Exact Analysis of a Continuous Material Flow Line with Limited Buffer Capacity and a Merging Flow of Material

Stefan Helber
Nicole Mehrtens
Institut für Wirtschaftswissenschaft
Fakultät für Bergbau, Hüttewesen und Maschinenwesen
Technische Universität Clausthal
Julius-Albert-Str. 2
D-38678 Clausthal-Zellerfeld

stefan.helber@tu-clausthal.de

November 1999

Abstract

We develop a Markov process model of a flow line consisting of three unreliable machines and one buffer of limited capacity. Two machines upstream of the buffer perform the same operation and feed the same buffer in such a way that one machine has priority over the other when the buffer is full. The third machine removes material from the buffer. Processing times are deterministic and machine-specific. Exponentially distributed times to failure and to repair are also machine-specific. We develop an exact procedure to compute the average production rate and inventory level. The procedure is intended to be used as a building block of decomposition methods for the analysis of flow lines with non-linear flow of material that may, for example, be due to rework loops in flow lines with quality inspections.

1 Introduction

1.1 The Problem of Merging Flow of Material

Complex flow lines with limited buffer capacity and stochastic disruptions in the flow of material can be analyzed using decomposition methods of the type described in [9]. In
these methods, a flow line with multiple machines and buffers is decomposed into a set of virtual two-machine lines, one for each buffer. One tries to find parameters for these virtual machines such that the flow of material through the buffers of the virtual line mimics those through the buffers in the original system. The reason for this procedure is that two-machine lines are often analytically tractable whereas more complex systems are often intractable.

In practical situations, however, there may be situations where more than one machine performs one particular operation and where these machines feed the same buffer as depicted in Figure 1, for example because the machines upstream of the buffer are slower than the downstream machine.

![Figure 1: Flow line with a merging flow of material](image1)

In Figure 1, the squares indicate machines and the circle represents a buffer of limited capacity. The flow of material follows the directed arcs. Such a merge configuration may also serve as a building block of a decomposition approach for the analysis of a flow line with a rework loop like the one depicted in Figure 2.

![Figure 2: Flow line with a rework loop](image2)

In this system, there is a quality inspection at Machine $M_6$ and bad parts are reworked at Machines $M_7$ and $M_8$ before they are sent back to $M_2$. Assume that, from the point of view of $M_2$, there is no difference between the new and the reworked parts in the buffer upstream of $M_2$. In order to be able to analyze the flow of material in Figure 2 through a decomposition approach, one must be able to analyze a sub-system like the one in Figure 1 exactly, i.e. without falling back on decomposition.

Flow lines are often automated and therefore subject to random machine failures and repairs. These failures lead to random disruptions in the flow of material. Processing times, however, are often deterministic and machine-specific. These features should be considered in a useful model of a merge system.
Stochastic models of dynamic systems tend to be easiest to analyze whenever they possess the Markov property, i.e. whenever the future behavior of the system depends only on its current state, but not on its history. Deterministic processing times for the processing of discrete parts in flow lines, however, cannot easily be modeled in Markov process models. The reason is that in the case of deterministic processing times, the end of an operation can be forecasted, given the knowledge of when the operation started. In this case the Markov property is lost and the system is difficult to analyze. It is possible to escape this problem by defining the processing time as the time unit in systems where all processing times are identical (see [1, 8]), but this approach excludes the general case of systems with machine-specific processing times. The standard approach to model deterministic and machine-specific processing times in a Markov process model is to assume that the machines process a liquid instead of discrete parts. It helps to imagine that the originally discrete parts have been molten and are processed continuously by machines that operate like pumps (see [12, 11, 7]). In this case it is no longer possible to distinguish separate parts and therefore it is also not possible to predict the end of their processing, i.e. the future system behavior, based on the knowledge of its history. For this reason we now develop a continuous material Markov process of a merge system.

1.2 The Continuous Material Merge Model

The system in Figure 1 consists of three machines denoted as \( M_i, i = 1, ..., 3 \), and a buffer of capacity \( N \). The supply of material upstream of \( M_1 \) and \( M_2 \) is unlimited and so is the available space downstream of \( M_3 \). Machines \( M_1 \) and \( M_2 \) send material into the buffer and \( M_3 \) removes material from the buffer. The inventory level in the buffer is denoted as \( x \). Machines are unreliable and machine failures are operation dependent, i.e. machines can only fail while they operate. Failures and repairs of different machines are stochastically independent. The state of Machine \( M_i \) is denoted as \( \alpha_i \) such that \( \{ \alpha_i = 1 \} \) if \( M_i \) is operational (or up) and \( \{ \alpha_i = 0 \} \) if \( M_i \) is under repair (or down). The complete description of the system state is therefore \( (x, \alpha_1, \alpha_2, \alpha_3) \).

Times to failure of the machines \( M_i, i = 1, ..., 3 \), are exponentially distributed. The rate of failures depends on the instantaneous processing rate as we assume operation dependent failures. Repair times are also exponentially distributed, with rate \( \eta \).

The instantaneous processing rate at time \( t \) of any machine that is operational depends on the buffer level \( x \). If the buffer is neither full nor empty \( (0 < x < N) \), Machines \( M_i, i = 1, ..., 3 \), in Figure 1 process continuous material at a constant rate \( \mu_i \). In this case the instantaneous failure rate is \( \phi_i \).

When the buffer is empty \( (x = 0) \) and both Machine \( M_1 \) and \( M_2 \) are down \( (\alpha_1 = \alpha_2 = 0) \), Machine \( M_3 \) cannot operate because it is (completely) starved. In this case it cannot fail as failures are operation dependent. However, if \( M_1 \) and \( M_2 \) are up and the combined processing speed \( \mu_1 + \mu_2 \) of Machines \( M_1 \) and \( M_2 \) is below the speed \( \mu_3 \) of
If Machine $M_3$ is slowed down by $M_1$ and $M_2$, i.e. it can operate only at the reduced speed $\mu_1 + \mu_2$ at which material flows through the buffer. If Machine $M_3$ is slowed down by $M_1$ and $M_2$, the buffer remains empty even though material is flowing through the buffer. In this case, $M_3$ is said to be partially starved. If $M_3$ is partially starved, it fails less often and the adjusted failure rate in this state is $\frac{\mu_1 + \mu_2}{\mu_3} \cdot p_3$.

This and other cases of partial starvation are discussed in more detail in the main section of the paper.

When the buffer is full ($n = N$) and Machine $M_3$ is down ($s_3 = 0$), both $M_1$ and $M_2$ cannot operate because they are (completely) blocked. In this case, they both cannot fail. If the buffer is full while Machine $M_3$ is up, Machine $M_1$ has priority over Machine $M_2$ to fill the buffer. In this case it is possible that $M_1$ is partially blocked and $M_2$ can be partially or completely blocked. Assume, for example, that all machines are up, and that both $M_1$ and $M_2$ are (in isolation) slower than $M_3$, but their combined speed exceeds that of $M_3$, i.e. we have $\mu_1 < \mu_3 \land \mu_2 < \mu_3 \land (\mu_1 + \mu_2 > \mu_3)$. If the buffer is full ($x = N$) and all machines are up, Machine $M_1$ can operate at its normal speed $\mu_1$ because it has priority over $M_2$ and $M_3$ is faster than $M_1$, i.e. $M_3$ cannot partially block $M_1$. The situation of Machine $M_2$, however, is different. Due to the priority of Machine $M_1$, Machine $M_2$ can only fill material into the buffer according to the remaining capacity of $M_3$, i.e. at rate $\mu_3 - \mu_1$. In this situation $M_2$ is partially blocked and fails at the reduced failure rate $\frac{\mu_3 - \mu_1}{\mu_3} \cdot p_2$. The reasoning for other ranges of machine speeds is analogous and is treated in detail in the remainder of the paper.

1.3 Related Research

While there is a rich body of literature on the analysis of flow lines with linear flow of material (see the review in [3] and the introduction in [9]), there is relatively little literature on non-linear patterns in the flow of material. Most of it focuses on assembly/disassembly systems, see, for example, [4, 6]. A comprehensive review of this literature is given in [10] where a slightly different model of a merge system with two buffers is analyzed through decomposition. However, this led to serious numerical problems and motivated the work that led to this paper. We are aware of only one other attempt to model a continuous material merge system which is reported in [2]. In that paper, the assumption is that material coming from Machines $M_1$ and $M_2$ is always processed in a constant ratio by Machine $M_3$. Due to this assumption, it is possible to reduce the problem to the (known) analysis of an assembly system. Instead of assuming constant ratios, we develop the merge system with a priority rule, and we do so for two reasons: Firstly, we want to have a variety of building blocks for the analysis of flow lines and networks in order to deal with the variety of real-world manufacturing systems. Secondly, if a merge system is operated in isolation, it is more attractive to use a priority rule as it leads to a higher production rate. If we demand fixed ratios, i.e. that for every
part coming from $M_1$ we take any specific number of parts coming from $M_2$, a failure of any of the two machines leads to starvation of $M_3$. In our merge model, however, $M_1$ and $M_2$ can truly replace each other in cases of failures, and this is a useful and realistic feature. While the model in [2] is analyzed through an approximate decomposition, we develop an exact solution by extending the two-machine model in [9, p.112-131].

