*

The terms and conditions for the reuse of this version of the manuscript are specified in the publishing policy. For all terms of use and more information see the publisher's website.

25/04/2024 10:29

## The two-machine one-buffer continuous time model with restart policy

Elisa Gebennini · Andrea Grassi · Cesare Fantuzzi

Received: date / Accepted: date

**Abstract** This paper deals with the performance evaluation of production lines in which well defined machine start/stop control policies are applied.

A modeling approach has been developed in order to reduce the complexity of a two-machine one-buffer line where a specific control policy, called “restart policy”, is adopted. The restart policy exercises control over the start/stop condition of the first machine: when the buffer gets full and, as a consequence, the first machine is forced to stop production (i.e. it is blocked), the control policy keeps the first machine in an idle state until the buffer becomes empty again. The rationale behind this policy is to reduce the blocking frequency of the first machine, i.e. the probability that a blockage occurs on the first machine due to the buffer filling up. Such a control policy is adopted in practice when outage costs (e.g. waste production) are related to each restart of the machine.

The two-machine one-buffer line with restart policy (*RP* line) is here modeled as a continuous time Markov process so as to consider machines having different capacities and working in an asynchronous manner. The mathematical *RP* model is described along with its analytical solution. Then, the most critical line performance measures are derived and, finally, some numerical examples are reported to show the effects of such a policy on the blocking frequency of the first machine.

---

Elisa Gebennini  
Dipartimento di Scienze e Metodi dell'Ingegneria,  
Università degli Studi di Modena e Reggio Emilia  
Via Amendola 2, 42122 Reggio Emilia, Italy  
Tel.: +39-0522-522672  
E-mail: elisa.gebennini@unimore.it

Andrea Grassi  
Dipartimento di Scienze e Metodi dell'Ingegneria,  
Università degli Studi di Modena e Reggio Emilia  
Via Amendola 2, 42122 Reggio Emilia, Italy  
E-mail: andrea.grassi@unimore.it

Cesare Fantuzzi  
Dipartimento di Scienze e Metodi dell'Ingegneria,  
Università degli Studi di Modena e Reggio Emilia  
Via Amendola 2, 42122 Reggio Emilia, Italy  
E-mail: cesare.fantuzzi@unimore.it

**Keywords** production line · restart policy · continuous time Markov process · performance estimation

## 1 Introduction

This paper presents an innovative Markov model for inhomogeneous and asynchronous two-machine one-buffer production lines where a control policy is applied to improve the line efficiency.

Production lines represent a particular configuration of manufacturing systems in which machines are connected to one another so as to form a series. This configuration is generally adopted by companies when high-volume production is required. Frequently, production lines are also automated so as to achieve higher production rates and repetitiveness of task processes.

The performance of a production line is strictly related to the nominal capacity and the reliability parameters of the machines involved. Specifically, this work considers machines that can operate at different nominal capacities and whose time-to-failure and time-to-repair distributions are exponential. As regards the latter assumption, practical evidence shows that failures and repairs can be often modeled as memory-less processes, especially in case of automated production lines (see, e.g., Perrica et al, 2008).

Moreover, also interactions between machines play an important role in determining the whole line performance. This is due to the fact that the effect of a failure occurring on a specific machine may propagate to other machines located both in the upstream and in the downstream of the line. We refer to a “starvation” when an operational machine is found idle as a consequence of an interruption of its incoming flow since a machine in the upstream of the line has failed; we refer to a “blockage” when its outgoing flow is prevented owing to a failure in the downstream. In order to reduce efficiency losses due to machine interactions, buffers can be positioned along the line that can avoid starvation and blocking events by accumulating and releasing material as needed.

Another strategy that can be employed to improve the line efficiency is to adopt “control policies”, i.e. to exercise a kind of control over the behavior of the involved machines when certain conditions occur. Specifically, this work focuses on a particular control policy typical of processes that incur significant extra costs if interrupted for any reason. Such a control policy is called here “restart policy” and acts as follows: when a machine is blocked because another machine has failed in the downstream and the intermediate buffer is full (so as no decoupling effect takes place), it is not allowed to leave the blocking state as soon as the buffer level starts to decrease. Hence, this restart policy forces a machine once it has become blocked, to remain idle until the downstream buffer becomes empty again. The aim is to reduce the probability of subsequent blocking events that can occur if the machine resumes production when the level of the buffer is just below its maximum size, i.e., when there is little storage space in the downstream.

The motivation to introduce this kind of control policy arises from a real world case study of automated packaging systems. In a typical line configuration package formation and filling is executed by the first machine, called filling machine, with a continuous and aseptic process where the packaging material passes through a heated hydrogen peroxide bath. The critical aspect is the production of a certain amount of waste when the filling machine resumes production after any stoppage. This is because when the filling process is interrupted for any reason (e.g., a failure or a blocking event) a portion of the packaging material remains in contact with the hydrogen peroxide for too long. As a consequence, the first packages produced when the stoppage is removed do not comply with the quality requirements.

Thus, in this case, such as in other applications with significant outage costs, the need arises to reduce to a bare minimum the stoppages of the first machine of the line. On one hand, the reduction

of internal stoppages can be only obtained by improving the failure rate of the very machine. On the other hand, a reduction of those stoppages due to the blocking of the outgoing flow can be obtained by introducing an intermediate buffer and the restart policy described above which forces the first machine to wait for the buffer to become empty before resuming production after any blocking event.

Hence, the production line under analysis results to be a complex system whose overall performance not only depends on machines' capacities and reliability parameters, but also on the buffer allocation along the line and, moreover, on the restart policy applied to the first machine.

Analytical modeling of production lines has been extensively investigated in the literature as discussed in Section 1.1. Nevertheless, to the author's knowledge, none of the previous analytical models is able to describe an inhomogeneous and asynchronous production line with a restart policy.

### 1.1 Literature review

Since there are several factors affecting the whole line performance (i.e. machines reliability parameters, machine capacities and buffer allocation along the line), the design of a production line is a complex task. As a consequence, there is an increasing need to develop efficient methodologies and approaches to quickly compute line efficiency and productivity with respect to all the key factors. Specifically, this work focuses on mathematical methods even if there are other methodologies such as simulation. The great advantage of mathematical modeling is that it can lead to the deepest understanding of the system and provide results in very short periods of time (e.g., the model proposed in this paper can be solved by a computer in few seconds). On the other hand, simulations may be more suitable to study large and complex systems but are often awkward and time-consuming.

For short lines, i.e. consisting of two machines decoupled by one finite buffer (2M-1B line), analytical modeling makes it possible to quickly obtain accurate estimations of the line performance. For extensive reviews, the reader can refer to Dallery and Gershwin (1992) and textbooks as Papadopoulos et al (1993) and Gershwin (2002). Analytical models of production lines can be adopted to evaluate the performance with respect to a predefined line configuration, as well as to be integrated in optimization techniques to derive optimal buffer allocation (Papadopoulos and Vidalis, 1998, 1999, 2001a,b; Bulgak et al, 1995; Gershwin and Schor, 2000; Spinellis and Papadopoulos, 2000; Lutz et al, 1998).

For lines longer than the 2M-1B system, exact analytical solutions are not available. Nevertheless, approximate methodologies have been developed. These methodologies can be classified into two main groups: "decomposition techniques", whose main idea is to decompose the line in a series of 2M-1B sub-systems (important works are Gershwin, 1987; Dallery et al, 1989; Choong and Gershwin, 1989; Dallery and Frein, 1993; Burman, 1995; Yeralan and Tan, 1997; Tan and Yeralan, 1997; Gershwin and Burman, 2000; Levantesi et al, 2003; Maggio et al, 2003; Gershwin and Werner, 2007), and "aggregation techniques", where the 2M-1B sub-system is replaced by one single equivalent machine (see Koster, 1987; Chiang et al, 2000, 2001; Enginarlar and Meerkov, 2005; Chiang et al, 2008). Finally, there exist other approaches, such as the Expansion Method by MacGregor Smith and Daskalaki (1988).

An interesting point is that the main techniques addressing long line performance estimation make use of models developed for the 2M-1B sub-system. This emphasizes the importance of investigating such a simple but not trivial system.

Since production lines are generally characterized by inhomogeneous asynchronous machines, system models must be able to take into consideration machines having different nominal capacities working on an unsynchronized material flow. This is obtained by modeling the system as a continuous time Markov process. This kind of modeling approach was originally proposed by Zimmermann (1956) and ex-

tended by Buzacott (1967a,b). Gershwin and Schick (1980) developed a continuous model for a 2M-1B production line with deterministic processing rates. This line is called here the “Basic line” and the corresponding model the “Basic model”.

Since then, extensions of the Basic model were carried out to represent more complex behaviors of the line. These studies significantly contribute to the literature since the more appropriate the model for the 2M-1B sub-system is, the more representative the approach for the analysis of the whole line will be. As an example, Tolio et al (2002) presented an analytical method where each machine can fail according to different failure modes. Kim and Gershwin (2005) investigated the relationship between productivity and quality and proposed a model with both quality and operation failures. Recent studies have been carried out to generalize continuous time Markov models by including machines with multiple up and down states (see, e.g., Tan and Gershwin, 2009, 2011; Tolio, 2011).

In this paper an interesting extension of the Basic model is discussed. Specifically, the introduction of a “restart policy” is investigated. Such a control policy consists in maintaining the first machine in a “controlled idle state” each time it gets blocked, as a consequence of the buffer filling up, until the buffer empties again. The objective is to reduce a key performance measure called here blocking frequency, i.e., the frequency at which the buffer becomes full.

The first attempt to model this kind of restart policy was carried out by Gebennini et al (2009, in press). Nevertheless, the model developed by the authors is a discrete time Markov model suitable for transfer lines only, that is lines in which machines have the same nominal capacity and the material flow is synchronized. Hence, by introducing this restart policy into the Basic model, the present study extends the work of Gebennini et al (2009, in press) to the case of inhomogeneous asynchronous production lines, so as to allow the consideration of machines having different capacities. This line with restart policy is called here the “*RP* line” and the new corresponding model the “*RP* model”.

Note that the introduction of this control policy significantly increases the complexity of the mathematics involved. This emphasizes the need for methodologies to facilitate mathematical modeling. The literature regarding production lines lacks such a discussion. Therefore, this paper presents a modeling approach for the 2M-1B *RP* line with restart policy based on the partitioning of the state space so as to reduce its complexity. Two partitions of the state space can be identified: first, each of them is mathematically treated as it were the only one representing the system behavior (i.e. in isolation); then, the solution to the original system is determined as a combination of the solutions found for the partitions solved in isolation.

The *RP* line is modeled as a continuous time Markov process and the analytical solution is provided by applying the modeling approach formalized in this paper.

The remainder of the paper is organized as follows. Section 2 describes the 2M-1B *RP* line and introduces the main steps of the modeling approach. Section 3 develops the continuous-time mixed state *RP* model. Section 4 provides key performance measures and proves the conservation of flow, while Section 5 reports the solution technique. Finally, Section 6 shows interesting results from the application of the proposed model and Section 7 points out some concluding remarks.

## 2 The 2M-1B *RP* line

The two-machine one-buffer line with restart policy (2M-1B *RP* line) is assumed to be made up of two Markovian machines decoupled by a finite intermediate buffer. As described in Section 1, the behavior of the system of interest is made complex by the application of a control policy, called restart policy (*RP*), whose aim is to reduce the number of stoppages of the first machine due to blocking events.

A blocking event occurs when the first machine is operational but it is prevented from processing parts because the downstream buffer is full. Specifically, the restart policy acts as follows: when the buffer fills up and the first machine gets blocked, the first machine is forced to remain idle until the buffer becomes empty again. When the first machine is forced to remain idle it is said to be in the so-called “controlled idle state”. This makes it possible to prevent situations where the first machine resumes processing parts when the buffer level is still high. Thus, the probability of the next blocking event occurring in a short time is significantly reduced.

As a consequence of the restart policy, it is possible to identify two different system behaviors:

- The *standard operation* way: both machines can interact, work, fail and be repaired according to their own failure/repair rates and to the buffer level;
- The *buffer drainage* way: the second machine acts as it were isolated while the first machine remains idle, i.e. in the “controlled idle state”. As a consequence, the buffer level can only decrease (if the second machine is up) or stay constant (if it is down).

The 2M-1B *RP* line is supposed to operate according to *standard operation* behavior. The switch to the *buffer drainage* behavior occurs when the buffer fills up; the switch from the *buffer drainage* behavior to the *standard operation* behavior occurs when the buffer becomes empty.

As discussed in Section 1, the restart policy aims to reduce the blocking frequency, i.e. the frequency at which the buffer becomes full. Thus, it is particularly useful when some waste parts or outage costs are produced each time the first machine resumes operation after a stoppage.

