Decomposition Analysis of a Deterministic, Multiple-Part-Type, Multiple-Failure-Mode Production Line

by

Diego A. Syrowicz

B.S. in Electrical Engineering and Computer Science, MIT (1998)
B.S. in Management Science, MIT (1998)

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Engineering in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 1999

Author . . . .
Department of Electrical Engineering and Computer Science
May 17, 1999

Certified by . . . .
Stanley B. Gershwin
Senior Research Scientist
Thesis Supervisor

Accepted by . . . .
Arthur C. Smith
Chairman, Department Committee on Graduate Students

Decomposition Analysis of a Deterministic, Multiple-Part-Type, Multiple-Failure-Mode Production Line

by

Diego A. Syrowicz

Submitted to the Department of Electrical Engineering and Computer Science
on May 17, 1999, in partial fulfillment of the
requirements for the degree of
Master of Engineering in Electrical Engineering and Computer Science

Abstract

This thesis proposes an analytic decomposition approximation to estimate the throughput and buffer level of two-part-type flow lines with deterministic processing times and homogeneous buffers. Machines are allowed to have multiple failure modes. Machines operate according to a priority rule, processing higher priority part-types whenever possible. Machines operate on lower priority part-types only when unable to operate on higher priority parts due to either starvation or blockage. The proposed method decomposes the line into a set of two-machine-lines. Two different two-machine lines are described, one for the higher priority part-type, the other for the lower priority part-type. The solutions to the individual two-machine-lines, in combination with the decomposition relationships among those two-machine-lines, yield the analytic approximation to the performance metrics of the line.

Thesis Supervisor: Stanley B. Gershwin
Title: Senior Research Scientist

Acknowledgments

This thesis was supported by many sources. First, and most important, I wish to thank my wonderful family. To my dad and mom, who never ceased to encourage me to pursue and achieve everything I desired, who were there for me unconditionally, and who helped me in every way they could. To my brother Gabriel, who was always willing to visit me, and who kept me up by telling me the good news about home. To my sister Valerie, who always looked forward to keep me company through the phone on weekends, and who never failed to sing to me. And finally, to my grandparents, Shilem, Alicia and Sofia, for always supporting me and showing me their love. To all of them, this thesis is dedicated.

From MIT academia, primarily, I would like to thank my advisor Stanley Gershwin, who supported me technically, financially, and emotionally throughout the process of writing this thesis. He kept me going, helped me up from the hundreds of times I thought there was no way to make things work, and never let me give up. A special thanks to Prof. Gerald L. Wilson, for the plentiful advice and guidance, the lunches, Thanksgiving dinners, and most important, for his friendship.

Finally, I would like to thank my friends. I thank them for being such good friends, for sticking with me in all good times and bad. For the trips we took, the laughs, the classes, and most important, for being who they are.

Contents

List of Figures

Chapter 1

Introduction

1.1 Motivation

The design, operation, and evaluation of production lines are essential parts of the study of manufacturing systems. Most of the fieldwork is done through computer simulation of the stochastic processes underlying the production flow line. However, simulations require a considerable time commitment to construct and run. Recent work done by Gershwin [6] suggests that it is possible to construct a closed formulation of production lines under various operational assumptions. Current formulations developed include those with single-class single-failure-mode deterministic behavior lines [6], single-class single-failure-mode continuous behavior lines [8], single-class multiple-failure-mode deterministic behavior lines [13], and multiple-class single-failure-mode deterministic behavior lines [11]. These formulations usually yield solutions that approximate the solutions of the simulation without the required computational power and time. This thesis constructs a formulation for a production line under deterministic, multiple-part-type, multiple-failure-mode assumptions.