2 Derivation of Transition Equations

In order to determine performance measures such as the expected production rate or the expected inventory level for the model described above, we compute the probabilities of the possible system states. For system states $(x, \alpha_1, \alpha_2, \alpha_3)$ with full $(x = N)$ or empty $(x = 0)$ buffer, there are discrete probability masses $p(x, \alpha_1, \alpha_2, \alpha_3)$. Some of these boundary states have zero probability, however.

System states with a buffer that is neither full nor empty ($0 < x < N$) are called internal states. Since the buffer level is a continuous quantity, we use probability density functions $f(x, \alpha_1, \alpha_2, \alpha_3)$ to describe the random behavior of the system for this range of states. The analysis of these intermediate storage levels leads to a set of differential equations that are coupled.

The equations for boundary and internal states and the condition that all probabilities sum up to one lead to a system of equations that can be solved simultaneously to compute steady-state probabilities and density functions which in turn allow us to determine the above mentioned performance measures, i.e. average production rates and the inventory level.

2.1 Intermediate Storage Levels

The derivation of the transition equations for the intermediate storage levels is lengthy. For this reason, we describe only one derivation in detail and simply state the remaining equations.

Assume that all machines are up at time $t + \delta t$ and that the buffer level is between $x$ and $x + \delta x$. The probability of seeing the system in this state is

$$f(x, 1, 1, t + \delta t)\delta x.$$

It is equal to the sum of the probabilities of all previous states at time $t$ times the probabilities of the respective transitions. In a continuous time Markov process model, we need to consider explicitly only those transitions with at most one change of a machine state since all other transitions of second and higher order have zero probability as $\delta t$ approaches zero.
The four relevant (previous) states at time $t$, their probabilities, and the respective transition probabilities are:

- Machines $M_1$, $M_2$, and $M_3$ are up and the buffer level is between $x + (\mu_3 - \mu_1 - \mu_2)\delta t$ and $x + (\mu_3 - \mu_1 - \mu_2)\delta t + \delta x$. The probability of this state is

$$f(x + (\mu_3 - \mu_1 - \mu_2)\delta t, 1, 1, t) \delta x.$$  

In this case, none of the machines failed between time $t$ and $t + \delta t$. During this time interval, the buffer level decreased by the amount $\mu_3 \delta t$ that Machine $M_3$ processed and increased by the amount processed by $M_1$ and $M_2$. The net decrease between time $t$ and $t + \delta t$ is therefore $(\mu_3 - \mu_1 - \mu_2)\delta t$. The transition probability that all machines are up at time $t + \delta t$, given that all are up at time $t$, is

$$(1 - p_1\delta t)(1 - p_2\delta t)(1 - p_3\delta t) + o(\delta t) = 1 - (p_1 + p_2 + p_3)\delta t + o(\delta t)$$

since machine failures and repairs are assumed to be independent. This is the probability that none of the machines fails. All second and higher order terms are collected in $o(\delta t)$. Therefore

$$(1 - (p_1 + p_2 + p_3)\delta t + o(\delta t)) \cdot f(x + (\mu_3 - \mu_1 - \mu_2)\delta t, 1, 1, t) \delta x$$

is the joint probability that all machines are up both at time $t$ and at time $t + \delta t$.

- Machines $M_2$ and $M_3$ are up, $M_1$ is down, and the buffer level is between $x + (\mu_3 - \mu_2)\delta t$ and $x + (\mu_3 - \mu_2)\delta t + \delta x$. The probability of this state is

$$f(x + (\mu_3 - \mu_2)\delta t, 0, 1, 1, t) \delta x.$$  

Machine $M_1$ must have been repaired between time $t$ and $t + \delta t$ and the other machines must not have failed. The probability of this transition is

$$r_1\delta t(1 - p_2\delta t)(1 - p_3\delta t) + o(\delta t) = r_1\delta t + o(\delta t)$$

and the joint probability is therefore:

$$(r_1\delta t + o(\delta t)) \cdot f(x + (\mu_3 - \mu_2)\delta t, 0, 1, 1, t) \delta x$$
• Machines $M_1$ and $M_3$ are up, $M_2$ is down, and the buffer level is between $x + (\mu_3 - \mu_1)\delta t$ and $x + (\mu_3 - \mu_1)\delta t + \delta x$. The probability of this state is

$$f(x + (\mu_3 - \mu_1)\delta t, 1, 0, 1, t)\delta x.$$ 

Machine $M_2$ must have been repaired between time $t$ and $t + \delta t$ and the other machines must not have failed. The probability of this transition is

$$(1 - p_1\delta t)r_2\delta t(1 - p_3\delta t) + o(\delta t) = r_2\delta t + o(\delta t)$$

and the joint probability is therefore:

$$(r_2\delta t + o(\delta t)) \cdot f(x + (\mu_3 - \mu_1)\delta t, 1, 0, 1, t)\delta x$$

• Machines $M_1$ and $M_2$ are up, $M_3$ is down, and the buffer level is between $x - (\mu_1 + \mu_2)\delta t$ and $x - (\mu_1 + \mu_2)\delta t + \delta x$. The probability of this state is

$$f(x - (\mu_1 + \mu_2)\delta t, 1, 1, 0, t)\delta x.$$ 

Machine $M_3$ must have been repaired between time $t$ and $t + \delta t$ and the other machines must not have failed. The probability of this transition is

$$(1 - p_1\delta t)(1 - p_2\delta t)r_3\delta t + o(\delta t) = r_3\delta t + o(\delta t)$$

and the joint probability is therefore:

$$(r_3\delta t + o(\delta t)) \cdot f(x - (\mu_1 + \mu_2)\delta t, 1, 1, 0, t)\delta x$$

Assembling all these expressions, we have:

$$f(x, 1, 1, 1, t + \delta t)\delta x =$$

$$ (1 - (p_1 + p_2 + p_3)\delta t) \cdot f(x + (\mu_3 - \mu_1 - \mu_2)\delta t, 1, 1, 1, t)\delta x +$$

$$r_1\delta t f(x + (\mu_3 - \mu_2)\delta t, 0, 1, 1, t)\delta x +$$

$$r_2\delta t f(x + (\mu_3 - \mu_1)\delta t, 1, 0, 1, t)\delta x +$$

$$r_3\delta t f(x - (\mu_1 + \mu_2)\delta t, 1, 1, 0, t)\delta x + o(\delta t)$$

(1)
Using a Taylor series expansion of the probability density function \( f(x, 1, 1, t) \) at \( x \), we can write

\[
f(x + (\mu_3 - \mu_1 - \mu_2)\delta t, 1, 1, t)
= f(x, 1, 1, t) + (\mu_3 - \mu_1 - \mu_2)\delta t \frac{\partial}{\partial x} f(x, 1, 1, t) + o(\delta t). \tag{2}
\]

Using the same type of expansion for the other density functions on the right hand side of (1) and collecting all second and higher order terms in \( o(\delta t) \), we find

\[
f(x, 1, 1, t + \delta t)\delta x = (1 - (p_1 + p_2 + p_3)\delta t)f(x, 1, 1, t)\delta x +
(\mu_3 - \mu_1 - \mu_2)\delta t \frac{\partial}{\partial x} f(x, 1, 1, t)\delta x +
r_1\delta t f(x, 0, 1, 1, t)\delta x + r_2\delta t f(x, 1, 0, 1, t)\delta x +
r_3\delta t f(x, 1, 1, 0, t)\delta x + o(\delta t). \tag{3}
\]

Subtracting \( f(x, 1, 1, t)\delta x \) and dividing by \( \delta x \) and \( \delta t \), we find:

\[
\frac{f(x, 1, 1, t + \delta t) - f(x, 1, 1, t)}{\delta t} =
-(p_1 + p_2 + p_3)f(x, 1, 1, t) + (\mu_3 - \mu_1 - \mu_2)\frac{\partial}{\partial x} f(x, 1, 1, t) +
r_1 f(x, 0, 1, 1, t) + r_2 f(x, 1, 0, 1, t) + r_3 f(x, 1, 1, 0, t) + \frac{o(\delta t)}{\delta t}. \tag{4}
\]

If we now take the limit for \( \delta t \to 0 \), this leads to a partial differential equation in time \( t \) and buffer level \( x \). If we assume that the system reaches steady state where the probabilities of the states remain constant, we have

\[
\lim_{\delta t \to 0} \frac{f(x, 1, 1, t + \delta t) - f(x, 1, 1, t)}{\delta t} = 0
\]

and we can omit the time variable \( t \) in (4) to find:

\[
(p_1 + p_2 + p_3)f(x, 1, 1, 1) = (\mu_3 - \mu_1 - \mu_2)\frac{\partial}{\partial x} f(x, 1, 1, 1) +
r_1 f(x, 0, 1, 1) + r_2 f(x, 1, 0, 1) +
r_3 f(x, 1, 1, 0). \tag{5}
\]
Equation (5) is a differential equation in the buffer level $x$. The right side of the equation is related to the rate at which state $(x, 1, 1, 1)$ is reached and the left side is related to the rate at which it is left. In steady state, both rates have to be equal.