## 2.1 Modeling approach

This section introduces to the main steps and definitions of the modeling approach applied to the continuous-time mixed state 2M-1B *RP* model in Section 3.

The hypothesis is to study the system in steady state. The Markov chain model consists of a state space  $\mathcal{S}$  and a transition probability matrix  $\mathbf{P}$  whose entry  $p(S_i, S_j)$  is the probability of a transition from state  $S_i \in \mathcal{S}$  to state  $S_j \in \mathcal{S}$ .

The procedure for obtaining the steady state probability distribution of the 2M-1B *RP* line is based on the partitioning of the state space  $\mathcal{S}$  in order to split the original model into two sub-models. The steps are as follows.

### Step 1: Partitioning the state space

It is assumed that the state space  $\mathcal{S}$  can be partitioned into two non empty subsets  $\mathcal{P}_1, \mathcal{P}_2 \in \mathcal{S}$  so that

$$\mathcal{P}_1 \cup \mathcal{P}_2 = \mathcal{S}, \quad (1)$$

$$\mathcal{P}_1 \cap \mathcal{P}_2 = \emptyset. \quad (2)$$

As described in more detail in Section 3, it is convenient to relate the two partitions  $\mathcal{P}_1$  and  $\mathcal{P}_2$  to the two possible ways the system behaves, i.e. the *standard operation* behavior ( $\mathcal{P}_1$ ) and the *buffer drainage* behavior ( $\mathcal{P}_2$ ).

The transitions linking states of one partition to states of the other partition are called here *switching transitions*. Hence, there is an occurrence probability of the switching transition from  $\mathcal{P}_1$  to  $\mathcal{P}_2$ ,

denoted as  $P_s^{12}$ , and an occurrence probability of the switching transition from  $\mathcal{P}_2$  to  $\mathcal{P}_1$ , denoted as  $P_s^{21}$ .

Specifically, in partition  $\mathcal{P}_i$ , with  $i = 1, 2$ , it is possible to identify a set of *internal states*. The set of *internal states*  $\mathbb{I}^i$  of partition  $\mathcal{P}_i$  is defined as the collection of states that are connected for any feasible transition to another state of the same partition, or, more precisely:

$$\mathbb{I}^i = \{S_l \in \mathcal{P}_i \mid \text{for all transition probability } p(S_l, S_m) \neq 0, S_m \in \mathcal{P}_i\}. \quad (3)$$

Transitions between the two partitions, i.e. *switching transitions*, involve the definition of *boundary states*. A boundary state is a passing gate from one partition to the other. We can define two types of boundary states, the *exit states*, i.e. the set of state for which there exists an outbound transition to states outside the partition, and the *arrival states*, i.e. the set of states for which there exist an inbound transition from states outside the partition.

In a formal manner, we define the set of *exit states* of partition  $\mathcal{P}_i$  towards partition  $\mathcal{P}_j$ , with  $i, j = 1, 2$  and  $i \neq j$ , as

$$\mathbb{E}^{ij} = \{E^{ij}\} = \{S_l \in \mathcal{P}_i \mid \exists S_k \in \mathcal{P}_j, i \neq j : p(S_l, S_k) \neq 0\}. \quad (4)$$

The set of *arrival states* of partition  $\mathcal{P}_i$  from partition  $\mathcal{P}_j$ , with  $i, j = 1, 2$  and  $i \neq j$ , is defined as

$$\mathbb{A}^{ij} = \{A^{ij}\} = \{S_l \in \mathcal{P}_i \mid \exists S_k \in \mathcal{P}_j, i \neq j : p(S_k, S_l) \neq 0\}. \quad (5)$$

Figure 1 shows the partitions  $\mathcal{P}_1$  and  $\mathcal{P}_2$  and the *switching transitions* between the exit states of one partition to the arrival states of the other partition.

#### Step 2: Partitions in isolation

The aim is to treat each partition independently, i.e. in “isolation”. In such a way the mathematical treatment of the problem is extremely simplified. The original Markov chain model, described by the whole state space  $\mathcal{S}$  is split into two Markov chain sub-models, each of them described by a single partition  $\mathcal{P}_i \subset \mathcal{S}$ ,  $i = 1, 2$ .

The main assumption for the application of this step is that the system is studied in steady state. Under this condition the following proposition holds.

*Proposition 1:* For the system to be in steady state, the occurrence probability  $P_s^{ij}$  of the switching transition from partition  $\mathcal{P}_i$  to partition  $\mathcal{P}_j$  equals the occurrence probability  $P_s^{ji}$  of the switching transition from partition  $\mathcal{P}_j$  to partition  $\mathcal{P}_i$ , for  $i, j = 1, 2$  and  $i \neq j$ .

Thus in steady state we have,

$$P_s^{ij} = P_s^{ji}, \quad i, j = 1, 2, i \neq j. \quad (6)$$

Let us consider partition  $\mathcal{P}_i$  with  $i = 1, 2$ . Given equation (6), it is possible to replace the switching transition to  $\mathcal{P}_j$  with  $j \neq i$  by a transition from its own exit states  $\mathbb{E}^{ij}$  to its own arrival  $\mathbb{A}^{ij}$  state, given that the new occurrence probability equals the probability  $P_s^{ij}$  of the original switching transition. Thus, as shown in Figure 1 partition  $\mathcal{P}_i$  with  $i = 1, 2$  results to be “isolated” (or “in isolation”) in the sense that each state of the partition is connected only to other states belonging to the same partition. The two partitions in isolation represent two renewal processes so that the corresponding Markov chain models can be solved independently. The solution to the Markov chain sub-model described by partition  $\mathcal{P}_i$  in isolation is the steady state probability distribution denoted here as  $\tilde{f}^i$ , with  $i = 1, 2$ .

[PUT FIGURE 1 HERE]

**Step 3: Partition probabilities**

Considering the original system behavior, at each renewal of partition  $\mathcal{P}_i$  in isolation, partition  $\mathcal{P}_j$  takes place, with  $i, j = 1, 2$  and  $i \neq j$ . Thus, in order to rebuild the original Markov model describing the 2M-1B *RP* line, it is necessary to ensure that the system is in exactly one of the two partitions at a time. As a consequence, it is necessary to introduce the term  $\pi^i$ , called here as the “partition probability” of  $\mathcal{P}_i$ , representing the probability for the system being into partition  $\mathcal{P}_i$  with  $i = 1, 2$ . The partition probabilities satisfy the following normalization equation:

$$\sum_{i=1}^2 \pi^i = 1. \quad (7)$$

**Step 4: System solution**

Once both the isolated solutions and the partition probabilities have been found, it is possible to express the system solution  $f$ , i.e. the solution to the original Markov chain model, as a combination of the solutions  $\tilde{f}^i$ , with  $i = 1, 2$ , found for each partition in isolation:

$$f = \sum_{i=1}^2 \pi^i \tilde{f}^i. \quad (8)$$

**3 The continuous-time mixed-state *RP* model**

In this section the modeling approach described in Section 2.1 is applied to the continuous-time mixed-state model of the 2M-1B *RP* line.

Recall that if no control policy is in place, the continuous 2M-1B *RP* line becomes a simple 2M-1B line investigated by the previous literature. In particular, we refer to a previous continuous-time mixed-state model for a 2M-1B line without restart policy that is called here Basic model (see Gershwin and Schick, 1980; Gershwin, 2002).

The *RP* model significantly extends the Basic model. As explained in Section 2, the restart policy introduces two different behaviors in the system, i.e. the *standard operation*, where no control policy is applied, and the *buffer drainage*, where the first machine is put in the controlled idle state allowing the buffer level to decrease.

**3.1 Notation and assumptions**

One of the main advantages of modeling the system by a continuous-time mixed-state model is that inhomogeneous asynchronous machines can be properly addressed.

The main assumptions of the 2M-1B *RP* model are as follows:

1. A machine is said to be “starved” if its incoming flow is interrupt (the buffer is empty and the upstream machine is down). A machine is said to be “blocked” if its outgoing flow is interrupt (the buffer is full and the downstream machine is down). For the 2M-1B *RP* line it is assumed that the first machine is never starved and the second machine is never blocked;
2. The material that is processed is treated as though it is a continuous fluid;
3. The machine are either up (i.e. operational) or down (i.e. under repair) so that there is no middle ground in the model;



4. The machines may have different production rates so that the line can be inhomogeneous and asynchronous;
5. The machines have exponentially distributed times between failures and time to repair;
6. The failures are assumed to be operation-dependent so as the first machine cannot fail while it is blocked or in the controlled idle state, and the second one cannot fail while it is starved;
7. The intermediate buffer has a finite capacity (note that, as a consequence of assumption 2 the buffer level is continuous in the range defined by zero and the maximum capacity);
8. The material in process is not destroyed or rejected at any stage in the system;
9. When the restart policy takes place the system switches from the *standard operation* behavior to the *buffer drainage* behavior. This happens when the buffer fills up. The system returns to the *standard operation* behavior when the buffer becomes empty again.

The following notation is adopted:

- $x \in \mathbb{R}$  is the buffer level, with  $0 \leq x \leq N$ , being  $N$  the buffer capacity (see assumption 7);
- $\alpha_i = 0, 1$  is the condition of machine  $i$ , with  $i = 1, 2$  (see assumption 3): if  $\alpha_i = 0$  machine  $i$  is down, if  $\alpha_i = 1$  machine  $i$  is operational;
- $\mu_i$  is the production rate of machine  $i$ , with  $i = 1, 2$  (see assumption 4);
- $p_i$  is the failure rate of machine  $i$ , with  $i = 1, 2$  (see assumption 5);
- $r_i$  is the repair rate of machine  $i$ , with  $i = 1, 2$  (see assumption 5).

Note that if  $\mu_1 < \mu_2$  and the buffer is empty the second machine is forced to slow down its speed to  $\mu_1$ . Thus, if  $x = 0$  the probability of failure of the second machine at time  $t + \delta t$ , provided that  $\alpha_2(t) = 1$ , is  $p_2^b \delta t$ , where

$$p_2^b = \frac{\mu_1}{\mu_2} p_2. \quad (9)$$

This is because failures are assumed to be operation dependent (see assumption 6). When the buffer is not empty, such a probability is  $p_2 \delta t$ .

Consider now the case where  $\mu_1 > \mu_2$  and, specifically, the state with the buffer level at the maximum capacity  $N$  and both machines operational. In the Basic model without any control policy the system can persist in this state, with the first machine reducing its speed to  $\mu_2$ . On the contrary, if the restart policy is implemented we assume that as soon as the buffer level equals  $N$  with the second machine in an operational state ( $\alpha_2 = 1$ ), either because it is repaired or because it does not fail (note that the first machine must be operational to allow the buffer to fill up), the system switches instantaneously into the *buffer drainage* behavior and the first machine is put in the controlled idle state.

The probability to have a repair at time  $t + \delta t$  of a machine  $i$  failed at  $t$  ( $\alpha_i(t) = 0$ ) is  $r_i \delta t$ .

### 3.2 Step 1: Partitioning the state space

It is convenient to partition the state space according to the two different ways the system behaves so that we have:

- the *standard operation* partition;
- the *buffer drainage* partition.

Thus, the system state can be defined as  $\mathcal{S} = (\beta, x, \alpha_1, \alpha_2)$  where, as described in Section 3.1,  $x \in \mathbb{R}$  is the buffer level with  $0 \leq x \leq N$ ;  $\alpha_i = 0, 1$  is the condition of the machine  $i = 1, 2$ ; and  $\beta$  is a binary parameter that is conveniently introduced here to distinguish between states belonging to the *standard*

*operation* partition ( $\beta = 0$ ) or the *buffer drainage* partition ( $\beta = 1$ ). It is assumed that when  $\beta = 1$ , i.e. the first machine is in the controlled idle state,  $\alpha_1$  is fixed and set to 1 (it is forced to remain idle, but it is operational and cannot fail) so that states  $(1, x, 0, \alpha_2)$  are not feasible.

Note that the system state includes three binary parameters and a continuous component  $x$ . Thus, the probability distribution has a density function on  $(0, N)$ , denoted as  $f(\beta, x, \alpha_1, \alpha_2)$ . The aim of this work is to express the solution  $f(\beta, x, \alpha_1, \alpha_2)$  in order to be able to compute some performance measures.

The switching transition from the *standard operation* partition to the *buffer drainage* partition occurs if the buffer fills up.

Let us consider the simplest case where  $\mu_1 \leq \mu_2$ . The buffer might fill up only if the first machine is up and the second is down. In fact, since the first machine is slower than the second one, if both the machines are operational the buffer level tends to decrease and the maximum buffer size  $N$  cannot be reached. In other words, the only non-transient state with the buffer level  $x = N$  is that with  $\alpha_1 = 1$  and  $\alpha_2 = 0$ . For convenience, we assume that this state belongs to the *standard operation* partition so that it is denoted as  $(0, N, 1, 0)$  where the first term  $\beta$  is set to zero.