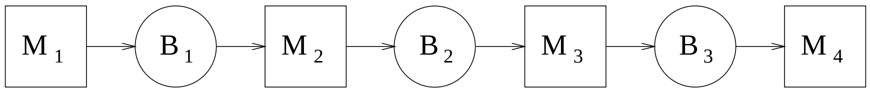

Figure 1-1: Generic Single-Class Transfer Line with Single-Failure-Modes

1.2 Literature Review

1.2.1 Single-class transfer lines with single-failure-modes

A transfer line is a production system whose work proceeds in a linear fashion from one machine to the next. A single-class line is one in which the transfer line only builds one type of part. An example of a single-class flow line is depicted in Figure (1-1). A flow line has machines (M) which perform some work in a part, and such are depicted by the squares in the figure. Parts flow from machines into buffers (B), or storage centers, which are depicted by circles. The arrows that connect the machines and buffers represent the path of work-in-process, and the direction is from left to right.

One way to analyze flow lines is to break them into simpler structures, specifically, two-machine-lines. This is the technique called *decomposition*. Once a formulation and solution to the two-machine-line is found, it may be possible to find an approximate solution to the complete flow line. A two-machine-line is depicted in Figure 1-2. In order to solve a two-machine-line, it is necessary to have a behavior assumption and a representation of the machines and the production process. The representation requires the size of the buffer (N), the failure rate of the machines (p), the repair rates (r), and the processing rates (μ).

The simplest characterization of the production flow is the deterministic model. Under a deterministic assumption, the processing rates of all machines are constant. A machine processes one part, in one time unit, asynchronously from other machines. In addition, a machine cannot process a part if it is starved (there is no available material

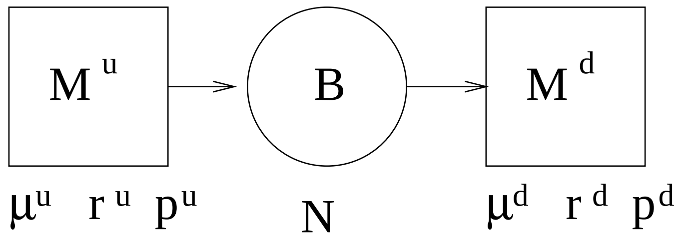

Figure 1-2: Two-Machine-Line Decomposition Component

in the buffer preceding it), or it is blocked (there is no space in the buffer receiving parts from the machine). Generally, a machine is not allowed to fail unless it is working on a part. In addition, in a two-machine-line, the upstream machine is never starved (there is always raw material), and the downstream machine is never blocked (there is always space to put completed parts). The formulation and solution of the resulting deterministic two-machine-line is achieved by solving a two-dimensional Markov chain with $4(N-1)$ states [6]. The solution to such a chain is given as the steady state probability of all states, the line's buffer levels, and the overall production rate.

Through other types of assumptions and solution techniques, other process behaviors can be captured. For example, using a continuous flow assumption, it is possible to allow for machines to have different processing rates.

1.2.2 Single-class transfer lines with multiple-failure-modes

The transfer line models discussed above assume that machines may fail only in one way. Current work done by Tolio [13] allows for a similar formulation of production lines with the added feature that a given machine may fail in one of several modes, and be repaired in the mode corresponding to the specific failure mode. Thus, for example, a machine may fail because a part got stuck, and take an average of 5 minutes to repair, or because the motor exploded, and take an average of 5 days to repair. A two-machine-line building block representation is depicted in Figure 1-3.