The derivation for the remaining internal equations is completely analogous. If we use the notation $f'(x, \alpha_1, \alpha_2, \alpha_3) = \frac{\partial}{\partial x} f(x, \alpha_1, \alpha_2, \alpha_3)$ and rearrange the terms slightly, we can state a complete set of $2^3 = 8$ differential equations where (5) corresponds to (11):

\[
\begin{align*}
\mu_2 f'(x, 0, 1, 0) &= -(r_1 + p_2 + r_3)f(x, 0, 1, 0) + p_1 f(x, 1, 1, 0) + r_2 f(x, 0, 0, 0) + p_3 f(x, 0, 1, 1) \quad (6) \\
(\mu_2 - \mu_3) f'(x, 0, 1, 1) &= -(r_1 + p_2 + p_3)f(x, 0, 1, 1) + p_1 f(x, 1, 1, 1) + r_2 f(x, 0, 0, 1) + r_3 f(x, 0, 1, 0) \quad (7) \\
\mu_1 f'(x, 1, 0, 0) &= -(p_1 + r_2 + r_3)f(x, 1, 0, 0) + r_1 f(x, 0, 0, 0) + p_2 f(x, 1, 1, 0) + p_3 f(x, 1, 0, 1) \quad (8) \\
(\mu_1 - \mu_3) f'(x, 1, 0, 1) &= -(p_1 + r_2 + p_3)f(x, 1, 0, 1) + r_1 f(x, 0, 0, 1) + p_2 f(x, 1, 1, 1) + r_3 f(x, 1, 0, 0) \quad (9) \\
(\mu_1 + \mu_2) f'(x, 1, 1, 0) &= -(p_1 + p_2 + r_3)f(x, 1, 1, 0) + r_1 f(x, 0, 1, 0) + p_2 f(x, 1, 0, 0) + p_3 f(x, 1, 1, 1) \quad (10) \\
(\mu_1 + \mu_2 - \mu_3) f'(x, 1, 1, 1) &= -(p_1 + p_2 + p_3)f(x, 1, 1, 1) + r_1 f(x, 0, 1, 1) + r_2 f(x, 1, 0, 1) + r_3 f(x, 1, 0, 0) \quad (11) \\
0 &= -(r_1 + r_2 + r_3)f(x, 0, 0, 0) + p_1 f(x, 1, 0, 0) + p_2 f(x, 0, 1, 0) + p_3 f(x, 0, 0, 1) \quad (12) \\
-\mu_3 f'(x, 0, 0, 1) &= -(r_1 + r_2 + p_3)f(x, 0, 0, 1) + p_1 f(x, 1, 0, 1) + p_2 f(x, 0, 1, 1) + r_3 f(x, 0, 0, 0) \quad (13)
\end{align*}
\]

Note that this is a set of coupled linear differential equations with constant coefficients. Adding up all these eight differential equations leads to

\[
\begin{align*}
\mu_2 f'(x, 0, 1, 0) + (\mu_2 - \mu_3) f'(x, 0, 1, 1) + \mu_1 f'(x, 1, 0, 0) + \\
(\mu_1 - \mu_3) f'(x, 1, 0, 1) + (\mu_1 + \mu_2) f'(x, 1, 1, 0) + \\
(\mu_1 + \mu_2 - \mu_3) f'(x, 1, 1, 1) - \mu_3 f'(x, 0, 0, 1) &= 0 \quad (14)
\end{align*}
\]

Integrating this equation with respect to $x$ and rearranging the result yields:

\[
\mu_1 (f(x, 1, 0, 0) + f(x, 1, 0, 1) + f(x, 1, 1, 0) + f(x, 1, 1, 1))
\]
\[ + \mu_2 (f(x,0,1,0) + f(x,0,1,1) + f(x,1,1,0) + f(x,1,1,1)) \]
\[ - \mu_3 (f(x,0,0,1) + f(x,0,1,1) + f(x,1,0,1) + f(x,1,1,1)) = K \quad (15) \]

The sum of the first two lines in Equation (15) is proportional to the rate at which the buffer level increases above a level \(x\) due to the production of Machines \(M_1\) and \(M_2\) while the remaining term on the left hand side term is proportional to the rate of decrease below \(x\) due to the production of Machine \(M_3\). If the system reaches steady state, there must be an upward crossing for every downward crossing for every \(x\), i.e. these two rates must be equal and we therefore have \(K = 0\). We can use (15) to replace (13). If we divide the differential equations by the coefficients of the respective left hand side, we find in matrix form the equation

\[
\begin{pmatrix}
f'(x,0,1,0) \\
f'(x,0,1,1) \\
f'(x,1,0,0) \\
f'(x,1,0,1) \\
f'(x,1,1,0) \\
f'(x,1,1,1) \\
0 \\
0
\end{pmatrix}
= \mathbf{A} \cdot \mathbf{F}(x) \quad (16)
\]

where the matrix \(\mathbf{A}\) is given in Figure 3 and where we have the vector of probability densities

\[
\mathbf{F}(x) = 
\begin{pmatrix}
f(x,0,1,0) \\
f(x,0,1,1) \\
f(x,1,0,0) \\
f(x,1,0,1) \\
f(x,1,1,0) \\
f(x,1,1,1) \\
f(x,0,0,0) \\
f(x,0,0,1)
\end{pmatrix}
\quad . \quad (17)
\]

If we have \(\mu_2 - \mu_3 \neq 0\), \(\mu_1 - \mu_3 \neq 0\), and \(\mu_1 + \mu_2 - \mu_3 \neq 0\), this is a set of six coupled differential equations that describes the structure of the probability density functions of internal states. If one or two of these inequalities do not hold, the order of the coupled set of differential equations reduces to five or four, but the solution principle remains unaffected.

In order to determine production rate and inventory level estimates, we will later determine closed-form expressions of this system of equations.
Figure 3: Content of matrix $A$

$$
A = \begin{pmatrix}
\frac{r_1 + p_2 + p_3}{\mu_2} & \frac{p_3}{\mu_2} & 0 & 0 & \frac{p_1}{\mu_2} & 0 & \frac{r_3}{\mu_2} & 0 \\
\frac{r_3}{\mu_1 - \mu_3} & -\frac{r_1 + p_2 + p_3}{\mu_2 - \mu_3} & 0 & 0 & 0 & \frac{p_1}{\mu_2 - \mu_3} & 0 & \frac{r_3}{\mu_2 - \mu_3} \\
0 & 0 & \frac{r_3}{\mu_1 - \mu_3} & -\frac{p_1 + p_2 + p_3}{\mu_1 - \mu_3} & 0 & \frac{p_3}{\mu_1 - \mu_3} & 0 & \frac{r_1}{\mu_1 - \mu_3} \\
\frac{r_1}{\mu_1 + \mu_2} & 0 & \frac{r_2}{\mu_1 + \mu_2} & 0 & -\frac{p_1 + p_2 + p_3}{\mu_1 + \mu_2} & \frac{p_3}{\mu_1 + \mu_2} & 0 & 0 \\
0 & \frac{r_1}{\mu_1 + \mu_3} & 0 & \frac{r_2}{\mu_1 + \mu_3} & \frac{r_3}{\mu_1 + \mu_3} & -\frac{p_1 + p_2 + p_3}{\mu_1 + \mu_2 - \mu_3} & 0 & 0 \\
p_2 & 0 & p_1 & 0 & 0 & 0 & -(r_1 + r_2 + r_3) & p_3 \\
\mu_2 & \mu_2 - \mu_3 & \mu_1 & \mu_1 - \mu_3 & \mu_1 + \mu_2 & \mu_1 + \mu_2 - \mu_3 & 0 & -\mu_3
\end{pmatrix}
$$
2.2 Boundary Storage Levels with Empty or Full Buffer

We use probability masses $p(x, \alpha_1, \alpha_2, \alpha_3)$ with $x = 0$ or $x = N$ to denote the probability of seeing the system in a state with empty or full buffer. There are 16 of these boundary states that need to be analyzed. Some of the boundary states have zero probability. An example is state $(0, 1, 0, 0)$: If Machine $M_1$ is up, the buffer becomes non-empty immediately and therefore the probability of seeing the system in this state is zero. The reasoning for several other states is similar and it can easily be shown that

$$p(0, 0, 0, 0) = p(0, 0, 1, 0) = p(0, 1, 0, 0) = p(0, 1, 1, 0) = 0,$$

$$p(N, 0, 0, 0) = p(N, 0, 0, 1) = 0 \quad (18)$$

Other states always have non-zero probabilities. Consider state $(0, 0, 0, 1)$. For all combinations of machine parameters it is perfectly possible that both Machine $M_1$ and $M_2$ are down, the buffer becomes empty, and the system remains in this state, at least until one machine is repaired. By a similar reasoning the following can be shown:

$$p(0, 0, 0, 1) > 0 \quad (19)$$
$$p(N, 0, 1, 0) > 0 \quad (20)$$
$$p(N, 1, 0, 0) > 0 \quad (21)$$
$$p(N, 1, 1, 0) > 0 \quad (22)$$

This accounts for 10 of the 16 boundary states. The situation for the remaining six boundary states is more complicated as we have to take the processing times into account. Assume, for example, that the combined processing rate of Machines $M_1$ and $M_2$ is not above the rate of $M_3$, i.e. $\mu_1 + \mu_2 \leq \mu_3$. In this case it is possible that all three machines are up while the buffer is empty. The only reason for the buffer to become non-empty is a failure of Machine $M_3$ and we therefore have $p(0, 1, 1, 1) \neq 0$ in this case. If, however, $M_3$ is slower, i.e. we have $\mu_1 + \mu_2 > \mu_3$, then the buffer becomes non-empty immediately and we have $p(0, 1, 1, 1) = 0$. Table 1 shows which states have zero or non-zero probability, depending on the processing rates.