Note that state  $(0, N, 1, 0)$  is peculiar for the following considerations:

- in this state the first machine is blocked, i.e. it is operational but it cannot work since its outgoing flow is prevented being the second machine down and no storage space in the buffer;
- since the maximum buffer size  $N$  is defined by a physical limit, this state acts as a *mass point* which itself has nonzero probability (see Gershwin, 2002). Let  $\mathbf{p}(0, N, 1, 0)$  be the probability for the system being in state  $(0, N, 1, 0)$ .

The behavior of the 2M-1M *RP* line under analysis is such that when the second machine is repaired (and, consequently, the buffer level starts to decrease) the first machine is not allowed to resume production but it is forced to remain idle, i.e. put in the “controlled idle state”. Thus, as soon as the second machine is repaired the switching transition from the *standard operation* partition to the *buffer drainage* partition occurs. This means that state  $(0, N, 1, 0)$  is the exit state of the *standard operation* partition. The arrival states of the *buffer drainage* partition are states where the buffer level is reduced by at most the amount processed by the second machine in  $\delta t$  and both machines up since the first machine cannot fail (it is prevented from working) and the second machine has been repaired. Thus, the arrival state of the *buffer drainage* partition is state  $(1, x, 1, 1)$  with  $\leq N - \mu_2 \delta t \leq x < N$  (note that  $\beta$  is set now to 1), where  $\mu_2 \delta t$  is what the second machine processes in  $\delta t$ .

Since the switching transition occurs if the system is in state  $(0, N, 1, 0)$  and the second machine is repaired, we have:

$$P_s = r_2 \delta t \mathbf{p}(0, N, 1, 0). \quad (10)$$

Once the system has entered the *buffer drainage* partition, it remains in this partition until the buffer becomes empty. Note that the first machine is in the “controlled idle state” so that it cannot work nor fail and the buffer level can only decrease (if the second machine is up) or stay constant (if the second machine is down).

Hence, the switching transition from the *buffer drainage* partition to the *standard operation* partition occurs when  $x$  reaches the physical limit at zero since the second machine is up ( $\alpha_2 = 1$ ). Specifically, we assume that the switching transition occurs between state  $(1, x, 1, 1)$  with  $0 < x \leq \mu_2 \delta t$  of the *buffer drainage* partition and state  $(0, 0, 1, 1)$  of the *standard operation* partition where state  $(0, 0, 1, 1)$  is a mass point, similarly to  $(0, N, 1, 0)$ .

Since this switching transition occurs if the system is in state  $(1, x, 1, 1)$  with  $0 < x \leq \mu_2 \delta t$  and the second machine does not fail, we have:

$$P_s = (1 - p_2 \delta t) \int_0^{\mu_2 \delta t} f(1, x, 1, 1) dx, \quad (11)$$

or, ignoring second order terms,

$$P_s = \mu_2 \delta t f(1, 0, 1, 1), \quad (12)$$

Since the system is studied in steady state, Proposition 1 holds so that we have:

$$r_2 \mathbf{p}(0, N, 1, 0) = \mu_2 f(1, 0, 1, 1), \quad (13)$$

The above considerations have been done for the case with  $\mu_1 \leq \mu_2$ , the exit and arrival states for the case with  $\mu_1 > \mu_2$  are reported in Table 1.

[PUT TABLE 1 HERE]

Note that when the system is in internal states of the *standard operation* partition (i.e. with intermediate buffer levels) it can be seen as a simple 2M-1B line where machines work, fail and are repaired according to their own reliability parameters. This occurs in a previous model presented by the literature that is referred here as the Basic model (see Gershwin and Schick, 1980; Gershwin, 2002). Thus, we recall the Basic model for what concerns state of the *standard operation* partition with intermediate buffer level, but we extend it significantly by considering the switching transitions to/from the *buffer drainage* partition introduced by the restart policy.

### 3.3 Step 2: Partitions in isolation

In this section the two partitions are treated separately, i.e. in isolation.

By applying Step 2 of the modeling approach, we isolate each partition by using a direct transition from its own exit states and its own arrival states. Once a partition has been “isolated” it can be solved by taking it as “standing alone”.

For convenience, we use a simplified notation for the partitions in isolation. Specifically, the system state for the *standard operation* partition in isolation is defined as  $\mathcal{S}^S = (x, \alpha_1, \alpha_2)$ , for the *buffer drainage* partition in isolation as  $\mathcal{S}^D = (x, \alpha_2)$ , being  $x$  the buffer level and  $\alpha_i = 0, 1$  the condition of machine  $i = 1, 2$ .  $\mathcal{S}^S = (x, \alpha_1, \alpha_2)$  corresponds to state  $(0, x, \alpha_1, \alpha_2)$  of the original Markov process,  $\mathcal{S}^D = (x, \alpha_2)$  corresponds to state  $(1, x, 1, \alpha_2)$  of the original Markov process. Note that states in the *buffer drainage* partition in isolation do not depend on  $\alpha_1$  since the first machine is in the controlled idle state (it cannot work nor fail) and the second machine operates as it were isolated.

Let  $f^S(x, \alpha_1, \alpha_2, t)$  and  $\mathbf{p}^S(x, \alpha_1, \alpha_2, t)$  be the probability density function and the probability of being in state  $(x, \alpha_1, \alpha_2, t)$  belonging to the *standard operation* partition,  $f^D(x, \alpha_2, t)$  be the probability density function for the *buffer drainage* partition.

### 3.3.1 Standard operation partition in isolation

This partition can be solved in isolation by considering it as the only one representing the system behavior. In words, transitions to/from the *buffer drainage* partition (characterizing the original complex behavior) are replaced with direct transitions from the *exit* to the *arrival* states of the *standard operation* partition itself. Thus, the *standard operation* partition in isolation can be thought of as modeling a system where each time the buffer fills up, it empties instantaneously as the second machine gets repaired (or does not fail, in case  $\mu_1 > \mu_2$ ).

Specifically, for  $\mu_1 \leq \mu_2$  we have that

if the system is in state  $(N, 1, 0)$ , it can enter only state  $(0, 1, 1)$ , when the second machine is repaired;

for  $\mu_1 > \mu_2$  we have that

if the system is state  $(N, 1, 0)$  or if both machines are operational and the buffer level is approaching the maximum capacity (i.e., the system is in states  $(x, 1, 1)$  with  $N - (\mu_1 - \mu_2)\delta t \leq x < N$ ) it will pass to state  $(x, 1, 1)$  with  $0 < x \leq (\mu_1 - \mu_2)\delta t$  if the second machine is repaired or no failures occur, respectively.

As introduced above, the *standard operation* partition in isolation can be modeled by means of the Basic model without restart policy (see Gershwin and Schick, 1980; Gershwin, 2002) except for the boundary equations that represent how the system leaves/enters the *exit/arrival* states.

For the sake of clarity, the most significant equations modeling this partition are discussed in the sequel by treating the three cases  $\mu_1 < \mu_2$ ,  $\mu_1 = \mu_2$  and  $\mu_1 > \mu_2$  separately.

All equations for this partition are listed in Appendix A.

Finally, the following normalization equation expressing that the sum of all probabilities must equal 1 is needed to solve the partition in isolation:

$$\sum_{\alpha_1=0}^1 \sum_{\alpha_2=0}^1 \left[ \int_0^N f^S(x, \alpha_1, \alpha_2) dx + \mathbf{p}^S(0, \alpha_1, \alpha_2) + \mathbf{p}^S(N, \alpha_1, \alpha_2) \right] = 1. \quad (14)$$

The technique used to determine the solution to the *standard operation* partition in isolation is explained in detail in Section 5.

#### Case $\mu_1 \leq \mu_2$

As regards the lower boundary ( $x = 0$ ), it is necessary to describe how the system arrives at the arrival state  $(0, 1, 1)$ .

Since  $\mu_1 < \mu_2$ , it is possible to get to  $(0, 1, 1)$  from

- $(0, 1, 1)$  if no failures occur;
- $(0, 0, 1)$  if the first machine is repaired;
- $(x, 1, 1)$ , with  $0 < x \leq (\mu_2 - \mu_1)\delta t$  (for the case  $\mu_1 < \mu_2$ ), if no failures occur;
- $(N, 1, 0)$  if the second machine is repaired;

where the latter is the exit state introduced by the restart policy.

Symbolically, ignoring the second order terms,

$$\begin{aligned} \mathbf{p}^S(0, 1, 1, t + \delta t) = & (1 - (p_1 + p_2^b)\delta t)\mathbf{p}^S(0, 1, 1, t) + r_1\delta t\mathbf{p}^S(0, 0, 1, t) + \\ & + \int_0^{(\mu_2 - \mu_1)\delta t} f^S(x, 0, 1, 1, t)dx + r_2\delta t\mathbf{p}^S(N, 1, 0, t). \end{aligned}$$

Letting  $\delta t \rightarrow 0$ , the equation becomes

$$\begin{aligned} \frac{d}{dt}\mathbf{p}^S(0, 1, 1) = & -(p_1 + p_2^b)\mathbf{p}^S(0, 1, 1) + r_1\mathbf{p}^S(0, 0, 1) + \\ & + (\mu_2 - \mu_1)f^S(0, 1, 1) + r_2\mathbf{p}^S(N, 1, 0). \end{aligned} \quad (15)$$

As regards the upper boundary ( $x = N$ ), the introduction of the restart policy prevents the system from reaching any internal states  $(x, \alpha_1, \alpha_2)$  from the upper boundary.

Thus, states  $(x, 0, 1)$ , with  $N - \mu_2\delta t \leq x < N$ , cannot be reached from the boundary (because of the restart policy) or from any intermediate-buffer-level state (they cannot be reached from states  $(x', \alpha_1, \alpha_2)$  in  $\delta t$ , if  $x' \leq x$  and  $\delta t$  is small, because when the second machine is working the buffer level decreases).

Symbolically, if the second order terms are ignored, we obtain

$$\int_{N - \mu_2\delta t}^N f^S(x, 0, 1, t + \delta t)dx = 0,$$

or,

$$f^S(N, 0, 1) = 0, \quad (16)$$

If  $\mu_1 < \mu_2$ , we notice that also states  $(x, 1, 1)$ , with  $N - (\mu_2 - \mu_1)\delta t \leq x < N$ , cannot be reached from the boundary since the restart policy forces the system to enter the arrival state  $(0, 1, 1)$ . Thus,

$$f^S(N, 1, 1) = 0. \quad (17)$$

Similarly, if  $\mu_1 = \mu_2$  state  $(N, 1, 1)$  can be reached only from itself in  $\delta t$ , if no failures occur. As a consequence,

$$\mathbf{p}^S(N, 1, 1) = 0. \quad (18)$$

*Case  $\mu_1 > \mu_2$*

It is convenient to start by discussing the upper boundary. Recall that, the *standard operation* partition in isolation with  $\mu_1 > \mu_2$  can be seen as representative of a fictitious system where as soon as the buffer gets full with the second machine operational, it falls down to zero (note that reaching states with  $x = N$  is possible only if the first machine is operational). This means that state  $(N, 1, 1)$  is transient. Thus, the system passes directly from states  $(x, 1, 1)$  with  $N - (\mu_1 - \mu_2)\delta t \leq x < N$  to states  $(x, 1, 1)$  with  $0 < x \leq (\mu_1 - \mu_2)\delta t$ . Note state  $(0, 1, 1)$  is also transient since the system cannot persist in that state if  $\mu_1 > \mu_2$ .

Therefore,

$$\mathbf{p}^S(N, 1, 1) = 0. \quad (19)$$

Another effect of the restart policy is that also states  $(x, 0, 1)$  with  $N - \mu_2 \delta t \leq x < N$  can be reached from the boundary. Thus, to the first order,

$$f^S(N, 0, 1) = 0. \quad (20)$$

As regards the lower boundary ( $x = 0$ ), to arrive at state  $(x, 1, 1)$  with  $0 < x \leq (\mu_1 - \mu_2) \delta t$  at time  $t + \delta t$ , the system may have been in one of three sets of states at time  $t$ . It could have been in state  $(0, 0, 1)$  with a repair of the first machine. It could have been in state  $(N, 1, 0)$  with a repair of the second machine. It could have been in any state  $(x, 1, 1)$  with  $N - (\mu_1 - \mu_2) \delta t \leq x < N$  if no failures occur. The latter two transitions are not feasible in the Basic model since they are related to the restart policy introduced in this work. Note that, since  $\mu_1 > \mu_2$ , it is not possible to reach state  $(x, 1, 1)$  with  $0 < x \leq (\mu_1 - \mu_2) \delta t$  from any intermediate-buffer-level state  $(x', \alpha_1, \alpha_2)$  with  $x' \geq x$ .