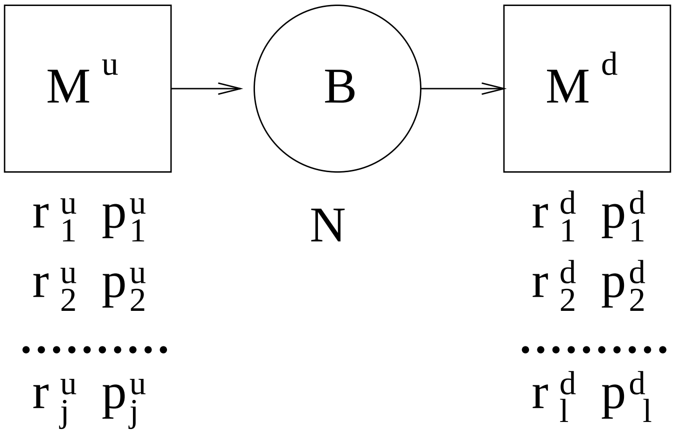

Figure 1-3: Singe-Part-Type, Multiple-Failure-Mode Two-Machine-Line

Single-failure-mode models cannot deal directly with multiple failure modes. In order to use those models on lines with multiple failure modes, one has to first average the multiple failure and repair rates, and use the averages to represent the parameter for a single-failure-mode machine. The problem with averaging is that, for sets of considerably different failure modes, the variance cannot be captured accurately. This results in less accurate steady state solutions for the performance measures.

Another feature of multiple failure lines is that in the decomposition process, two-machine-lines can assign failure modes to account for the probability of starvation and blockage due to failures of machines outside of the two-machine-line. These failure modes are called *virtual failure modes* as they are not *real failure modes*. During decomposition, the steady state solution is reached when the convergence of the production behavior of every two-machine-line is achieved [3]. For every two-machine-line, behavior paramenters are analyzed and changed in an ordered way until convergence is achieved. By allowing a two-machine-line to account directly for new possibilities of failure, the accuracy of the solution is usually improved.

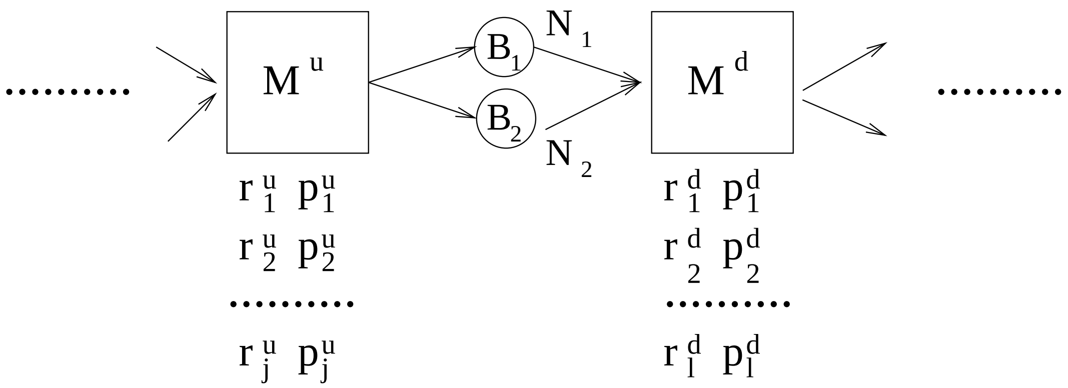

Figure 1-4: Multiple-Part-Type Line

1.2.3 Multiple-class transfer lines with single-failure-modes

Recent work conducted by Nemec [11] formulated and solved for deterministic behavior lines that processed more than one part type. A simple multiple-part-type line is depicted in Figure 1-4.

Because the line works on different parts, there must be a policy. Nemec describes the policy as one with priorities. Thus, part 1 is always worked on if there are parts to work with and the machines are not blocked or starved. Only if there are no part 1s to work with, part 2s are started, and so on. If a higher ranking part arrives to be worked on while a machine is working on a lower ranking part, the next part to be processed will be the higher ranking part. Setup times are assumed to be zero. Buffers are *homogeneous*. In other words, buffers only hold parts of a single part type. Because there is a buffer for every part-type, blockage and starvation are part-dependent events.

1.3 Multiple-class-type, Multiple-failure-mode transfer lines

Nemec formulated a deterministic single-failure multiple-class transfer line. However, this formulation only worked for small two-class-type lines. The reason why the formulation worked only in a limited set of lines is that possibly, the two-machine-lines are unable to describe accurately all the failures, blockages, and starvations possible due to other part-types and other machines.

One goal for the research of decomposition is to achieve a formulation that accounts for both multiple-part-types and different production speeds for each machine. Nemec's work tries to account for multiple-part-types. However, he was unsuccessful in formulating a deterministic model that could work for more than six machines and two part-types. The extension to a continuous case model (one with different processing speeds per machine) would prove to be difficult. The work done by Tolio suggests that there is a potential solution to the underlying problems in Nemec's model.