In order to determine the probabilities of the boundary states, we again relate probabilities at time $t + \delta t$ to probabilities of possible previous states at time $t$ and the probabilities of the respective transition. Consider, for example, the probability $p(0, 0, 0, 1)$ of Machines $M_1$ and $M_2$ being down and $M_3$ being up while the buffer is empty. If this state is observed at time $t + \delta t$, the system may have been in four possible states at time $t$:

- Machines $M_1$ and $M_2$ were down and the buffer was empty and Machine $M_3$ was up and idle such that it cannot fail. The probability of this state at time $t$ is
<table>
<thead>
<tr>
<th>Case</th>
<th>Processing rates</th>
<th>$p(0,0,1,1)$</th>
<th>$p(0,1,0,1)$</th>
<th>$p(0,1,1,1)$</th>
<th>$p(\infty,0,1,1)$</th>
<th>$p(\infty,1,0,1)$</th>
<th>$p(\infty,1,1,1)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$\mu_1 + \mu_2 &lt; \mu_3$</td>
<td>n</td>
<td>n</td>
<td>n</td>
<td>z</td>
<td>z</td>
<td>z</td>
</tr>
<tr>
<td>2</td>
<td>$\mu_1 + \mu_2 = \mu_3$</td>
<td>n</td>
<td>n</td>
<td>n</td>
<td>z</td>
<td>z</td>
<td>n</td>
</tr>
<tr>
<td></td>
<td>$\mu_1 + \mu_2 &gt; \mu_3$.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>$\mu_1 &lt; \mu_3, \mu_2 &lt; \mu_3$</td>
<td>n</td>
<td>n</td>
<td>z</td>
<td>z</td>
<td>z</td>
<td>n</td>
</tr>
<tr>
<td>4</td>
<td>$\mu_1 &lt; \mu_3, \mu_2 = \mu_3$</td>
<td>n</td>
<td>n</td>
<td>z</td>
<td>n</td>
<td>z</td>
<td>n</td>
</tr>
<tr>
<td>5</td>
<td>$\mu_1 = \mu_3, \mu_2 &lt; \mu_3$</td>
<td>n</td>
<td>n</td>
<td>z</td>
<td>z</td>
<td>n</td>
<td>n</td>
</tr>
<tr>
<td>6</td>
<td>$\mu_1 = \mu_3, \mu_2 = \mu_3$</td>
<td>n</td>
<td>n</td>
<td>z</td>
<td>n</td>
<td>n</td>
<td>n</td>
</tr>
<tr>
<td>7</td>
<td>$\mu_1 &gt; \mu_3, \mu_2 &lt; \mu_3$</td>
<td>n</td>
<td>z</td>
<td>z</td>
<td>z</td>
<td>n</td>
<td>n</td>
</tr>
<tr>
<td>8</td>
<td>$\mu_1 &gt; \mu_3, \mu_2 = \mu_3$</td>
<td>n</td>
<td>z</td>
<td>z</td>
<td>n</td>
<td>n</td>
<td>n</td>
</tr>
<tr>
<td>9</td>
<td>$\mu_1 &lt; \mu_3, \mu_2 &gt; \mu_3$</td>
<td>z</td>
<td>n</td>
<td>z</td>
<td>n</td>
<td>n</td>
<td>n</td>
</tr>
<tr>
<td>10</td>
<td>$\mu_1 = \mu_3, \mu_2 &gt; \mu_3$</td>
<td>z</td>
<td>n</td>
<td>z</td>
<td>n</td>
<td>n</td>
<td>n</td>
</tr>
<tr>
<td>11</td>
<td>$\mu_1 &gt; \mu_3, \mu_2 &gt; \mu_3$</td>
<td>z</td>
<td>z</td>
<td>z</td>
<td>n</td>
<td>n</td>
<td>n</td>
</tr>
</tbody>
</table>
The buffer level was between 0 and \( \mu_3 \delta t \). The probability that the system remains in this state is essentially the probability that neither \( M_1 \) nor \( M_2 \) is repaired:

\[
(1 - r_1 \delta t)(1 - r_2 \delta t) + o(\delta t) = 1 - (r_1 + r_2) \delta t + o(\delta t) \tag{23}
\]

- The buffer was empty and Machines \( M_1 \) and \( M_2 \) were up and \( M_2 \) was down at time \( t \), with probability \( p(0, 1, 0, 1, t) \). Note that \( p(0, 1, 0, 1, t) > 0 \) only if \( \mu_1 \leq \mu_3 \). Otherwise the system cannot remain in this states as the buffer fills immediately. The probability of the transition is essentially the probability of a failure of Machine \( M_1 \):

\[
p_1 \delta t (1 - r_2 \delta t)(1 - p_3 \delta t) + o(\delta t) = p_1 \delta t + o(\delta t) \tag{24}
\]

- The buffer was empty and Machines \( M_2 \) and \( M_3 \) were up and \( M_1 \) was down at time \( t \), with probability \( p(0, 0, 1, 1, t) \). Note that \( p(0, 0, 1, 1, t) > 0 \) only if \( \mu_2 \leq \mu_3 \). The transition probability is essentially the probability of a failure of Machine \( M_2 \):

\[
(1 - r_1 \delta t)p_2 \delta t (1 - p_3 \delta t) + o(\delta t) = p_2 \delta t + o(\delta t) \tag{25}
\]

- The buffer level was between 0 and \( \mu_3 \delta t \), Machines \( M_1 \) and \( M_2 \) were down and Machine \( M_3 \) was up at time \( t \). The probability is

\[
\int_0^{\mu_3 \delta t} f(x, 0, 0, 1, t)dx = F(\mu_3 \delta t, 0, 0, 1, t) - F(0,0,0,1,t) \tag{26}
\]

if we define \( F(x, 0, 0, 1, t) \) such that \( \frac{\partial}{\partial x} F(x, 0, 0, 1, t) = f(x, 0, 0, 1, t) \). A Taylor series expansion of \( F(x, 0, 0, 1, t) \) at \( x = 0 \) yields

\[
F(\mu_3 \delta t, 0, 0, 1, t) = F(0,0,0,1,t) + \frac{\partial}{\partial x} F(x, 0, 0, 1, t) \frac{(\mu_3 \delta t - 0)^1}{1!} + o(\delta t) = F(0,0,0,1,t) + \mu_3 \delta tf(x,0,0,1,t) + o(\delta t) \tag{27}
\]

and inserting this result in (26) we find

\[
\int_0^{\mu_3 \delta t} f(x, 0, 0, 1, t)dx = \mu_3 \delta tf(0,0,0,1,t) + o(\delta t). \tag{28}
\]

Between times \( t \) and \( t + \delta t \) no machine fails or gets repaired. The transition probability is

\[
(1 - r_1 \delta t)(1 - r_2 \delta t)(1 - p_3 \delta t) + o(\delta t) = 1 - (r_1 + r_2 + p_3) \delta t + o(\delta t) \tag{29}
\]
If we assemble all these results, we get:

\[
p(0,0,0,1,t + \delta t) = (1 - (r_1 + r_2)\delta t)p(0,0,0,1,t) + p_1\delta tp(0,1,0,1,t) + p_2\delta tp(0,0,1,1,t) + \mu_3\delta tf(0,0,0,1,t) + o(\delta t). \tag{30}
\]

Rearranging this equation and dividing by \(\delta t\) we find:

\[
\frac{p(0,0,0,1,t + \delta t) - p(0,0,0,1,t)}{\delta t} = -(r_1 + r_2)p(0,0,0,1,t) + p_1p(0,1,0,1,t) + p_2p(0,0,1,1,t) + \mu_3f(0,0,0,1,t) + o(\delta t)\frac{1}{\delta t}. \tag{31}
\]

If we take the limit \(\delta t \rightarrow 0\) and assume that the system is in steady state, the time derivative vanishes and we can omit the time index to conclude:

\[
(r_1 + r_2)p(0,0,0,1) = p_1p(0,1,0,1) + p_2p(0,0,1,1) + \mu_3f(0,0,0,1) \tag{32}
\]

The left hand side of (32) is the rate at which the system leaves state \((0,0,0,1)\) and the right hand side is the rate at which this state is entered.

Note that an Equation like (32) relates probability masses like \(p(0,0,0,1)\) to probability density functions like \(f(x,0,0,0,1)\). These relations are needed as boundary conditions when we solve the system of differential equations developed in Section 2.1 that describe probability density functions of intermediate storage levels.