Symbolically, ignoring the second order terms,

$$\begin{aligned} \int_0^{(\mu_1 - \mu_2) \delta t} f^S(x, 1, 1, t + \delta t) dx = & r_1 \int_0^{t + \delta t} \mathbf{p}^S(0, 0, 1, s) ds + r_2 \int_0^{t + \delta t} \mathbf{p}^S(N, 1, 0, s) ds + \\ & + \int_{N - (\mu_1 - \mu_2) \delta t}^N f^S(x, 1, 1, t + \delta t) dx. \end{aligned}$$

Letting  $\delta t \rightarrow 0$ , the equation becomes

$$(\mu_1 - \mu_2) f^S(0, 1, 1) = r_1 \mathbf{p}^S(0, 0, 1) + r_2 \mathbf{p}^S(N, 1, 0) + (\mu_1 - \mu_2) f^S(N, 1, 1). \quad (21)$$

### 3.3.2 Buffer drainage partition in isolation

In the *buffer drainage* partition the first machine is operational but in the controlled idle state, thus it does not process material and, as a consequence, failures cannot occur. Hence, the system state for this partition does not depend on the state of the first machine and can be represented simply as  $(x, \alpha_2)$ , with  $\alpha_2 = 0, 1$ .

Moreover, states belonging to the *buffer drainage* partition are characterized by a buffer level  $x$  with  $0 < x < N$  and, consequently, only states with intermediate buffer levels are involved.

This leads to the following equations for the *buffer drainage* partition in isolation:

$$\frac{\partial f^D}{\partial t}(x, 1, 1) = -p_2 f^D(x, 1, 1) + r_2 f^D(x, 1, 0) + \mu_2 \frac{\partial f^D}{\partial x}(x, 1, 1), \quad (22)$$

$$\frac{\partial f^D}{\partial t}(x, 1, 0) = -r_2 f^D(x, 1, 0) + p_2 f^D(x, 1, 1), \quad (23)$$

where the  $t$  argument is suppressed and  $f^D(x, \alpha_2)$  represents the probability density function of state  $(x, \alpha_2)$

Since the steady state versions of equations (22) and (23) have to be simultaneously satisfied, it leads to the following:

$$\frac{\partial f^D}{\partial x}(x, 1) = 0. \quad (24)$$

Therefore,  $f^D(x, \alpha_2)$  is constant.

The *buffer drainage* partition in isolation can be thought of as describing the reduction of the buffer level according to the production, failure and repair rates of the second machine only. Thus, the probability density function describing the system behavior in this case depends only on  $\alpha_2$  and it can be indicated simply as follows:

$$f^D(x, \alpha_2) = f^{*D}(\alpha_2). \quad (25)$$

Equations related to this partition are listed in Appendix A.

Similarly as in the previous case, the isolation procedure requires the following normalization equation in order to find the partition solution:

$$\sum_{\alpha_2=0}^1 \int_0^N f^{*D}(\alpha_2) dx = N \sum_{\alpha_2=0}^1 f^{*D}(\alpha_2) = 1. \quad (26)$$

This makes it possible to obtain the solution to the *buffer drainage* partition in isolation, as detailed in Section 5.

### 3.4 Step 3: Partition probabilities

Once the solutions to both the isolated partitions have been found, it is necessary to consider that the system can be in exactly one of the two partitions at a time, i.e. we have to compute the probability of being either in the *standard operation* partition or in the *buffer drainage* partition.

Let  $\mathbf{T}$  be the mean time between the occurrences of the same switching transition in steady state. The following equation holds:

$$\mathbf{T} = \mathbf{T}^S + \mathbf{T}^D, \quad (27)$$

where  $\mathbf{T}^S$  is the mean time spent in states of the *standard operation* partition during  $\mathbf{T}$ , and  $\mathbf{T}^D$  is the mean time spent in states of the *buffer drainage* partition during  $\mathbf{T}$ .

Thus, the probability of being in each partition can be expressed in terms of  $\mathbf{T}$ ,  $\mathbf{T}^S$  and  $\mathbf{T}^D$  as follows:

$$\pi^S = \frac{\mathbf{T}^S}{\mathbf{T}}, \quad (28)$$

$$\pi^D = \frac{\mathbf{T}^D}{\mathbf{T}}. \quad (29)$$

Since  $\mathbf{T}^S$  can be seen as the mean time between two switching transitions from the *standard operation* partition to the *buffer drainage* partition, we have:

$$\mathbf{T}^S = \frac{1}{\phi^{S,D}}, \quad (30)$$

where  $\phi^{S,D}$  is the frequency of the switch from the *standard operation* partition to the *buffer drainage* partition, being the system in states of the *standard operation* partition.

In other words, given that the system is into the *standard operation* partition,  $\phi^{S,D}$  is the probability of entering (and not persisting) state  $(N, 1, 0)$  or, if  $\mu_1 > \mu_2$ , the probability of being in states  $(x, 1, 1)$ , with  $N - (\mu_1 - \mu_2)\delta t \leq x < N$ , and no failures occur. Thus,

$$\phi^{S,D} = \begin{cases} \mu_1 f^S(N, 1, 0), & \text{if } \mu_1 \leq \mu_2, \\ \mu_1 f^S(N, 1, 0) + (\mu_1 - \mu_2) f^S(N, 1, 1), & \text{if } \mu_1 > \mu_2. \end{cases} \quad (31)$$

By similar reasoning,  $\mathbf{T}^D$  is defined as the mean time between two switching transitions from the *buffer drainage* partition to the *standard operation* partition. Thus,

$$\mathbf{T}^D = \frac{1}{\phi^{D,S}}, \quad (32)$$

where  $\phi^{D,S}$  is the frequency of the switch from the *buffer drainage* partition to the *standard operation* partition, being the system in states of the *buffer drainage* partition. Recall that in the *buffer drainage* partition we consider internal states only and the probability of entering any of them is constant, independent on the buffer level  $x$ . So,

$$\phi^{D,S} = \mu_2 f^{*D}(1). \quad (33)$$

Finally, the expressions of  $\pi^S$  and  $\pi^D$  become:

$$\pi^S = \frac{\phi^{D,S}}{\phi^{D,S} + \phi^{S,D}} = \begin{cases} \frac{\mu_2 f^{*D}(1)}{\mu_2 f^{*D}(1) + \mu_1 f^S(N, 1, 0)}, & \text{if } \mu_1 \leq \mu_2, \\ \frac{\mu_2 f^{*D}(1)}{\mu_2 f^{*D}(1) + \mu_1 f^S(N, 1, 0) + (\mu_1 - \mu_2) f^S(N, 1, 1)}, & \text{if } \mu_1 > \mu_2 \end{cases}, \quad (34)$$

$$\pi^D = \frac{\phi^{S,D}}{\phi^{D,S} + \phi^{S,D}} = \begin{cases} \frac{\mu_1 f^S(N, 1, 0)}{\mu_2 f^{*D}(1) + \mu_1 f^S(N, 1, 0)}, & \text{if } \mu_1 \leq \mu_2, \\ \frac{\mu_1 f^S(N, 1, 0) + (\mu_1 - \mu_2) f^S(N, 1, 1)}{\mu_2 f^{*D}(1) + \mu_1 f^S(N, 1, 0) + (\mu_1 - \mu_2) f^S(N, 1, 1)}, & \text{if } \mu_1 > \mu_2 \end{cases}. \quad (35)$$

### 3.5 Step 4: System solution

In the case of interest, the system behavior is represented by either the *standard operation* partition or the *buffer drainage* partition according to the buffer level. Specifically, the *standard operation* partition works until the buffer gets full and a blocking event occurs. The *buffer drainage* partition represent the system behavior if a blocking event has occurred, until the buffer becomes empty again.

As explained in Step 4 of the modeling approach described in Section 2.1, the solution to the original system is a combination of the solutions found for the two partitions in isolation.

Given that the system state is defined as in Section 3.2, we have:

$$f(\beta, x, \alpha_1, \alpha_2) = (1 - \beta) \pi^S f^S(x, \alpha_1, \alpha_2) + \beta \pi^D f^{*D}(\alpha_2), \quad (36)$$

where  $\pi^S$  is the partition probability of being in the *standard operation* partition and  $\pi^D = 1 - \pi^S$  is the partition probability of being in the *buffer drainage* partition as defined in Section 3.4.

Recall that the first machine cannot fail during the *buffer drainage* partition since it is in the controlled idle state. Thus,

$$f(1, x, 0, \alpha_2) = 0. \quad (37)$$

In order to complete the solution, boundary probabilities must be considered. Since states with  $x = 0$  or  $x = N$  belong to the *standard operation* partition only, we have:



$$\mathbf{p}(\beta, 0, \alpha_1, \alpha_2) = \begin{cases} \pi^S \mathbf{p}^S(0, \alpha_1, \alpha_2) & \text{if } \beta = 0 \\ 0 & \text{if } \beta = 1, \end{cases} \quad (38)$$

and

$$\mathbf{p}(\beta, N, \alpha_1, \alpha_2) = \begin{cases} \pi^S \mathbf{p}^S(N, \alpha_1, \alpha_2) & \text{if } \beta = 0 \\ 0 & \text{if } \beta = 1. \end{cases} \quad (39)$$

The solution to the 2M-1B *RP* model described in this paper can be obtained in closed form as reported in Section 5.

#### 4 Blocking frequency, production rate and conservation of flow

Since the aim of the restart policy is to reduce the stoppages of the first machine due to blocking events, the blocking frequency  $f^b$  is a fundamental performance measure.

Recall that the first machine gets blocked when the system reaches state  $(0, N, 1, 0)$ , i.e. when the buffer is full, the second machine is down and the first machine, even if still operational, cannot release material. Therefore, the blocking frequency  $f^b$  can be determined as the probability of entering (or, equally, of exiting) that state. So,

$$f^b = r_2 \mathbf{p}(0, N, 1, 0), \quad (40)$$

where, for the sake of simplicity, the blocking frequency is expressed as the probability of exiting state  $(0, N, 1, 0)$ .

Another important performance measure is the line production rate. The production rate of each machine  $i$  (with  $i = 1, 2$ ), i.e. the rate at which material leaves the machine, is equal to its capacity multiplied by its efficiency  $E_i$ . Specifically, the speed at which machine  $i$  can operate is  $\mu_i$  if machine  $i$  is not limited by the other one (e.g., if  $\mu_1 < \mu_2$ , when the buffer is empty and the first machine is operational, the second machine cannot be faster than the first one).

Consequently, considering non-zero probabilities only, we have

$$P_1 = \mu_1 \left[ \int_0^N (f(0, x, 1, 0) + f(0, x, 1, 1)) dx + \mathbf{p}(0, 0, 1, 1) \right], \quad (41)$$

for the first machine, and

$$P_2 = \mu_2 \left[ \int_0^N (f(0, x, 0, 1) + f(0, x, 1, 1)) dx + \int_0^N f(1, x, 1, 1) dx \right] + \mu_1 \mathbf{p}(0, 0, 1, 1), \quad (42)$$

for the second one. Note that if  $\mu_1 > \mu_2$  the term  $\mathbf{p}(0, 0, 1, 1)$  is equal to zero.

Since the first machine is in the controlled idle state during the *buffer drainage* partition, states belonging to this partition influences the production rate of the second machine only.

For the system to be in the steady state, the following equation (conservation of flow equation) must be verified:

$$P_1 = P_2. \quad (43)$$

The proof is reported in Appendix B.

## 5 Solution technique

In this section the solution technique adopted to solve the two partitions in isolation is explained in detailed. Given the isolated solutions  $f^S(x, \alpha_1, \alpha_2)$  and  $f^D(x, \alpha_2)$ , the partition probabilities can be easily determined according to equations (34) and (35) and, finally, the system solution according to equation (36).