By using a multiple-failure-model formulation, most of the second moments in the multiple-type line could be captured. This would increase the accuracy and decrease the complexity of the desired formulation. The first step of this thesis will be to determine the state transition dynamics of such a model. Then, a decomposition method for this type of line will be determined. Finally, the general two-machine building blocks will be constructed and analytically solved.

Chapter 2

Decomposition Derivation

2.1 Introduction

In this chapter, a decomposition analysis of a processing line with finite homogeneous buffers, unreliable machines, and two different part-types is presented. Like the decomposition analysis done by Nemec [11], the multiple-part-type line analysis is conducted by decomposing the line into single-part-type two-machine sections corresponding to all real homogeneous buffers. Although the two-machine sections are part-type specific, the state transitions seen by all of these sections are interwoven with events occurring in other two-machine lines, including ones of different part type.

2.2 Notation

The decomposition of the two-part-type line is illustrated in Figure 2-1. The notation used to refer to items within the decomposition will follow, for the most part, the convention set by Nemec [11]. Part-specific notation must be introduced to deal with new event types contemplated by the decomposition; those will be introduced in later sections.

Machines and buffers in the decomposition are part-type specific, and therefore,

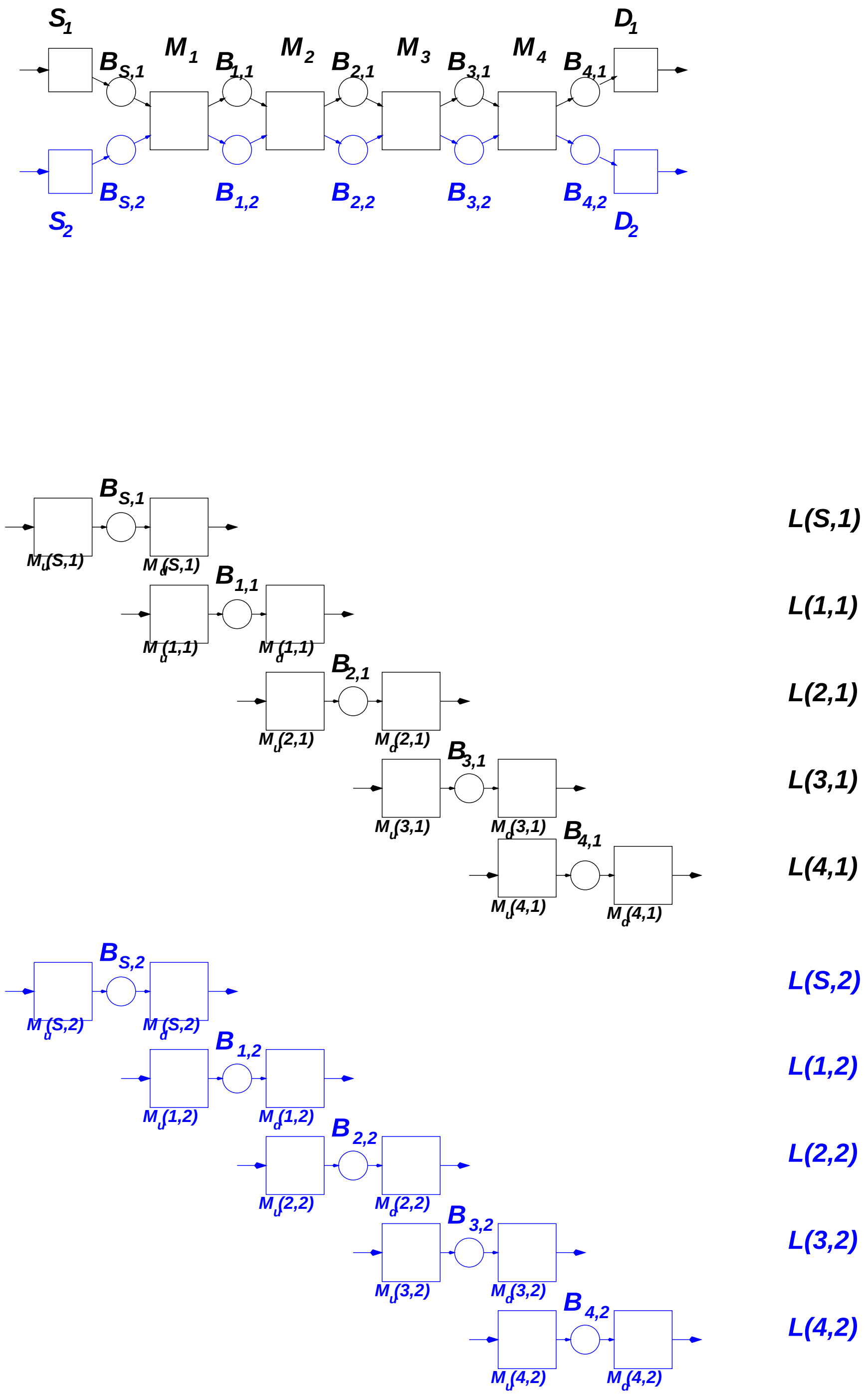

Figure 2-1: Decomposition for typical two-part-type line

unlike single part-type lines, identifiers must include a part-type index, $c = \{1, 2\}$, in order to differentiate between similar items for different part-types. In the decomposition, each buffer $B_{\eta,c}$ has a corresponding two-machine line, $L(\eta, c)$; where η is the two-machine-line index number, and c the part-type. Machines corresponding to the real processing line are called *real machines*, whereas machines from the two-machine-lines are called pseudo-machines. The upstream pseudo-machine for $L(\eta, c)$ is denoted $M^u(\eta, c)$; the downstream pseudo-machine is denoted $M^d(\eta, c)$. The size of the real buffer $B_{\eta,c}$ is the same size as $B(\eta, c)$, and is denoted $N(\eta, c)$. The current buffer level of $L(m, c)$ is denoted by $n(\eta, c)$.

2.3 Part 1 Decomposition