Equation (32) holds for all 11 cases in Table 1. However, in several of these cases \(p(0,0,0,1)\) and/or \(p(0,1,0,1)\) are zero as the respective state is left immediately because Machines \(M_1\) and/or \(M_2\) are faster than \(M_3\).

In order to state the remaining equations in a compact form, we first define an indicator function

\[
I(x) = \begin{cases} 
1, & x > 0 \\
0, & x \leq 0 
\end{cases} \tag{33}
\]

Using this indicator function, the complete set of boundary equations is given in Table 2. Since some of the states can have both zero and non-zero probability, depending on the processing rates, not all of the equations apply to all of the cases in Table 1. Table 3 shows which of the boundary equations are used to solve which case.
Table 2: Boundary equations

\[
(r_1 + r_2)p(0, 0, 0, 1) = p_1 p(0, 1, 0, 1) + p_2 p(0, 0, 1, 1) + \mu_3 f(0, 0, 0, 1) \tag{34}
\]

\[
(r_1 + p_2 + \frac{\mu_2}{\mu_3} p_3) p(0, 0, 1, 1) = p_1 p(0, 1, 1, 1) + r_2 p(0, 0, 0, 1) + (\mu_3 - \mu_2) f(0, 0, 0, 1) I(\mu_3 - \mu_2) \tag{35}
\]

\[
(p_1 + r_2 + \frac{\mu_1}{\mu_3} p_3) p(0, 1, 0, 1) = r_1 p(0, 0, 0, 1) + p_2 p(1, 1, 1) + (\mu_3 - \mu_1) f(0, 1, 0, 1) I(\mu_3 - \mu_1) \tag{36}
\]

\[
(p_1 + p_2 + \frac{\mu_1 + \mu_2}{\mu_3} p_3) p(0, 1, 1, 1) = r_1 p(0, 0, 1, 1) + r_2 p(1, 0, 1, 1) + (\mu_3 - \mu_1 - \mu_2) f(0, 1, 1, 1) I(\mu_3 - \mu_1 - \mu_2) \tag{37}
\]

\[
(r_1 + r_3) p(N, 0, 0, 1) = p_3 p(N, 0, 0, 1) \tag{38}
\]

\[
(r_1 + \frac{\mu_3}{\mu_2} p_2 + p_3) p(N, 0, 1, 1) = \min \left\{ \frac{\mu_3}{\mu_1}, 1 \right\} p_1 p(N, 1, 1, 1) + r_3 p(N, 0, 1, 0) + (\mu_2 - \mu_3) f(N, 0, 1, 1) I(\mu_2 - \mu_3) \tag{39}
\]

\[
(r_2 + r_3) p(N, 1, 0, 0) = p_3 p(N, 1, 0, 1) + \mu_3 f(N, 1, 0, 1) \tag{40}
\]

\[
(\frac{\mu_3}{\mu_1} p_1 + r_2 + p_3) p(N, 1, 0, 1) = r_3 p(N, 1, 0, 0) + (\mu_1 - \mu_3) f(N, 1, 0, 1) I(\mu_1 - \mu_3) \tag{41}
\]

\[
r_3 p(N, 1, 1, 0) = r_1 p(N, 0, 1, 0) + r_2 p(N, 1, 0, 0) + p_3 p(N, 1, 1, 1) + (\mu_1 + \mu_2) f(N, 1, 1, 0) \tag{42}
\]

\[
(\min \left\{ \frac{\mu_3}{\mu_1}, 1 \right\} p_1 + \max \left\{ \frac{\mu_3 - \mu_1}{\mu_2}, 0 \right\} p_2 + p_3) p(N, 1, 1, 1) = r_3 p(N, 1, 1, 0) + r_3 p(N, 1, 1, 0) + (\mu_1 + \mu_2) f(N, 1, 1, 1) I(\mu_1 + \mu_2 - \mu_3) \tag{43}
\]
Table 3: Cases and boundary equations

<table>
<thead>
<tr>
<th>Cases</th>
<th>Valid Equations</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(34), (35), (36), (37), (38), (40), (42)</td>
</tr>
<tr>
<td>2</td>
<td>(34), (35), (36), (37), (38), (40), (42), (43)</td>
</tr>
<tr>
<td>3</td>
<td>(34), (35), (36), (38), (40), (42), (43)</td>
</tr>
<tr>
<td>4</td>
<td>(34), (35), (36), (38), (39), (40), (42), (43)</td>
</tr>
<tr>
<td>5</td>
<td>(34), (35), (36), (38), (40), (41), (42), (43)</td>
</tr>
<tr>
<td>6</td>
<td>(34), (35), (36), (38), (39), (40), (41), (42), (43)</td>
</tr>
<tr>
<td>7</td>
<td>(34), (35), (38), (40), (41), (42), (43)</td>
</tr>
<tr>
<td>8</td>
<td>(34), (35), (38), (39), (40), (41), (42), (43)</td>
</tr>
<tr>
<td>9</td>
<td>(34), (36), (38), (39), (40), (42), (43)</td>
</tr>
<tr>
<td>10</td>
<td>(34), (36), (38), (39), (40), (41), (42), (43)</td>
</tr>
<tr>
<td>11</td>
<td>(34), (38), (39), (40), (41), (42), (43)</td>
</tr>
</tbody>
</table>

2.3 Boundary Storage Levels with Almost Empty or Full Buffer

A second set of equations that relates probability masses of boundary states to probability density functions of internal states can be derived if we analyze states where the buffer is almost empty or full at time \(t + \delta t\) and the processing rates are such that it must have been completely empty or full at time \(t\). Consider, for example, in Case 1 with \(\mu_1 + \mu_2 < \mu_3\) the situation that Machine \(M_1\) is up while Machines \(M_2\) and \(M_3\) are down at time \(t + \delta t\). We now ask for the probability of having a buffer level \(x\) between 0 and \(\mu_1 \delta t\), the amount of material Machine \(M_1\) can have filled into the buffer between time \(t\) and \(t + \delta t\). This probability is given by

\[
\int_0^{\mu_1 \delta t} f(x, 1, 0, 0, t + \delta t) dx. \tag{44}
\]

The only possible (previous) boundary state is \((0, 1, 0, 1, t)\), with probability \(p(0, 1, 0, 1, t)\), and there must have been a failure of Machine \(M_3\). The probability of this failure during a time interval of length \(\delta t\) is

\[
\frac{\mu_1}{\mu_3} \delta t. \tag{45}
\]

Note that the probability of state \((0, 1, 0, 1, t)\) is non-zero if and only if \(\mu_3 \leq \mu_3\), i.e. if Machine \(M_3\) can be partially starved and the failure rate in this state has to be reduced by a factor \(\frac{\mu_1}{\mu_3}\). There is no other possible state at time \(t\) with non-zero probability or a transition of at most first order. Furthermore, we do *not* have to take internal states with
\( \mu_1 \delta t < x \) into account. These would only add second or higher order terms. For this reason, we have

\[
\int_0^{\mu_1 \delta t} f(x, 1, 0, 0, t + \delta t) dx = \frac{\mu_1}{\mu_3} p_3 \delta t p(0, 1, 0, 1, t).
\] (46)

Applying the same reasoning as in the derivation of Equation (28), we find

\[
\mu_1 \delta t f(0, 1, 0, 0, t + \delta t) = \frac{\mu_1}{\mu_3} p_3 \delta t p(0, 1, 0, 1, t).
\] (47)

If we assume that the system is in steady state and hence omit the time indicator \( t \) and if we multiply by \( \mu_3 \), we eventually get:

\[
\mu_3 f(0, 1, 0, 0) = p_3 p(0, 1, 0, 1)
\] (48)

for all cases with \( \mu_1 \leq \mu_3 \). Now consider cases where Machine \( M_1 \) is faster that \( M_3 \), i.e. \( \mu_1 > \mu_3 \). There is no possible state at time \( t \) that requires a transition of at most first order. It is, for example, possible that the previous state was \((0, 0, 0, 1, t)\). In order for the buffer level to increase, we need a repair of \( M_1 \), with probability \( r_1 \delta t \). However, we also need a failure of \( M_3 \), with probability \( p_3 \delta t \). This leads to second order terms like \( r_1 p_3 (\delta t)^2 \) that can be omitted. In all these cases with \( \mu_1 > \mu_3 \) we therefore have

\[
\int_0^{\mu_1 \delta t} f(x, 1, 0, 0, t + \delta t) dx = 0
\] (49)

and this is included in (48) as \( p(0, 1, 0, 1) = 0 \) if \( \mu_1 > \mu_3 \). We conclude that (48) holds in all cases. Proceeding in the same way, we can derive the following second set of equations for near-boundary states:

\[
\begin{align*}
\mu_3 f(0, 0, 1, 0) & = p_3 p(0, 0, 1, 1) \\
\mu_3 f(0, 1, 0, 0) & = p_3 p(0, 1, 0, 1) \\
\mu_3 f(0, 1, 1, 0) & = p_3 p(0, 1, 1, 1) \\
(\mu_1 - \mu_3) f(0, 1, 0, 1) & = r_1 p(0, 0, 0, 1) \\
(\mu_2 - \mu_3) f(0, 0, 1, 1) & = r_2 p(0, 0, 0, 1) \\
(\mu_1 + \mu_2 - \mu_3) f(0, 1, 1, 1) & = r_1 p(0, 0, 1, 1) + r_2 p(0, 1, 0, 1) \\
\mu_3 f(N, 0, 0, 1) & = \frac{\mu_3}{\mu_1} p_1 p(N, 0, 1, 1) + \frac{\mu_3}{\mu_2} p_2 p(N, 0, 1, 1) \\
(\mu_3 - \mu_2) f(N, 0, 1, 1) & = \min\left\{\frac{\mu_3}{\mu_1}, 1\right\} p_1 p(N, 1, 1, 1) + r_3 p(N, 0, 1, 0) \\
(\mu_3 - \mu_1) f(N, 1, 0, 1) & = \frac{\mu_3 - \mu_1}{\mu_2} p_2 p(N, 1, 1, 1) + r_3 p(N, 1, 0, 0) \\
(\mu_3 - \mu_1 - \mu_2) f(N, 1, 0, 1) & = r_3 p(N, 1, 1, 0)
\end{align*}
\]
Table 4: Cases and near-boundary equations