### 5.1 Standard operation partition in isolation

It is natural to assume the following exponential form for the solution to the steady state density equations of the *standard operation* partition in isolation:

$$f^S(x, \alpha_1, \alpha_2) = C^S e^{\lambda x} Y_1^{\alpha_1} Y_2^{\alpha_2}. \quad (44)$$

By substituting (44) in the internal equations belonging to this partition (equations (92)-(95) in Appendix A) we obtain the three following parametric equations:

$$\sum_{i=1}^2 (p_i Y_i - r_i) = 0, \quad (45)$$

$$\lambda \mu_2 = (p_2 Y_2 - r_2) \frac{Y_2 + 1}{Y_2}, \quad (46)$$

$$-\lambda \mu_1 = (p_1 Y_1 - r_1) \frac{Y_1 + 1}{Y_1}. \quad (47)$$

If  $\mu_1 \neq \mu_2$ , equations (45)-(47) can be reduced to a single quadratic equation in  $Y_1$ :

$$-(\mu_2 - \mu_1) p_1 Y_1^2 + [(\mu_2 - \mu_1)(r_1 + r_2) - (\mu_2 p_1 + \mu_1 p_2)] Y_1 + \mu_2(r_1 + r_2) = 0. \quad (48)$$

Equation (48) has the two following solutions:

$$Y_{11} = \frac{[(\mu_2 - \mu_1)(r_1 + r_2) - (\mu_2 p_1 + \mu_1 p_2)]}{2(\mu_2 - \mu_1) p_1} + \frac{\sqrt{[(\mu_2 - \mu_1)(r_1 + r_2) - (\mu_2 p_1 + \mu_1 p_2)]^2 + 4\mu_2(\mu_2 - \mu_1) p_1(r_1 + r_2)}}{2(\mu_2 - \mu_1) p_1}, \quad (49)$$

$$Y_{12} = \frac{[(\mu_2 - \mu_1)(r_1 + r_2) - (\mu_2 p_1 + \mu_1 p_2)]}{2(\mu_2 - \mu_1) p_1} + \frac{\sqrt{[(\mu_2 - \mu_1)(r_1 + r_2) - (\mu_2 p_1 + \mu_1 p_2)]^2 + 4\mu_2(\mu_2 - \mu_1) p_1(r_1 + r_2)}}{2(\mu_2 - \mu_1) p_1}. \quad (50)$$

By substituting (49) and (50) in (45) and (46) it is possible to find out the expression for the remaining parameters  $Y_{21}$ ,  $Y_{22}$ ,  $\lambda_1$ ,  $\lambda_2$ .

Moreover, another feasible solution is the following:

$$Y_{13} = \frac{r_1}{p_1}, \quad (51)$$

$$Y_{23} = \frac{r_2}{p_2}, \quad (52)$$

$$\lambda_3 = 0. \quad (53)$$

Consequently, the solution for internal states can be expressed as

$$f^S(x, \alpha_1, \alpha_2) = \sum_{j=1}^3 C_j^S e^{\lambda_j x} Y_{1j}^{\alpha_1} Y_{2j}^{\alpha_2}, \quad (54)$$

where, from equation (53), the third component of the solution is constant.

If  $\mu_1 = \mu_2 = \mu$ , equation (48) reduces to a linear equation whose solution is

$$Y_{11} = \frac{r_1 + r_2}{p_1 + p_2}. \quad (55)$$

From the parametric equations (46) and (47) we obtain:

$$Y_{21} = \frac{r_1 + r_2}{p_1 + p_2} = Y_{11}, \quad (56)$$

$$\lambda_1 = \frac{1}{\mu} (r_1 p_2 - r_2 p_1) \left( \frac{1}{p_1 + p_2} + \frac{1}{r_1 + r_2} \right). \quad (57)$$

It is convenient to treat separately the three cases related to  $\mu_1 = \mu_2$ ,  $\mu_1 < \mu_2$  and  $\mu_1 > \mu_2$ .

*Case  $\mu_1 < \mu_2$*

The boundary conditions yield

$$\mathbf{p}(0, 0, 0) = 0, \quad (58)$$

$$\mathbf{p}(0, 1, 0) = 0, \quad (59)$$

$$\mathbf{p}(0, 0, 1) = \frac{\mu_2}{r_1} \left[ C_1^S \left( \frac{p_1}{p_2} Y_{11} + Y_{21} \right) + C_2^S \left( \frac{p_1}{p_2} Y_{12} + Y_{22} \right) + C_3^S \left( \frac{p_1}{p_2} Y_{13} + Y_{23} \right) \right], \quad (60)$$

$$\mathbf{p}(0, 1, 1) = \frac{\mu_2}{p_2} (C_1^S Y_{11} + C_2^S Y_{12} + C_3^S Y_{13}), \quad (61)$$

$$\mathbf{p}(N, 0, 0) = 0, \quad (62)$$

$$\mathbf{p}(N, 1, 0) = \frac{\mu_1}{r_2} (C_1^S e^{\lambda_1 N} Y_{11} + C_2^S e^{\lambda_2 N} Y_{12} + C_3^S Y_{13}), \quad (63)$$

$$\mathbf{p}(N, 0, 1) = 0, \quad (64)$$

$$\mathbf{p}(N, 1, 1) = 0. \quad (65)$$

It is possible to express the constants  $C_2^S$  and  $C_3^S$  in term of  $C_1^S$  by equation (16) and (17). Specifically,

$$C_2^S = -\frac{e^{\lambda_1 N}}{e^{\lambda_2 N}} \frac{Y_{21}(Y_{11} - Y_{13})}{Y_{22}(Y_{12} - Y_{13})} C_1^S, \quad (66)$$

and

$$C_3^S = e^{\lambda_1 N} \frac{Y_{21}}{Y_{23}} \left( \frac{Y_{11} - Y_{13}}{Y_{12} - Y_{13}} - 1 \right) C_1^S. \quad (67)$$

Therefore, the only unknown parameter in the *standard operation* partition model is  $C_1^S$ . The value of  $C_1^S$  can be found by means of the normalization equation for the *standard operation* partition in isolation, i.e. equation (14). Note that the Basic model without restart policy presents only two constants while  $C_3^S \neq 0$  in the *RP* model as a consequence of the restart policy.

Case  $\mu_1 > \mu_2$

The boundary probabilities are the following:

$$\mathbf{p}(0, 0, 0) = 0, \quad (68)$$

$$\mathbf{p}(0, 1, 0) = 0, \quad (69)$$

$$\begin{aligned} \mathbf{p}(0, 0, 1) = & \frac{(\mu_2 - \mu_1)}{r_1} (C_1^S Y_{11} Y_{21} (1 - e^{\lambda_1 N}) + C_2^S Y_{12} Y_{22} (1 - e^{\lambda_2 N})) + \\ & - \frac{\mu_1}{r_1} (C_1^S e^{\lambda_1 N} Y_{11} + C_2^S e^{\lambda_2 N} Y_{12} + C_3^S Y_{13}), \end{aligned} \quad (70)$$

$$\mathbf{p}(0, 1, 1) = 0, \quad (71)$$

$$\mathbf{p}(N, 0, 0) = 0, \quad (72)$$

$$\mathbf{p}(N, 1, 0) = \frac{\mu_1}{r_2} (C_1^S e^{\lambda_1 N} Y_{11} + C_2^S e^{\lambda_2 N} Y_{12} + C_3^S Y_{13}), \quad (73)$$

$$\mathbf{p}(N, 0, 1) = 0, \quad (74)$$

$$\mathbf{p}(N, 1, 1) = 0. \quad (75)$$

The constants  $C_2^S$  and  $C_3^S$  can be expressed in terms of  $C_1^S$  by equation (101), reported in Appendix A, evaluated at the steady state, and by equation (20).

Thus,

$$C_2^S = - \frac{e^{\lambda_1 N} Y_{21} Y_{13} - Y_{23} Y_{11}}{e^{\lambda_2 N} Y_{22} Y_{13} - Y_{23} Y_{12}} C_1^S, \quad (76)$$

and

$$C_3^S = \left[ - \frac{Y_{11}}{Y_{13}} + \frac{Y_{12}}{Y_{13}} \left( \frac{e^{\lambda_1 N} Y_{21} Y_{13} - Y_{23} Y_{11}}{e^{\lambda_2 N} Y_{22} Y_{13} - Y_{23} Y_{12}} \right) \right] C_1^S. \quad (77)$$

Even in this case, the only unknown parameter is  $C_1^S$  whose value can be obtain by the normalization equation for the *standard operation* partition in isolation (equation 14).

Case  $\mu_1 = \mu_2 = \mu$

In this case  $Y_{12} = 0$  and, consequently,  $Y_{22} = 0$  and  $\lambda_2 = 0$ . Nevertheless, it is still necessary to consider the parameters  $Y_{13}$  and  $Y_{23}$  given by equations (51) and (52) ( $\lambda_3 = 0$  from equation 53).

Consequently, the solution for internal states can be simply expressed as

$$f^S(x, \alpha_1, \alpha_2) = C_1^S e^{\lambda_1 x} Y_{11}^{\alpha_1 + \alpha_2} + C_3^S Y_{13}^{\alpha_1} Y_{23}^{\alpha_2}, \quad (78)$$

where, for the sake of clarity, the same notation used for the previous cases has been maintained.

The boundary conditions yield

$$\mathbf{p}(0,0,0) = 0, \quad (79)$$

$$\mathbf{p}(0,1,0) = 0, \quad (80)$$

$$\mathbf{p}(0,0,1) = \frac{\mu}{r_1} Y_{11} \left( \frac{p_1}{p_2} + 1 \right) C_1^S + \frac{\mu}{r_1} \left( \frac{p_1}{p_2} Y_{13} + Y_{23} \right) C_3^S, \quad (81)$$

$$\mathbf{p}(0,1,1) = \frac{\mu}{p_2} (C_1^S Y_{11} + C_3^S Y_{13}), \quad (82)$$

$$\mathbf{p}(N,0,0) = 0, \quad (83)$$

$$\mathbf{p}(N,1,0) = \frac{\mu}{r_2} [C_1^S e^{\lambda_1 N} Y_{11} + C_3^S Y_{13}], \quad (84)$$

$$\mathbf{p}(N,0,1) = 0, \quad (85)$$

$$\mathbf{p}(N,1,1) = 0. \quad (86)$$

Recall that in the Basic model without restart policy the constant  $C_3^S$  in equation (78) is zero. On the contrary, here the relation between  $C_1^S$  and  $C_3^S$  can be found by considering equation (16) leading to the following:

$$C_3^S = -\frac{e^{\lambda_1 x} Y_{11}}{Y_{23}} C_1^S. \quad (87)$$

The normalization equation (14) allows us to find the value of  $C_1^S$  and complete the solution.

## 5.2 Buffer drainage partition in isolation

As regards the *buffer drainage* partition in isolation, a solution satisfying equations (22) and (23) is the following:

$$f^{*D}(\alpha_2) = C^D \left( \frac{r_2}{p_2} \right)^{\alpha_2}. \quad (88)$$

The value of  $C^D$  can be obtained by the normalization equation for the *buffer drainage* partition in isolation (equation 26) as follows:

$$N C^D \left( 1 + \frac{r_2}{p_2} \right) = 1, \quad (89)$$

so,

$$C^D = \frac{p_2}{N(p_2 + r_2)}. \quad (90)$$

## 5.3 System solution

Once the solutions have been determined for both the partitions in isolation, we can derive the probabilities of the system being in each partition from equation (34) and (35).

Finally, the solution to the original system can be expressed as follows:

$$f(\beta, x, \alpha_1, \alpha_2) = (1 - \beta) \pi^S \left( \sum_{j=1}^3 C_j^S e^{\lambda_j x} Y_{1j}^{\alpha_1} Y_{2j}^{\alpha_2} \right) + \beta \pi^D \frac{p_2}{N(p_2 + r_2)} \left( \frac{r_2}{p_2} \right)^{\alpha_2}, \quad (91)$$

where the constant parameters  $C_j$ ,  $Y_{ij}$ ,  $\lambda_j$  with  $i = 1, 2$  and  $j = 1, 2, 3$  are computed according to Section 5.

## 6 Numerical results

In the following some interesting numerical examples are proposed where the *RP* model described in this paper is compared with the Basic model without restart policy (see Gershwin and Schick, 1980; Gershwin, 2002). In this way it is possible to discuss the benefits that may derive from the adoption of the restart policy.

In particular, scenarios with  $\mu_1 e_1 > \mu_2 e_2$  and with  $\mu_1 e_1 < \mu_2 e_2$  are investigated. Configurations in which  $\mu_1 < \mu_2$ ,  $\mu_1 = \mu_2$ , and  $\mu_1 > \mu_2$  are evaluated as well.

Figure 2 depicts the blocking frequency of Example 1a, whose input data are reported in Table 2. In this example the isolated productivity of the first machine  $\mu_1 e_1$  is greater than that of the second machine  $\mu_2 e_2$ , while  $\mu_1 \leq \mu_2$ . Light lines represent the *RP* model and bold lines represent the Basic model. Same line type means same input data. As can be easily seen, when the restart policy is not adopted, the blocking frequency approaches a limit greater than zero as the buffer capacity increases. Thus, when  $\mu_1 e_1 > \mu_2 e_2$ ,

- if the restart policy is not adopted, there is a nonzero probability of the buffer filling up even if large buffer capacities are involved;
- the introduction of the restart policy makes it possible to significantly reduce the blocking frequency, allowing it to tend to zero when the buffer capacity is large enough.