In the real line the introduction of multiple part-types creates a more complex environment than that of single part-type lines. Such added complexities must also be captured by the decomposition analysis, and, as a result, additional notation is necessary. A development of the decomposition analysis for type one parts follows. The additional notation required is described as need arises.

2.3.1 New Events in the Part-1 Two-Machine-Line

Like in the Tolio decomposition [13], pseudo-machines can suffer from *real failures* and *virtual failures*. Real failures are failures of the real machines as represented by the pseudo-machines of the two-machine-line. Virtual failures are the failures modes introduced to account for the effect that real failures outside of the two-machine-line have on the two-machine-line itself. However, the events which an observer standing in the buffer of a part-1 two-machine-line would see are more complicated than those in a single-part-type line. Sometimes, from the perspective of the observer, when a pseudo-machine is not allowed to work[1], it could still fail[2]. The new failure types are *idleness failures*, and *failure-mode-changes.*

Idleness Failures

As in the case with the one-part-type deterministic line model, machines are prevented from working when they are starved or blocked. However, since buffers are homogeneous, when a real machine is starved or blocked for part 1, it is not necessarily blocked or starved for other part-types. Indeed, because the machine is not in any real failure mode, the it can process lower-priority part-types. While the real machine is working on such part-types, the it can fail as well. From a part-1 observer's perspective, this means that while the observed buffer is completely full or empty, failures that could not occur in the one-part-type case are now possible. If the originally blocked or starved pseudo-machine gets unblocked or unstarved, and it is down because of a real failure, the pseudo-machine is said to have seen an *idleness failure of mode j*, where j is the indicator of the failure mode observed. The identifier q will be used to describe such probability. Thus, for example, $q_3^u(5,1)$ is said to be the probability that $M^u(5,1)$ fails in mode 3 when it is blocked. Notice that idleness failures in the two-machine-line context occur only when the upstream pseudo-machine is blocked, or the downstream pseudo-machine is starved.