<table>
<thead>
<tr>
<th>Cases</th>
<th>Valid Equations</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(50), (51), (52), (57), (58), (59)</td>
</tr>
<tr>
<td>2</td>
<td>(50), (51), (52), (57), (58)</td>
</tr>
<tr>
<td>3</td>
<td>(50), (51), (52), (55), (56), (57), (58)</td>
</tr>
<tr>
<td>4</td>
<td>(50), (51), (52), (55), (56), (57)</td>
</tr>
<tr>
<td>5</td>
<td>(50), (51), (52), (55), (56), (57)</td>
</tr>
<tr>
<td>6</td>
<td>(50), (51), (52), (55), (56)</td>
</tr>
<tr>
<td>7</td>
<td>(50), (51), (53), (52), (55), (56), (57)</td>
</tr>
<tr>
<td>8</td>
<td>(50), (51), (53), (52), (55), (56)</td>
</tr>
<tr>
<td>9</td>
<td>(50), (54), (51), (52), (55), (56), (58)</td>
</tr>
<tr>
<td>10</td>
<td>(50), (54), (51), (52), (55), (56)</td>
</tr>
<tr>
<td>11</td>
<td>(50), (51), (54), (53), (52), (55), (56)</td>
</tr>
</tbody>
</table>

Table 4 shows which of the equations hold in which case. (The equations that we actually used in our algorithm—to be described below—are underlined.)

2.4 Solution of the Transition Equations

In order to determine production rate and inventory level estimates, we first have to determine probability masses \( p_{\mathcal{B}_4} \) of internal states. This solution is general as it contains up to six coefficients \( \alpha, \beta \) that need to determined later on.

2. Determine expressions for the probabilities masses \( p(x, \alpha_1, \alpha_2, \alpha_3), x \in \{0, N\} \), of the boundary states in terms of the probability density functions by applying the equations for boundary states.

3. Solve a set of up to six equations in order to determine the coefficients \( \alpha_i \) by applying the near-boundary equations and the condition that all probabilities sum up to one. This leads to unique expressions for the probabilities.
4. Use the probabilities to eventually determine the performance measures.

The remainder of this subsection describes these steps in detail for the relatively complicated Case 3 in Table 1. Several of the other cases are slightly easier to deal with as the number of differential equations for internal states decreases to five or even four. Consider, for example, Case 2 in Table 1 where we have $\mu_1 + \mu_2 = \mu_3$. In this case the left side of Equation (11) vanishes and the second to last row of the matrix $A$ in Figure 3 has to be adjusted accordingly.

### 2.4.1 Solution of the Internal Equations

In this section we describe how to solve the set of differential equations (16) for Case 3. Define

$$F_1(x) = \begin{pmatrix} f(x, 0, 1, 0) \\ f(x, 0, 1, 1) \\ f(x, 0, 0, 0) \\ f(x, 1, 0, 1) \\ f(x, 1, 1, 0) \\ f(x, 1, 1, 1) \end{pmatrix}$$

$$F_2(x) = \begin{pmatrix} f(x, 0, 0, 0) \\ f(x, 0, 0, 1) \end{pmatrix}$$

and the matrices

$$A_{11} = \begin{pmatrix} \frac{-r_1 + p_2 + p_3}{\mu_2} & p_3 & 0 & 0 & \frac{p_1}{\mu_2} & 0 \\ \frac{-r_2}{\mu_2 - \mu_3} & \frac{-r_1 + p_2 + p_3}{\mu_2 - \mu_3} & 0 & 0 & 0 & \frac{p_1}{\mu_2 - \mu_3} \\ 0 & 0 & \frac{p_1 + r_2 + p_3}{\mu_1} & p_3 & \frac{p_2}{\mu_1} & 0 \\ 0 & 0 & \frac{r_3}{\mu_1 + \mu_3} & \frac{-p_1 + r_2 + p_3}{\mu_1 - \mu_3} & 0 & \frac{p_2}{\mu_1 - \mu_3} \\ \frac{r_1}{\mu_1 + \mu_2} & 0 & \frac{r_3}{\mu_1 + \mu_2} & 0 & \frac{-p_1 + r_2 + p_3}{\mu_1 + \mu_2} & \frac{p_3}{\mu_1 + \mu_2} \\ 0 & \frac{r_1}{\mu_1 + \mu_2 - \mu_3} & 0 & \frac{-p_1 + r_2 + p_3}{\mu_1 + \mu_2 - \mu_3} & \frac{r_3}{\mu_1 + \mu_2 - \mu_3} & \frac{-p_1 + p_2 + p_3}{\mu_1 + \mu_2 - \mu_3} \end{pmatrix}$$

20
as well as

\[
A_{21} = \begin{pmatrix}
\frac{r_2}{\mu_2} & 0 \\
0 & \frac{r_2}{\mu_2 - \mu_3} \\
\frac{r_1}{\mu_1} & 0 \\
0 & \frac{r_1}{\mu_1 - \mu_3} \\
0 & 0 \\
0 & 0
\end{pmatrix}
\]

\[
A_{22} = \begin{pmatrix}
-(r_1 + r_2 + r_3) & p_3 \\
0 & -\mu_3
\end{pmatrix}
\]

In order to write system (16) as:

\[
\begin{pmatrix}
F_1'(x) \\
0
\end{pmatrix} = \begin{pmatrix}
A_{11} & A_{12} \\
A_{21} & A_{22}
\end{pmatrix} \begin{pmatrix}
F_1(x) \\
F_2(x)
\end{pmatrix}
\]

(60)

A similar partitioning is possible in all the other cases as well if we adjust the matrices accordingly. If we solve the second equation of (60) for \(F_2(x)\), we find the linear matrix equation

\[
F_2(x) = -A_{22}^{-1}A_{21}F_1(x) = \Omega F_1(x).
\]

(61)

and, using (61) in the first equation of (60), the linear matrix differential equation

\[
F_1'(x) = (A_{11} - A_{12}A_{22}^{-1}A_{21})F_1(x) = \Lambda F_1(x).
\]

(62)

with

\[
\Omega = -A_{22}^{-1}A_{21}
\]

(63)

\[
\Lambda = A_{11} - A_{12}A_{22}^{-1}A_{21}
\]

(64)

In order to solve the set of equations \(F_1'(x) = \Lambda F_1(x)\) in (62), we have to determine the eigenvalues \(s_i\) and corresponding eigenvectors \(y_i^{(i)}\) of \(\Lambda\) that are defined as the solution to equation \((\Lambda - sI)y = 0\). These eigenvalues and eigenvectors have to be
computed numerically using standard procedures like those in [5]. In Cases 1, 3, 7, 9, and 11, there are up to six different eigenvalues. If we find six independent eigenvectors, we get the fundamental matrix:

\[
\Psi(x) = \begin{pmatrix}
  y_1^{[1]} \\
y_2^{[1]} \\
y_3^{[1]} \\
y_4^{[1]} \\
y_5^{[1]} \\
y_6^{[1]}
\end{pmatrix} e^{s_1 x} + \begin{pmatrix}
  y_1^{[2]} \\
y_2^{[2]} \\
y_3^{[2]} \\
y_4^{[2]} \\
y_5^{[2]} \\
y_6^{[2]}
\end{pmatrix} e^{s_2 x} + \ldots + \begin{pmatrix}
  y_1^{[6]} \\
y_2^{[6]} \\
y_3^{[6]} \\
y_4^{[6]} \\
y_5^{[6]} \\
y_6^{[6]}
\end{pmatrix} e^{s_6 x}
\]  

(65)

Defining a coefficient vector \( \mathbf{c}^T = (c_1, c_2, \ldots, c_6) \), we can write the general solution to (62) as

\[
\mathbf{F}_1(x) = \Psi(x) \mathbf{c}.
\]  

(66)

In Cases 1, 3, 7, 9, and 11, we find six equations:

\[
f(x, 0, 1, 0) = c_1 y_1^{[1]} \exp(s_1 x) + c_2 y_2^{[2]} \exp(s_2 x) + \ldots + c_6 y_6^{[6]} \exp(s_6 x) \]  