Therefore, if  $\mu_1 e_1 > \mu_2 e_2$  and the outage costs on the first machine are critical (so that the blocking frequency results to be a key performance measure that should be taken as low as possible), the adoption of the restart policy results to be convenient. This situation occurs, e.g., in automated packaging lines of the food and beverage sector.

[PUT TABLE 2 HERE]

[PUT FIGURE 2 HERE]

The same result is obtained in Figure 3 reporting Example 1b (refer to Table 3). While  $\mu_1 e_1$  is still greater than  $\mu_2 e_2$ , now  $\mu_1 \geq \mu_2$ .

[PUT TABLE 3 HERE]

[PUT FIGURE 3 HERE]

Finally, Figure 4 reports the results of Example 2 representing scenarios with  $\mu_1 e_1 < \mu_2 e_2$  (refer to Table 4). As in previous figures, light lines represent the *RP* model and bold lines the Basic model, and same line type means same input data. In such situations, the blocking frequency naturally approaches the limit zero as the buffer capacity increases. This is true for both models, with or without restart policy. What can be noted is that the restart policy affects the rapidity of the convergence. This is especially evident for the case  $\mu_1 > \mu_2$  (see Table 4 and the dashed line in Figure 4) where the restart policy makes it possible to reduce the blocking frequency even when small buffers are adopted.

[PUT TABLE 4 HERE]

[PUT FIGURE 4 HERE]

Thus, if  $\mu_1 e_1 < \mu_2 e_2$  it is possible to reduce the blocking frequency by either using a large buffer or adopting the restart policy. Moreover, it is important to consider that if, on one hand, the restart policy has a beneficial effect on the blocking frequency; on the other hand, the probability of starvation increases. Figure 5 shows the starvation probability for Example 2. Since the adoption of the restart policy implies an higher starvation probability, it may be convenient only if outage costs on the first machine are highly critical and there exist constraints on the buffer capacity.

[PUT FIGURE 5 HERE]

In general, the decision whether or not to adopt the restart policy would be based on a carefully considered evaluation of other constraints that are out of the scope of this work (e.g., buffer capacity constraints and costs). This is especially true if  $\mu_1 e_1 < \mu_2 e_2$ .

## 7 Conclusions

The work addresses the performance estimation of a 2M-1B production line in which a control policy is adopted to control the machines' behavior according to a specific event happening in the line, i.e. the buffer filling up. Since the introduction of such a control policy increases the complexity of the problem, a modeling approach based on the partitioning of the state space has been developed so as to facilitate mathematical tractability.

The production line under study consists of two machines decoupled with a finite buffer where a restart control policy (*RP*) is introduced on the first machine. The aim is to prevent the very machine from producing parts each time the buffer gets full until it empties again. This policy is frequently adopted in industrial installations where outage costs (e.g. production of a certain amount of waste) are generated during the restart phase of the machines.

The *RP* model is developed as a continuous time Markov process so as to allow the consideration of machines having different capacities and working in an asynchronous manner. The exact analytical solution of the model is provided and the conservation of flow is proved. Moreover, the expression of the most important performance measures is derived.

Numerical examples prove the ability of the *RP* model to represent the effects of the adopted restart policy on the blocking frequency (and, as a consequence, on the line efficiency), as a function of the buffer capacity and the machines' parameters. The resulting model represents an important tool able to point out the convenience of adopting a restart policy in a production line and to measure its effects as a function of the line characteristics.

**Acknowledgements** The authors would like to thank Dr. Stanley B. Gershwin for his useful suggestions and comments.

## References

- Bulgak AA, Diwan PD, Inozu B (1995) Buffer size optimization in asynchronous assembly systems using genetic algorithms. *Computers & Industrial Engineering* 28:309–322
- Burman MH (1995) New results in flow line analysis. Ph.D. thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology

- Buzacott JA (1967a) Automatic transfer lines with buffer stocks. *International Journal of Production Research* 5:183–200
- Buzacott JA (1967b) Markov chain analysis of automatic transfer line with buffer stock. Ph.D. thesis, University of Birmingham
- Chiang SY, Kuo CT, Meerkov SM (2000) DT-bottlenecks in serial production lines: theory and application. *IEEE Transactions on Robotics and Automation* 16:567–580
- Chiang SY, Kuo CT, Meerkov SM (2001) c-Bottlenecks in serial production lines: Identification and application. *Mathematical Problems in Engineering* 7:543–578
- Chiang SY, Hu A, Meerkov SM (2008) Lean buffering in serial production lines with non-identical exponential machines. *IEEE Transactions on Automation Science and Engineering* 5:298–306
- Choong YF, Gershwin SB (1989) A decomposition method for the approximate evaluation of capacitated transfer lines with unreliable machines and random processing times. *IEEE Transactions* 19:150–159
- Dallery Y, Frein Y (1993) On decomposition methods for tandem queueing networks with blocking. *Operations Research* 41(2):386–399
- Dallery Y, Gershwin SB (1992) Manufacturing flow line systems: a review of models and analytical results. *Queueing Systems* 12:3–94
- Dallery Y, David R, Xie XL (1989) Approximate analysis of transfer lines with unreliable machines and finite buffers. *IEEE Transactions on Automatic Control* 34(9):943–953
- Enginarlar E, Meerkov SM (2005) Lean buffering in serial production lines with non-exponential machines. *OR Spectrum* 27:195–219
- Gebennini E, Grassi A, Fantuzzi C, Gershwin SB, Schick IC (2009) On the introduction of a restart policy in the two-machine, one-buffer transfer line model. In: *Proceedings of the 7<sup>th</sup> International Conference on Stochastic Models of Manufacturing and Service Operations*, Ostuni, Italy, pp 81–88
- Gebennini E, Grassi A, Fantuzzi C, Gershwin SB, Schick IC (in press) Discrete time model for two-machine one-buffer transfer lines with restart policy. *Annals of Operations Research*
- Gershwin SB (1987) An efficient decomposition method for the approximate evaluation of tandem queues with finite storage space and blocking. *Operations Research* 35(2):291–305
- Gershwin SB (2002) *Manufacturing Systems Engineering*. Massachusetts Institute of Technology. Second private printing, available via <http://web.mit.edu/manuf-sys/www/gershwin.book.html>
- Gershwin SB, Burman MH (2000) A decomposition method for analyzing inhomogeneous assembly/disassembly systems. *Annals of Operations Research* 93:91–115
- Gershwin SB, Schick IC (1980) Continuous model of an unreliable two-stage material flow system with a finite interstage buffer. Tech. Rep. LIDS-R-1039, OSP No. 87049, Laboratory for Information and Decision Systems, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA
- Gershwin SB, Schor JE (2000) Efficient algorithms for buffer space allocation. *Annals of Operations Research* 93:117–144
- Gershwin SB, Werner LM (2007) An approximate analytical method for evaluating the performance of closed-loop flow systems with unreliable machines and finite buffers. *International Journal of Production Research* 45(14):3085–3111
- Kim J, Gershwin SB (2005) Integrated quality and quantity modeling of a production line. *OR Spectrum* 27(2–3):287–314
- Koster MBMD (1987) Estimation of line efficiency by aggregation. *International Journal of Production Research* 25:615–626
- Levantese R, Matta A, Tolio T (2003) Performance evaluation of continuous production lines with machines having different processing times and multiple failure modes. *Performance Evaluation* 51:247–



268

- Lutz CM, Davis KR, Sun M (1998) Determining buffer location and size in production lines using tabu search. *European Journal of Operational Research* 106:301–316
- MacGregor Smith J, Daskalaki S (1988) Buffer space allocation in automated assembly lines. *Operations Research* 36(2):343–358
- Maggio N, Matta A, Gershwin SB, Tolio T (2003) An approximate analytical method for evaluating the performance of closed loop flow systems with unreliable machines and finite buffers – Part I: Small loops. Tech. rep., Laboratory for Manufacturing and Productivity, Massachusetts Institute of Technology, Cambridge, Massachusetts 02139, USA
- Papadopoulos H, Heavey C, Browne J (1993) *Queuing Theory in Manufacturing Systems Analysis and Design*. Chapman & Hall, London
- Papadopoulos HT, Vidalis MI (1998) Optimal buffer storage allocation in balanced reliable production lines. *Int Trans Opl Res* 5(4):325–339
- Papadopoulos HT, Vidalis MI (1999) Optimal buffer allocation in short  $\mu$ -balanced unreliable production lines. *Computers & Industrial Engineering* 37:691–710
- Papadopoulos HT, Vidalis MI (2001a) A heuristic algorithm for the buffer allocation in unreliable unbalanced production lines. *Computers & Industrial Engineering* 41:261–277
- Papadopoulos HT, Vidalis MI (2001b) Minimizing WIP inventory in reliable production lines. *International Journal of Production Economics* 70:185–197
- Perrica G, Fantuzzi C, Grassi A, Goldoni G, Raimondi F (2008) Time to failure and time to repair profiles identification. In: *Proceedings of the FOODSIM 2008*, Dublin, Ireland, pp 59–64
- Spinellis DD, Papadopoulos CT (2000) A simulated annealing approach for buffer allocation in reliable production. *Annals of Operations Research* 93:373–384
- Tan B, Gershwin SB (2009) Analysis of a general markovian two-stage continuous-flow production system with a finite buffer. *International Journal of Production Economics* 120(2):327–339
- Tan B, Gershwin SB (2011) Modelling and analysis of markovian continuous flow systems with a finite buffer. *Annals of Operations Research* 182(1):5–30
- Tan B, Yeralan S (1997) Analysis of multistation production systems with limited buffer capacity. Part II: The decomposition method. *Mathematical and Computer Modelling* 25(11):109–123
- Tolio T (2011) Performance evaluation of two-machines line with multiple up and down states and finite buffer capacity. In: *Proceedings of the 8<sup>th</sup> International Conference on Stochastic Models of Manufacturing and Service Operations*, Kusadasi, Turkey, pp 117–127
- Tolio T, Matta A, Gershwin SB (2002) Analysis of two-machine lines with multiple failure modes. *IIIE Transactions* 34(1):51–62
- Yeralan S, Tan B (1997) Analysis of multistation production systems with limited buffer capacity. Part I: The subsystem model. *Mathematical and Computer Modelling* 25(7):109–122
- Zimmern B (1956) Études de la propagation des arrêts aléatoires dans les chaînes de production. *Rev Statist Appl* 4:85–104

## A Partitions in isolation: Model Equations

### A.1 Standard Operation Partition in Isolation

#### *Intermediate Buffer Levels*

The internal equations are the same as the Basic model (see Gershwin and Schick, 1980; Gershwin, 2002).

$$\frac{\partial f^S}{\partial t}(x, 1, 1) = -(p_1 + p_2)f^S(x, 1, 1) + (\mu_2 - \mu_1)\frac{\partial f^S}{\partial x}(x, 1, 1) + r_1f^S(x, 0, 1) + r_2f^S(x, 1, 0), \quad (92)$$

$$\frac{\partial f^S}{\partial t}(x, 0, 0) = -(r_1 + r_2)f^S(x, 0, 0) + p_1f^S(x, 1, 0) + p_2f^S(x, 0, 1), \quad (93)$$

$$\frac{\partial f^S}{\partial t}(x, 0, 1) = \mu_2\frac{\partial f^S}{\partial x}(x, 0, 1) - (r_1 + p_2)f^S(x, 0, 1) + p_1f^S(x, 1, 1) + r_2f^S(x, 0, 0), \quad (94)$$

$$\frac{\partial f^S}{\partial t}(x, 1, 0) = -\mu_1\frac{\partial f^S}{\partial x}(x, 1, 0) - (p_1 + r_2)f^S(x, 1, 0) + p_2f^S(x, 1, 1) + r_1f^S(x, 0, 0). \quad (95)$$

#### *Lower Boundary – $x = 0$*

The equations modified to include the restart policy are equation (100) and equation (102).