Failure-Mode Changes

When failures are virtual, although part 1 cannot be processed by the affected real machine, it is conceivable that part 2 could be processed instead. As with idleness failures, real machines could continue working on lower priority parts while the part-1 virtual failure is repaired. The usual scenario would be one in which the virtual failure would be repaired while the real machine worked on lower-priority parts, and

[1] whether because it is down, or it is blocked or starved
[2] since the machine could be doing lower priority parts

thus it would return to working on part 1. In a similar way, even if some machine failure caused a virtual failure to the lower-priority-part production, it is conceivable that such failure would be repaired before the **initiating failure** was repaired. The initiating failure is defined as the failure that caused production to start for a lower-priority part-type. In such instances there would be nothing new added to what a part-1 observer would see. However, there is the possibility that the initiating failure is repaired while some other failure was felt by the observed two-machine line. In other words, this is the case that will be referred to as a *failure-mode change*, and will be symbolized by variable z.

The importance of failure-mode changes relies on the fact that even though the part-1 pseudo-machine will continue to be down, there would be a change in the repair probability. In order to capture this probability change, a transition probability between down modes must be specified.

It is important to notice two important observations in failure-mode changes. The first is that a failure mode change can only occur from the initiating mode to a mode corresponding to a machine which is closer to the observer's location. The reason for this is that the initiating failure corresponds to a real failure of some machine, which has propagated by means of starvation or blockages to the observer's location. A real machine under a real failure mode may not work on any part type, and thus, even if machines farther away from the observer's corresponding real machine fail, those failures will not propagate to that location unless the initiating failure is repaired. However, real failures that occur to machines closer to the observer's location than the real machine to which the initiating failure corresponds will block the effects of a repair of the initiating failure.

The other observation has to do with the *timing* of a failure-mode change. The situation in which a failure occurs when processing a lower-priority part is not enough to cause a failure-mode change. After all, not only must a new failure occur and the initiating failure be repaired, but also the repair of the initiating failure must propagate to the new failure location before the initiating failure is repaired. In other words, a failure-mode change is only said to occur after both, the initiating failure is repaired, and part-1s have propagated to the location of the new failure. If the new failure was repaired for the lower-priority parts before the full propagation occurred, the initiating failure's repair would reach the observer's location, thus eliminating the need for a failure change possibility.

Assumption 1: *A failure-mode change is not experienced by an observer in the part-1 two-machine line until the initiating failure is repaired, and part-1 type parts have propagated to the new failure's location.*

2.3.2 Calculation of Idleness Failure Probability

The changes in the decomposition process with respect to Tolio's single-part-type decomposition have to do with the new failure types. The idleness failures complicate the process insofar as the boundary states are concerned. In other words, since idleness failures only occur in blockage or starvation instances (i.e. the observer's buffer is full or empty), then q's will only be seen in boundary transition events.

Because q is conditional on being at a given boundary state, the expression for q is only contingent on the probability of a given failure type occurring. Since a pseudo-machine could only fail if the local real machine was working on an alternative part type, q will be an expression which includes the probability of the real-machine working on the alternative part type. In other words, idleness failures only occur due to failures of the local real machines. Therefore, idleness failures only cause failures to real modes. In a two-part line, this translates into

$$q_j^u(\eta, 1) = p_j(\eta) P\left(M^u(\eta, 2) \text{ non-idle }\right) \tag{2.1}$$

and

$$q_l^d(\eta, 1) = p_l(\eta + 1) P\left(M^d(\eta, 2) \text{ non-idle }\right) \tag{2.2}$$

where $p_j(\eta)$ and $p_l(\eta)$ are the probabilities of real failures for the real machine η, and the non-idleness probability can be calculated as a sum of states from the two-machine line analysis.