(67)

\[
f(x, 0, 1, 1) = c_1 y_1^{[1]} \exp(s_1 x) + c_2 y_2^{[2]} \exp(s_2 x) + \ldots + c_6 y_6^{[6]} \exp(s_6 x) \]  

(68)

\[
f(x, 1, 0, 0) = c_1 y_3^{[1]} \exp(s_1 x) + c_2 y_3^{[2]} \exp(s_2 x) + \ldots + c_6 y_6^{[6]} \exp(s_6 x) \]  

(69)

\[
f(x, 1, 0, 1) = c_1 y_4^{[1]} \exp(s_1 x) + c_2 y_4^{[2]} \exp(s_2 x) + \ldots + c_6 y_6^{[6]} \exp(s_6 x) \]  

(70)

\[
f(x, 1, 1, 0) = c_1 y_5^{[1]} \exp(s_1 x) + c_2 y_5^{[2]} \exp(s_2 x) + \ldots + c_6 y_6^{[6]} \exp(s_6 x) \]  

(71)

\[
f(x, 1, 1, 1) = c_1 y_6^{[1]} \exp(s_1 x) + c_2 y_6^{[2]} \exp(s_2 x) + \ldots + c_6 y_6^{[6]} \exp(s_6 x) \]  

(72)

In these cases, Equation (61) boils down to the following system:

\[
f(x, 0, 0, 0) = \omega_{11} f(x, 0, 1, 0) + \ldots + \omega_{16} f(x, 1, 1, 1)
\]

(73)

\[
f(x, 0, 0, 1) = \omega_{21} f(x, 0, 1, 0) + \ldots + \omega_{26} f(x, 1, 1, 1)
\]

(74)

which indirectly relates density functions to eigenvalues, eigenvectors and unknowns \( \xi \).

For any given matrix \( \mathbf{A} \) and corresponding eigenvalues and eigenvectors, these are six functions in \( x \) with unknowns \( c_i \). These equations are used in two ways: Firstly, they are applied to the boundary and near-boundary equations to determine numerical values for the coefficients \( c_i \) for any given set of system parameters. Secondly, for any given set of both system parameters and coefficients \( c_i \), they are used to compute production rate and inventory level estimates.
2.4.2 Determination of the Boundary Probabilities

Given the expressions for the probability density functions in (66), we now use equations (34) to (43) in order to determine expressions for the probabilities $p(x, \alpha_1, \alpha_2, \alpha_3)$ of boundary states. Note that these equations relate probability masses to densities $f(x, \alpha_1, \alpha_2, \alpha_3)$ for specific numerical values of $x$. In order to solve Case 3, we first use from Table 3 Equations (34), (35), and (36) for the empty buffer case and write

$$
\begin{pmatrix}
    p(0,0,0,1) \\
    p(0,0,1,1) \\
    p(0,1,0,1)
\end{pmatrix} = H_{3,0}^{-1} \cdot \begin{pmatrix}
    \mu_3 f(0,0,0,1) \\
    (\mu_3 - \mu_2) f(0,0,1,1) \\
    (\mu_3 - \mu_1) f(0,1,0,1)
\end{pmatrix}
$$

where we define

$$
H_{3,0} = \begin{pmatrix}
    r_1 + r_2 & -p_1 & 0 \\
    -r_2 & r_1 + p_2 + \frac{\mu_2}{\mu_3} p_3 & 0 \\
    -r_1 & 0 & p_1 + r_2 + \frac{\mu_3}{\mu_1} p_3
\end{pmatrix}.
$$

Equations (38), (40), (42), and (43) for the full buffer case yield

$$
\begin{pmatrix}
    p(N,0,1,0) \\
    p(N,1,0,0) \\
    p(N,1,1,0) \\
    p(N,1,1,1)
\end{pmatrix} = H_{3,N}^{-1} \cdot \begin{pmatrix}
    \mu_2 f(N,0,1,0) \\
    \mu_1 f(N,1,0,0) \\
    (\mu_1 + \mu_2) f(N,1,1,0) \\
    (\mu_1 + \mu_2 + \mu_3) f(N,1,1,1)
\end{pmatrix}
$$

where we define

$$
H_{3,N} = \begin{pmatrix}
    r_1 + r_3 & 0 & 0 & 0 \\
    0 & r_2 + r_3 & 0 & 0 \\
    -r_1 & -r_2 & r_3 & -p_3 \\
    0 & 0 & -r_3 & p_1 + \frac{\mu_1}{\mu_2} p_2 + p_3
\end{pmatrix}.
$$

We now have a procedure to compute expressions for the probabilities of boundary states. Note again that these expressions are not stated here explicitly as they are rather clumsy, but it is not difficult to handle them symbolically using a program like Mathematica.

2.4.3 Determination of Coefficients $c_i$

Even for a given set of system parameters and the corresponding eigenvalues and eigenvectors, all we have so far are equations for the probability masses and densities with the unknowns $c_i$. What needs to be done now is to determine numerical values for these
unknown coefficients. If we have, as in Case 3, six coefficients, we need six independent equations. The equations we derive in this section are linear in the six unknowns \( q \) and it is therefore straightforward to solve them.

One equation is used in all cases. It states that all the probabilities sum up to one, or formally:

\[
\sum_{\alpha_1=0}^{1} \sum_{\alpha_2=0}^{1} \sum_{\alpha_3=0}^{1} \int_{0}^{N} f(x, \alpha_1, \alpha_2, \alpha_3) \, dx + p(0, \alpha_1, \alpha_2, \alpha_3) + p(N, \alpha_1, \alpha_2, \alpha_3) = 1
\]

(79)

Since all expressions for probability masses \( p(x, \alpha_1, \alpha_2, \alpha_3) \) and all densities \( f(x, \alpha_1, \alpha_2, \alpha_3) \) are linear in the unknowns \( \alpha_i \), and since summation and integration are linear operations, Equation (79) is also linear in \( \alpha_i \). In order to determine six unknowns \( \alpha_i \) in Case 3, we need five additional equations. For this purpose we use the near-boundary equations (50) to (59) to establish the remaining conditions. From Table 4 we know that seven equations hold in Case 3. Out of these we chose (50), (51), (55), (57), and (58), i.e. we demand that

\[
0 = \mu_3 f(0, 0, 1, 0) - p_3 p(0, 0, 1, 1) \quad (80)
\]
\[
0 = \mu_3 f(0, 1, 0, 0) - p_3 p(0, 1, 0, 1) \quad (81)
\]
\[
0 = (\mu_1 + \mu_2 - \mu_3) f(0, 1, 1, 1) - r_1 p(0, 0, 1, 1) - r_2 p(0, 1, 0, 1) \quad (82)
\]
\[
0 = (\mu_3 - \mu_2) f(N, 0, 1, 1) - p_3 p(N, 1, 1, 1) - r_3 p(N, 0, 1, 0) \quad (83)
\]
\[
0 = (\mu_3 - \mu_1) f(N, 1, 0, 1) - \frac{\mu_3 - \mu_1}{\mu_2} p_2 p(N, 1, 1, 1) - r_3 p(N, 1, 0, 0) \quad (84)
\]

Note that these equations are also linear in the unknowns \( \alpha_i \), i.e. we can write equations (79) to (84) as

\[
B \cdot \begin{pmatrix} c_1 \\ c_2 \\ \vdots \\ c_6 \end{pmatrix} = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix}.
\]

(85)

where the matrix \( B \) contains symbolic expressions for the coefficients of the unknowns \( \alpha_i \) that we get by inserting the symbolic expressions for probability masses and densities. If we solve this linear system of equations for given numerical values of the system parameters and eigenvalues and eigenvectors of \( A \), we get the desired numerical values of the probability masses and densities. These can then be used to determine performance measures as described in the next section. The underlined entries in Table 4
indicate those equations that we used in each of the cases to establish the required set of equations. Note that the number of required equations is case-specific. In Case 6, for example, the set of differential equations for internal states has only four degrees of freedom and we need only four equations (including (79)) to determine the four unknowns $c_i, i = 1, \ldots, 4$.