– *Boundary-to-Boundary Equations*

$$\frac{d}{dt}\mathbf{p}^S(0, 0, 0) = -(r_1 + r_2)\mathbf{p}^S(0, 0, 0) + p_1\mathbf{p}^S(0, 1, 0), \quad (96)$$

$$\mathbf{p}^S(0, 1, 0) = 0. \quad (97)$$

– *Interior-to-Boundary Equations*

$$\frac{d}{dt}\mathbf{p}^S(0, 0, 1) = r_2\mathbf{p}^S(0, 0, 0) - r_1\mathbf{p}^S(0, 0, 1) + p_1\mathbf{p}^S(0, 1, 1) + \mu_2f^S(0, 0, 1), \quad (98)$$

$$\begin{aligned} \frac{d}{dt}\mathbf{p}^S(0, 1, 1) = & -(p_1 + p_2^b)\mathbf{p}^S(0, 1, 1) + r_1\mathbf{p}^S(0, 0, 1) + (\mu_2 - \mu_1)f^S(0, 1, 1) + \\ & + r_2\mathbf{p}^S(N, 1, 0) \quad \text{if } \mu_1 \leq \mu_2, \end{aligned} \quad (99)$$

$$\mathbf{p}^S(0, 1, 1) = 0 \quad \text{if } \mu_1 > \mu_2. \quad (100)$$

– *Boundary-to-Interior Equations*

$$\mu_1f^S(0, 1, 0) = r_1\mathbf{p}^S(0, 0, 0) + p_2^b\mathbf{p}^S(0, 1, 1), \quad (101)$$

$$(\mu_1 - \mu_2)f^S(0, 1, 1) = r_1\mathbf{p}^S(0, 0, 1) + r_2\mathbf{p}^S(N, 1, 0) + (\mu_1 - \mu_2)f^S(N, 1, 1) \quad \text{if } \mu_1 > \mu_2. \quad (102)$$

#### *Upper Boundary – $x = N$*

The equations modified to include the restart policy are equation (107) and equation (108). Moreover, if a restart policy is applied, equation (105) holds in all the cases of interest (i.e.  $\mu_1 < \mu_2$ ,  $\mu_1 = \mu_2$  and  $\mu_1 > \mu_2$ ).

– *Boundary-to-Boundary Equations*

$$\frac{d}{dt}\mathbf{p}^S(N, 0, 0) = -(r_1 + r_2)\mathbf{p}^S(N, 0, 0). \quad (103)$$

– *Interior-to-Boundary Equations*

$$\frac{d}{dt}\mathbf{p}^S(N, 1, 0) = r_1\mathbf{p}^S(N, 0, 0) - r_2\mathbf{p}^S(N, 1, 0) + p_2\mathbf{p}^S(N, 1, 1) + \mu_1f^S(N, 1, 0), \quad (104)$$

$$\mathbf{p}^S(N, 1, 1) = 0. \quad (105)$$

– *Boundary-to-Interior Equations*

$$\mathbf{p}^S(N, 0, 1) = 0, \quad (106)$$

$$f^S(N, 0, 1) = 0. \quad (107)$$

$$f^S(N, 1, 1) = 0 \quad \text{if } \mu_1 < \mu_2. \quad (108)$$

### Normalization

$$\sum_{\alpha_1=0}^1 \sum_{\alpha_2=0}^1 \left[ \int_0^N f^S(x, \alpha_1, \alpha_2) dx + \mathbf{p}^S(0, \alpha_1, \alpha_2) + \mathbf{p}^S(N, \alpha_1, \alpha_2) \right] = 1. \quad (109)$$

## A.2 Buffer Drainage Partition in Isolation

### Intermediate Buffer Levels

$$\frac{d}{dt} f^{*D}(1) = -p_2 f^{*D}(1) + r_2 f^{*D}(0). \quad (110)$$

$$\frac{d}{dt} f^{*D}(0) = -p_2 f^{*D}(1) + r_2 f^{*D}(0). \quad (111)$$

### Normalization

$$\sum_{\alpha_2=0}^1 \int_0^N f^{*D}(\alpha_2) dx = 1. \quad (112)$$

## B Conservation of flow: Proof

For the system to be in the steady state, the following equation (conservation of flow equation) must be verified:

$$P_1 = P_2. \quad (113)$$

**Proof:** For the sake of clarity, the expression of the total system solution can be split into the components related to the two partitions as follows:

$$P_1 = \pi^S \mu_1 \left[ \int_0^N (f^S(x, 1, 0) + f^S(x, 1, 1)) dx + \mathbf{p}^S(0, 1, 1) \right], \quad (114)$$

$$P_2 = \pi^S \left[ \mu_2 \int_0^N (f^S(x, 0, 1) + f^S(x, 1, 1)) dx + \mu_1 \mathbf{p}^S(0, 1, 1) \right] + \pi^D \mu_2 \left[ \int_0^N f^{*D}(1) dx \right] \quad (115)$$

By adding the steady state versions of the internal differential equations (92)-(95) in Appendix A, we obtain

$$\frac{d}{dx} [(\mu_2 - \mu_1) f^S(x, 1, 1) + \mu_2 f^S(x, 0, 1) - \mu f^S(x, 1, 0)] = 0. \quad (116)$$

Therefore,

$$(\mu_2 - \mu_1) f^S(x, 1, 1) + \mu_2 f^S(x, 0, 1) - \mu f^S(x, 1, 0) = K, \quad (117)$$

where  $K$  is a constant that can be found by evaluating (117) for particular values of  $x$  and distinguishing between  $\mu_1 \leq \mu_2$  and  $\mu_1 > \mu_2$ .

Consider first the case  $\mu_1 \leq \mu_2$  and  $x = 0$ , by adding equations (98), (99) and (101), in Appendix A, evaluated at steady state and considering non-zero probabilities only, we obtain

$$(\mu_2 - \mu_1) f^S(0, 1, 1) + \mu_2 f^S(0, 0, 1) - \mu f^S(0, 1, 0) = -r_2 \mathbf{p}^S(N, 1, 0). \quad (118)$$

On the other hand, if  $\mu_1 > \mu_2$  and we add equations (98), (100) and (101) reported in Appendix A, it results

$$(\mu_2 - \mu_1)f^S(0, 1, 1) + \mu_2 f^S(0, 0, 1) - \mu f^S(0, 1, 0) = -r_2 \mathbf{p}^S(N, 1, 0) - (\mu_1 - \mu_2)f^S(N, 1, 1). \quad (119)$$

Thus,

$$K = \begin{cases} -r_2 \mathbf{p}^S(N, 1, 0), & \text{if } \mu_1 \leq \mu_2, \\ -r_2 \mathbf{p}^S(N, 1, 0) - (\mu_1 - \mu_2)f^S(N, 1, 1), & \text{if } \mu_1 > \mu_2 \end{cases}. \quad (120)$$

Moreover, from the steady state version of equation (104) in Appendix A, we have:

$$r_2 \mathbf{p}^S(N, 1, 0) = \mu_1 f^S(N, 1, 0),$$

and consequently,

$$K = \begin{cases} -\mu_1 f^S(N, 1, 0), & \text{if } \mu_1 \leq \mu_2, \\ -\mu_1 f^S(N, 1, 0) - (\mu_1 - \mu_2)f^S(N, 1, 1), & \text{if } \mu_1 > \mu_2 \end{cases}. \quad (121)$$

The same value for  $K$  can be found at  $x = N$  considering the equations related to the upper boundary.

Subtracting (114) from (115) yields

$$P_2 - P_1 = \pi^S \int_0^N [(\mu_2 - \mu_1)f^S(x, 1, 1) + \mu_2 f^S(x, 0, 1) - \mu_1 f^S(x, 1, 0)] dx + \pi^D \mu_2 \int_0^N f^{*D}(1) dx, \quad (122)$$

where equation (122) is valid for both  $\mu_1 \leq \mu_2$  and  $\mu_1 > \mu_2$ .

Finally, considering equations (117) and (121) and recalling the expressions for  $\pi^S$  and  $\pi^D$  (equations (34) and (35)), we obtain

$$\begin{aligned} P_2 - P_1 &= \pi^S [-\mu_1 f^S(N, 1, 0)]N + \pi^D \mu_2 f^{*D}(1)N = \\ &= -\frac{\mu_2 f^{*D}(1)}{\mu_2 f^{*D}(1) + \mu_1 f^S(N, 1, 0)} \mu_1 f^S(N, 1, 0)N + \frac{\mu_1 f^S(N, 1, 0)}{\mu_1 f^{*D}(1) + \mu_2 f^S(N, 1, 0)} \mu_2 f^{*D}(1)N = \\ &= 0, \quad \text{if } \mu_1 \leq \mu_2, \end{aligned} \quad (123)$$

and

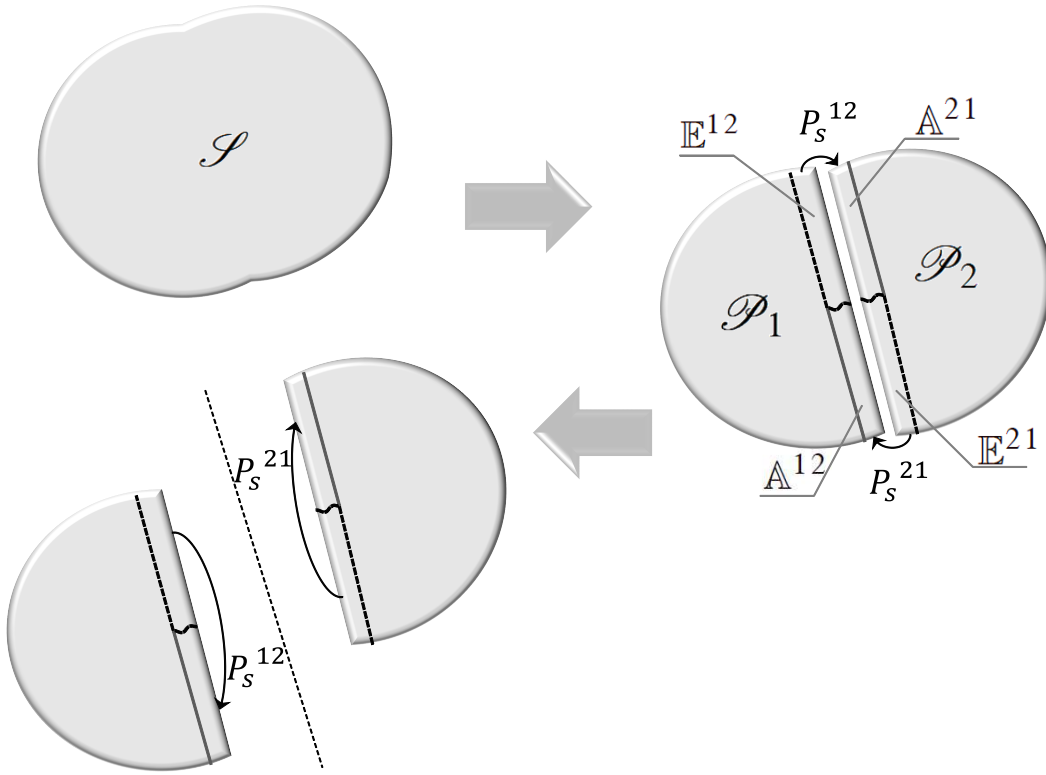
$$\begin{aligned} P_2 - P_1 &= \pi^S [-(\mu_1 f^S(N, 1, 0) + (\mu_1 - \mu_2)f^S(N, 1, 1))]N + \pi^D \mu_2 f^{*D}(1)N = \\ &= -\frac{\mu_2 f^{*D}(1)}{\mu_2 f^{*D}(1) + \mu_1 f^S(N, 1, 0) + (\mu_1 - \mu_2)f^S(N, 1, 1)} (\mu_1 f^S(N, 1, 0) + (\mu_1 - \mu_2)f^S(N, 1, 1))N + \\ &\quad + \frac{\mu_1 f^S(N, 1, 0) + (\mu_1 - \mu_2)f^S(N, 1, 1)}{\mu_2 f^{*D}(1) + \mu_1 f^S(N, 1, 0) + (\mu_1 - \mu_2)f^S(N, 1, 1)} \mu_2 f^{*D}(1)N = \\ &= 0, \quad \text{if } \mu_1 > \mu_2. \end{aligned} \quad (124)$$

Therefore, equation (113) is proved.