2.3.3 Probability of Change of Failure-Mode

A convenient way to begin to think about failure-mode changes is to study the relationship between neighboring machines which are in identical failure modes. When a failure occurs somewhere in the line, as the failure propagates through the line causing virtual failures, more than one observer will see this failure mode. Because all intermediate buffers for part-1 would empty out as the virtual failure propagates downstream, then failure-mode changes propagate as well. In fact, when a failure-mode change is experienced by an observer, all the observers which were in the same failure mode will simultaneously experience it too. The reason for this behavior relies in the fact that all part-1 buffers are empty between the initiating failure location and any observer who has felt the failure. Thus, if any of the observers has seen a failure mode change, since all buffers between his location and any observer in the initiating failure mode are still empty, all observers see the same failure type.

$$z^u_{j',j}(\eta + 1, 1) = z^u_{j',j}(\eta, 1) \text{ , for } j' < j, \tag{2.3}$$

and

$$z^d_{l',l}(\eta - 1, 1) = z^d_{l',l}(\eta, 1) \text{ , for } l' > l, \tag{2.4}$$

where j' and l' refer to the initiating failure modes, and j and l refer to the mode to which a transition occurred.

Because of (2.3), it is only necessary to calculate $z^u_{j',j}(\eta, 1)$ for the machine η to which mode j belongs. The complexity of calculating this probability increases as the separation between j' and j increases.

The simplest case is when j' and j refer to adjacent real machines. For example, if for simplicity we assume that mode numbers correspond to specific machines[3], $z^u_{3,4}(4, 1)$ would refer to the probability that the observer in $L(4, 1)$ sees a failure mode change from failure mode 3 to failure mode 4. More specifically, in the case of a two-part line, $z^u_{3,4}(4, 1)$ means that the following events happened in order (from that observer's point of view):

1. A virtual failure of type three occurred in $M^u(4, 1)$ while working type 1 parts.
2. Although $M^u(4, 1)$ is virtually down, M_4 is not truly down and can work on type 2 parts.
3. While making type two parts, M_4 fails.
4. M_3 got repaired while M_4 was still down.

The moment that M_3 gets repaired, a part is put in $B(3, 1)$. Following with *Assumption 1*, if $M^u(4, 1)$ was not repaired at the same time, this immediately means that a change of failure from mode 3 to mode 4 was experienced by the observer.

The probability calculation for $z^u_{3,4}(4, 1)$ is dependent on failure-mode 3 having been experienced by M_4. Thus, the calculation reduces to the probability of any of the following events occurring:

- $M^u(4, 1)$ fails and M_3 gets repaired at the same time.
- $M^u(4, 1)$ fails, and after one time step, M_3 gets repaired but $M^u(4, 1)$ is not repaired.

[3] which would also imply that each machine has only one failure mode.

- $M^u(4,1)$ fails, after one time step neither M_3 nor $M^u(4,1)$ gets repaired, and after two time steps M_3 gets repaired and $M^u(4,1)$ is not repaired.
- $M^uu(4,1)$ fails, after $s-1$ time steps neither M_3 nor $M^u(4,1)$ gets repaired, and after s time steps M_3 gets repaired and $M^u(4,1)$ is not repaired.

Before calculating the probability corresponding to such events, a simplifying assumption must be made. Specifically, that after an originating failure occurs, part-2 would start to be processed, and work on part-2 would not be starved or blocked on M_4.

Assumption 2: *Once an originating failure occurs, part-2 is processed by the line without interruption unless there is another failure, or the initiating failure is repaired*

What *Assumption 2* means is that the probability that pseudo-machines are idle for part-2 do not have to be calculated. This assumption may be justified by the fact that one is most interested in evaluating the performance of lines that have limited capacity. Thus, in a two-part line, if there is overwhelming capacity, all demand would be satisfied easily. However, if capacity is limited, part-1 would usually be the one being processed, and when there was an opportunity to work on part-2, it would rarely be the case that the machine would not be able allowed to do so because of starvation or blockage.