2.4.4 Determination of Performance Measures

For given probability masses and density functions, we define the average inventory level $\bar{x}$ as

$$
\bar{x} = \sum_{\alpha_1=0}^1 \sum_{\alpha_2=0}^1 \sum_{\alpha_3=0}^1 \int_0^N x \cdot f(x, \alpha_1, \alpha_2, \alpha_3) \, dx + N \cdot p(N, \alpha_1, \alpha_2, \alpha_3)
$$

(86)

The production rate can be calculated along several paths. Firstly, we can ask for the rate at which material leaves the system in the possible states and evaluate the expression

$$
PR = \mu_3 \int_0^N (f(x,0,1,0) + f(x,1,0,1) + f(x,0,0,1) + f(x,1,1,1)) \, dx
$$

$$
+ \mu_3 (p(0,1,0,1) + p(0,0,1,1) + p(0,1,1,1))
$$

$$
+ \mu_1 p(0,0,1,1) + \mu_2 p(0,1,0,1) + (\mu_1 + \mu_2) p(0,1,1,1)
$$

(87)

However, it is also possible to determine the production rates related to the separate machines. If $p_k$ and $r_i$ are the rates of failures and repairs, respectively, Machine $M_i$ is operational a fraction of time that equals

$$
e_i = \frac{r_i}{r_i + p_i}
$$

(88)

If Machine $M_i$ operates in isolation, its production rate is $e_i \cdot \mu_i$. However, if the machines are part of a system, we have to take blocking and starvation into account. For Machines $M_1$ and $M_2$ we denote by $p_{b_1}$ and $p_{b_2}$ the probability of being blocked and find the following production rates:

$$
PR_1 = \mu_1 e_1 (1 - p_{b_1})
$$

(89)

$$
PR_2 = \mu_2 e_2 (1 - p_{b_2})
$$

(90)

With $p_{b_3}$ as starvation probability of Machine $M_3$ we also find:

25
\[
PR_3 = \mu_3 c_3 (1 - p_{s_3}) \tag{91}
\]

Since Machine \(M_3\) processes all the material coming from \(M_1\) and \(M_2\), we must have

\[
PR_1 + PR_2 = PR_3. \tag{92}
\]

Furthermore, \(PR_3\) must equal \(PR\) from Equation (87). It is useful to determine the production rate estimates along several paths in order to detect possible modeling or programming bugs.

In order to determine the probabilities of blocking and starvation, we have to take into account that \(M_2\) tends to be blocked more often than \(M_1\) as the latter machine has priority to fill the buffer.

The blocking probabilities are

\[
p_{b_1} = p(N, 1, 1, 0) + p(N, 1, 0, 0) + \frac{\mu_1 - \mu_3}{\mu_1} p(N, 1, 0, 1) + \max\{\mu_1 - \mu_3, 0\} p(N, 1, 1, 1) \tag{93}
\]

\[
p_{b_2} = p(N, 1, 1, 0) + p(N, 0, 1, 0) + \frac{\mu_2 - \mu_3}{\mu_2} p(N, 0, 1, 1) + \frac{\mu_2 - \max\{\mu_3 - \mu_1, 0\}}{\mu_2} p(N, 1, 1, 1) \tag{94}
\]

and the probability of starvation is

\[
p_{s_3} = p(0, 0, 0, 1) + (1 - \frac{\mu_2}{\mu_3}) p(0, 0, 1, 1) + (1 - \frac{\mu_1}{\mu_3}) p(0, 1, 0, 1) + (1 - \frac{\mu_1 + \mu_2}{\mu_3}) p(0, 1, 1, 1) \tag{95}
\]

where some of the probabilities \(p(x, c_1, c_2, c_3)\) can be zero (see Table 1).

### 3 The Algorithm

It is possible to use a program like Mathematica to evaluate the equations derived above numerically to determine the performance measures. However, the same type of program can also be used to determine analytic expressions that can be transferred to, for example,
a C-program for the analysis of the merge system. This is necessary if one wants to incorporate this merge system in larger decomposition methods for systems with rework loops.

In this section we firstly describe step by step how to develop the required symbolic expressions and to incorporate them in a specialized program. We do not state the expressions explicitly.

1. State density functions (66), i.e. (67) to (72) for Case 3, symbolically in Mathematica for undetermined values $c_i$, $s_i$, and $y^i$.

2. Use these symbolic density functions and symbolic expressions for the undetermined matrices $H_{3,0}^{-1}$ and $H_{3,N}^{-1}$ (in Case 3) in Equations (75) and (77) to determine symbolic expressions for the boundary state probabilities.

3. Plug these symbolic expressions for probabilities and density functions into (79) and the corresponding underlined equations from Table 4. Determine symbolically the matrix $B$ in Equation (85), i.e. the coefficients of the unknowns $c_i$. These coefficients are symbolic expressions that can be transferred into a C-program where it is easy to evaluate them numerically for given numerical values of the system parameters.

4. Plug the same symbolic expressions for probabilities and density functions into (86), (87), (89), (90), and (91) and evaluate these expressions symbolically to determine general symbolic expressions for the inventory level and production rate estimates.

We now describe the steps of the program that uses the symbolic expressions for the matrix $B$, the probability masses and densities, and the performance measures.

1. Determine the matrices $A_{11}$, $A_{12}$, $A_{21}$, and $A_{22}$ from the given parameters ($\mu_i, p_i$, and $r_i, i = 1, \ldots, 3$ and the buffer capacity $N$). Determine matrices $\Omega$ and $A$ via Equations (63) and (64), respectively.

2. Determine the eigenvalues $s_i$ and eigenvectors $y^{[i]}$ of matrix $A$. Determine numerically the matrices that are needed to compute upper and lower boundary probability masses. In Case 3, for example, compute numerically matrices $H_{3,0}^{-1}$ and $H_{3,N}^{-1}$, starting with (76) and (78). Plug all these numerical values into the symbolic expressions of matrix $B$. Compute numerically the inverse of $B$ to solve (85) for the numerical values of the coefficients $c_i$.

3. Plug the all the numerical values computed so far into the symbolic expressions for buffer level and production rate estimates (86) and (87) or (89) to (91). This gives the numerical values of the sought performance measures.
The computer program we wrote essentially consists of 11 very similar parts for the 11 cases. The code is relatively lengthy due to the clumsy symbolic expressions, but its execution times are negligible.

4 Numerical Results

The purpose of this section is to show that the algorithm yields numerical results whose structural properties agree with our knowledge of how the system should behave.

In all experiments we assume the following failure and repair rates:

- \( p_1 = 0.03, r_1 = 0.07 \) (\( M_1 \) is available \( e_1 = \frac{0.07}{0.07+0.03} = 70\% \) of the time)
- \( p_2 = 0.04, r_2 = 0.06 \) (\( M_2 \) is available \( e_2 = \frac{0.06}{0.06+0.04} = 60\% \) of the time)
- \( p_3 = 0.01, r_3 = 0.09 \) (\( M_3 \) is available \( e_3 = \frac{0.09}{0.09+0.01} = 90\% \) of the time)

Processing rates for the four experiments are given in Table 5, the buffer size was \( N = 10 \) in all experiments.

Table 5: Processing rates in the numerical experiments

<table>
<thead>
<tr>
<th>Experiment</th>
<th>( \mu_1 )</th>
<th>( \mu_2 )</th>
<th>( \mu_3 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.1, ..., 20</td>
<td>2</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>0.1, ..., 20</td>
<td>7.5</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>2</td>
<td>0.1, ..., 20</td>
<td>5</td>
</tr>
<tr>
<td>4</td>
<td>7.5</td>
<td>4</td>
<td>0.1, ..., 20</td>
</tr>
</tbody>
</table>

Average production rates and inventory levels as a function of varying processing rates are depicted in Figures 4 to 7. We report the production rate estimates related to Machines \( M_1, M_2 \), and \( M_3 \) (see Equations (89) to (91)).

Figure 4 shows that in the first experiment \( PR_1 \) increases almost linearly with \( \mu_1 \) for low values of \( \mu_1 \). As \( \mu_1 \) increases further, Machine \( M_2 \) is blocked more often and \( PR_2 \) decreases, but the total production rate of the system still increases. As \( \mu_1 \) approaches 20, \( M_2 \) is almost always blocked and \( PR_2 \) approaches zero while \( M_3 \) is almost never starved and \( PR_3 \) approaches \( \frac{\mu_3}{\mu_3 + p_3} = 4.5 \). Figure 5 indicates a strong increase of the average buffer level as \( \mu_1 \) increases.

The second experiment in Figure 6 shows that both \( PR_1 \) and \( PR_3 \) increase with an increasing value of \( \mu_1 \). Note, however, that Machine \( M_1 \)'s production almost completely displaces those of \( M_2 \), i.e. the increase of \( PR_1 \) is much higher than those of \( PR_3 \).
In the third experiment we increased the speed of Machine $M_2$ in a situation where the upstream machines are initially the bottleneck. Note that both $PR_2$ and $PR_3$ increase (as we would expect). What is interesting is a slight decrease of the production rate of the priority one machine $M_1$ in Figure 7. The reason for this decrease of $PR_1$ is that as $\mu_2$ increases, the operation-dependent failures of $M_3$ occur more frequently and $M_1$ is therefore blocked more often.

The last experiment in Figure 8 shows what happens as the speed of the last
machine increases. Initially, almost all of the increase is dedicated to the priority-one machine $M_1$. However, as $M_1$ is eventually saturated, Machine $M_2$ shows a stronger increase of its production rate and $PR_3$ keeps increasing until $M_2$ is saturated as well.
5 Conclusion

We have developed a Markov process model of a three-machine merge system with limited buffer capacity. In order to cope with machine-specific deterministic processing times, we used a continuous material model. The transition equations for the model were formulated and we described a procedure to solve them numerically in a specialized computer program. In order to handle the large symbolic expressions required in this program, we explained how to develop these expressions in Mathematica. Numerical experiments show that the procedures predicts the behavior of the merge system in a way that agrees with our intuition. The developed model is a new building block for the analysis of larger networks that contain merge structures, for example due to rework loops.

References