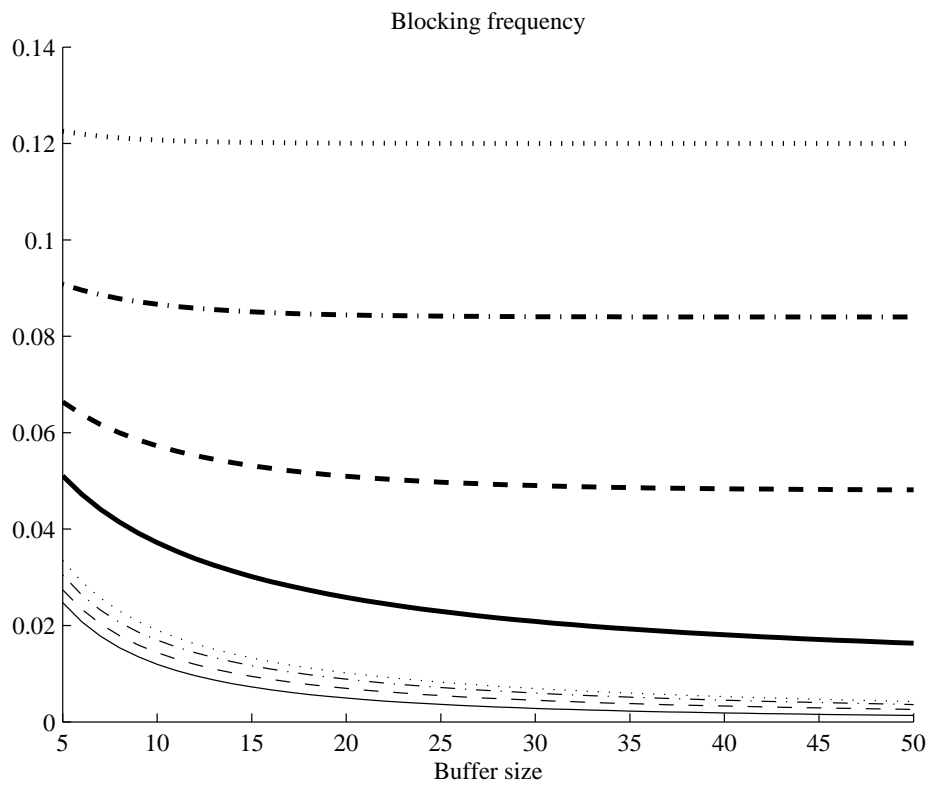
---

**List of Figures**

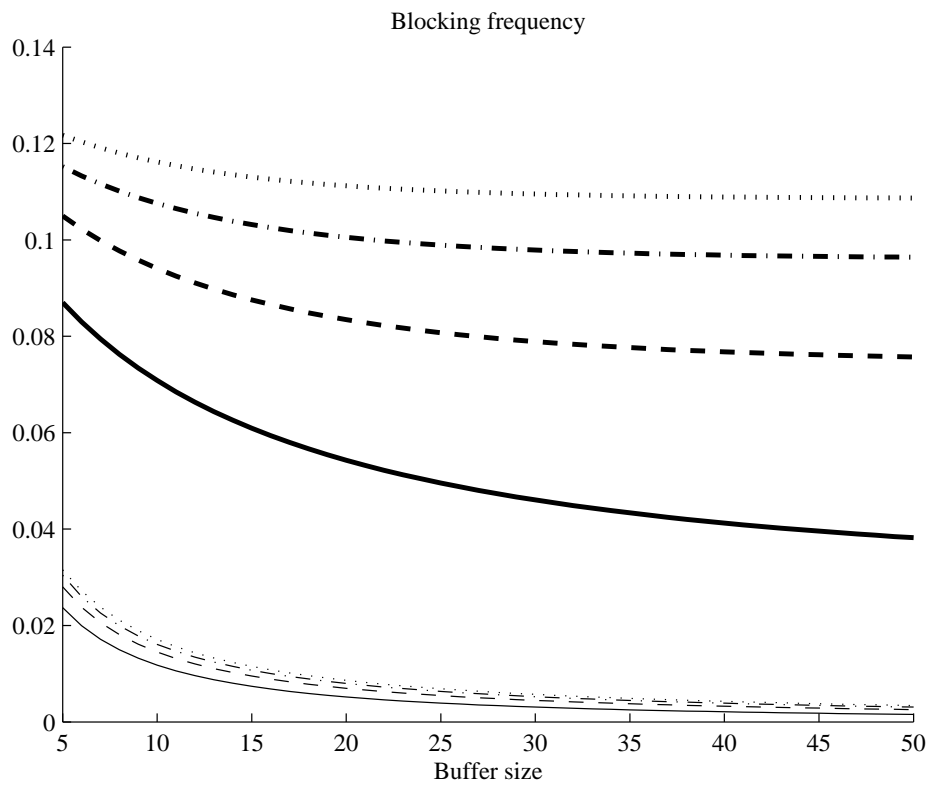
1	Step 1 and Step 2 of the modeling approach. . . . .	29
2	Blocking frequency for Example 1a: bolt lines for the Basic model; light lines for the <i>RP</i> model. . . . .	30
3	Blocking frequency for Example 1b: bolt lines for the Basic model; light lines for the <i>RP</i> model. . . . .	31
4	Blocking frequency for Example 2: bolt lines for the Basic model; light lines for the <i>RP</i> model. . . . .	32
5	Starvation probability for Example 2: bolt lines for the Basic model; light lines for the <i>RP</i> model. . . . .	33



**Fig. 1** Step 1 and Step 2 of the modeling approach.

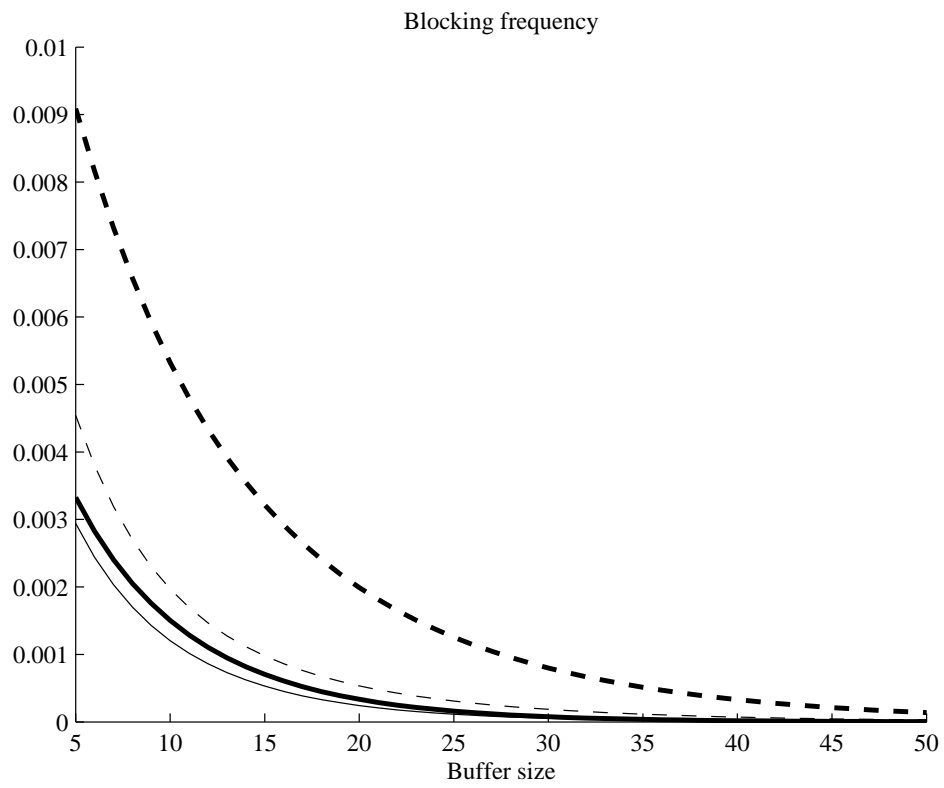


**Fig. 2** Blocking frequency for Example 1a: bolt lines for the Basic model; light lines for the *RP* model.

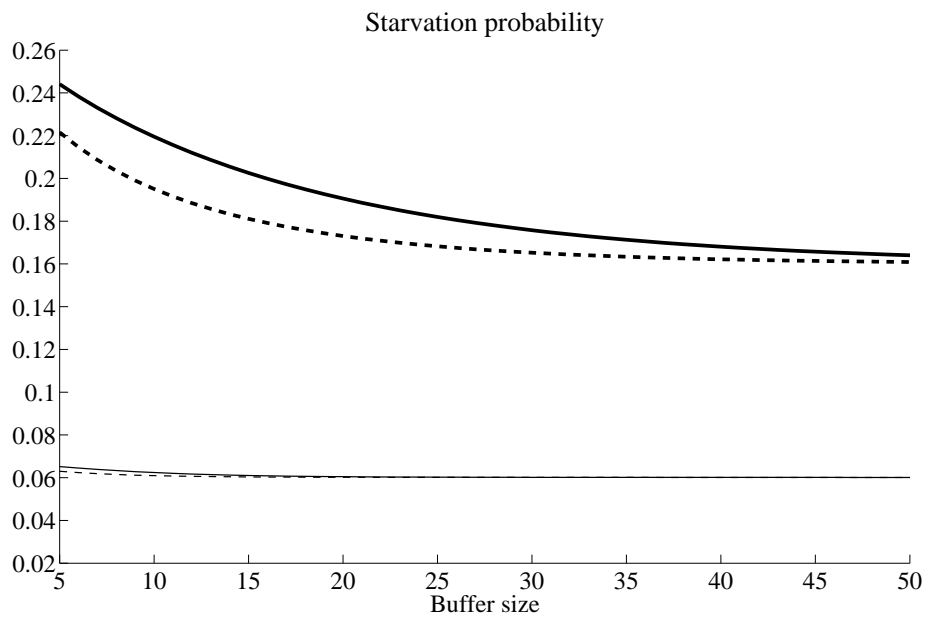


**Fig. 3** Blocking frequency for Example 1b: bold lines for the Basic model; light lines for the  $RP$  model.





**Fig. 4** Blocking frequency for Example 2: bold lines for the Basic model; light lines for the *RP* model.



**Fig. 5** Starvation probability for Example 2: bolt lines for the Basic model; light lines for the  $RP$  model.

---

**List of Tables**

1	Exit and arrival states of the <i>standard operation</i> and <i>buffer drainage</i> partition. . . . .	35
2	Input data for Example 1a ( $\mu_1 e_1 > \mu_2 e_2$ with $\mu_1 \leq \mu_2$ ). . . . .	36
3	Input data for Example 1b ( $\mu_1 e_1 > \mu_2 e_2$ with $\mu_1 \geq \mu_2$ ). . . . .	37
4	Input data for Example 2 ( $\mu_1 e_1 < \mu_2 e_2$ ). . . . .	38

	$\mu_1 \leq \mu_2$		$\mu_1 > \mu_2$	
	<i>standard op.</i>	<i>buffer drain.</i>	<i>standard op.</i>	<i>buffer drain.</i>
Exit States	$(0, N, 1, 0)$	$(1, x, 1, 1), 0 < x \leq \mu_2 \delta t$	$(0, N, 1, 0)$ $(0, x, 1, 1), N - (\mu_1 - \mu_2) \delta t \leq x < N$	$(1, x, 1, 1), 0 < x \leq \mu_2 \delta t$
Arrival States	$(0, 0, 1, 1)$	$(1, x, 1, 1), N - \mu_2 \delta t \leq x < N$	$(0, x, 1, 1), 0 < x \leq (\mu_1 - \mu_2) \delta t$	$(1, x, 1, 1), N - \mu_2 \delta t \leq x < N$

**Table 1** Exit and arrival states of the *standard operation* and *buffer drainage* partition.

$r_1$	$r_2$	$p_1$	$p_2$	$\mu_1$	$\mu_2$	$e_1$	$e_2$	$\mu_1 e_1$	$\mu_2 e_2$	type of curve
0.2	0.3	0.02	0.25	1	1	0.91	0.55	0.91	0.55	solid
0.2	0.3	0.02	0.25	1	1.2	0.91	0.55	0.91	0.65	dashed
0.2	0.3	0.02	0.25	1	1.4	0.91	0.55	0.91	0.76	dotted-dashed
0.2	0.3	0.02	0.25	1	1.6	0.91	0.55	0.91	0.87	dotted

**Table 2** Input data for Example 1a ( $\mu_1 e_1 > \mu_2 e_2$  with  $\mu_1 \leq \mu_2$ ).

$r_1$	$r_2$	$p_1$	$p_2$	$\mu_1$	$\mu_2$	$e_1$	$e_2$	$\mu_1 e_1$	$\mu_2 e_2$	type of curve
0.1	0.4	0.05	0.25	1	1	0.67	0.62	0.67	0.62	solid
0.1	0.4	0.05	0.25	1.2	1	0.67	0.62	0.80	0.62	dashed
0.1	0.4	0.05	0.25	1.4	1	0.67	0.62	0.93	0.62	dotted-dashed
0.1	0.4	0.05	0.25	1.6	1	0.67	0.62	1.07	0.62	dotted

**Table 3** Input data for Example 1b ( $\mu_1 e_1 > \mu_2 e_2$  with  $\mu_1 \geq \mu_2$ ).

$r_1$	$r_2$	$p_1$	$p_2$	$\mu_1$	$\mu_2$	$e_1$	$e_2$	$\mu_1 e_1$	$\mu_2 e_2$	type of curve
0.4	0.2	0.03	0.01	1	1.1	0.93	0.95	0.93	1.05	solid
0.1	0.4	0.05	0.02	1.2	1	0.67	0.95	0.80	0.95	dashed

**Table 4** Input data for Example 2 ( $\mu_1 e_1 < \mu_2 e_2$ ).