Given *Assumption 2*, the calculation reduces to the sum of all the aforesaid events:

$$\begin{aligned} z^u_{3,4}(4,1) &= p_4r_3 + p_4(1-r_3)(1-r_4)r_3 + \ldots \\ &= p_4r_3\sum_{s=0}^{\infty}[(1-r_3)(1-r_4)]^s \\ &= \frac{p_4r_3}{1-(1-r_3)(1-r_4)} \end{aligned}$$

Generalizing,

$$z^u_{j-1,j}(j,1) = \frac{p_j r_{j-1}}{1-(1-r_{j-1})(1-r_j)}$$

Using (2.3),

$$z^u_{j-1,j}(\eta,1) = \frac{p_j r_{j-1}}{1-(1-r_{j-1})(1-r_j)} \text{ , for } \eta \geq j. \tag{2.5}$$

The calculation of $z^u_{j',j}(\eta,1)$ for $j > j'+1$ is harder because as the separation between real machines increases, there are increasingly more event-sequences through which a change of failure mode is possible. In order to calculate this quantity, another simplifying assumption must be made: that once a machine is in originating failure mode j', all machines between j' and j are up. The reason why this is an acceptable assumption is that if the real machines between j' and j were down, or allowed to fail and then be repaired before the transition from j' to j occurs, the probability contribution would be comparatively small. Machines which start down between j' and j must be repaired before the effects of the repair of j' reaches them. If this was not the case, then the failure-mode change would not be from j' to j, but from j' to some $j'' < j$. However, this repair would require a repair probability factor, which would make the probability contribution smaller. Similarly, terms which include failures of machines between j' and j will not be included as otherwise not only would the failure probability need be included as a factor, but also its corresponding repair probability.

Assumption 3: *In calculating the failure-mode change probability from j' to j, all machines between j' and j are assumed to be up. In addition, terms requiring failures of real machines between j' and j will be ignored as their probability contribution is minimal.*

Using *Assumption 3* and equation (2.3), it can be shown that

$$z^u_{j',j}(\eta,1) = \frac{p_j r_{j'}(1-r_j)^{j-j'-1}\prod_{l=j'+1}^{j-1}(1-p_l)^{l-j'+1}}{1-(1-r_{j'})(1-r_j)\prod_{v=j'+1}^{j-1}(1-p_v)} \tag{2.6}$$

for $\eta \geq j$, and $j' < j$.

Note that, following the convention of single failure modes per machine, j's refer to both failure mode type, and machine number. In the case that multiple failure modes exist for machines, quantities like $j - j'$ must be expressed in terms of machine numbers.

A similar process for the downstream pseudo-machine yields

$$z^d_{l+1,l}(l,1) = \frac{p_l r_{l+1}}{1-(1-r_{l+1})(1-r_l)}$$

Using (2.4),

$$z^d_{l+1,l}(\eta,1) = \frac{p_l r_{l+1}}{1-(1-r_{l+1})(1-r_l)}\ , \tag{2.7}$$

and

$$z^d_{l',l}(\eta,1) = \frac{p_l r_{l'}(1-r_l)^{l'-l-1}\prod_{h=l+1}^{l'-1}(1-p_h)^{h-l+1}}{1-(1-r_{l'})(1-r_l)\prod_{v=l+1}^{l'-1}(1-p_v)} \tag{2.8}$$

for $l \geq \eta$, and $l' > l$.

2.3.4 Calculation of p and r

The decomposition derivation for p and r follow the methodology line of Tolio's decomposition [13], but for a few modifications to the equations. Once again, notation must be slightly modified to accommodate the fact that there are two part types.

Notation Summary for Decomposition

The required notation for the upstream pseudo-machine is

$W^u(\eta, 1)$ The probability that machine $M^u(\eta, 1)$ is operating on a part.

$D^u_f(\eta, 1)$ The probability that machine $M^u(\eta, 1)$ is down with real failure mode f.

$X^u_{(j,f)}(\eta, 1)$ The probability that machine $M^u(\eta, 1)$ is down with virtual failure mode (j, f), where f refers to the real failure mode of initiating machine j (upstream from η).

$P_{s(j,f)}(\eta, 1)$ The probability that machine $M^d(\eta, 1)$ is starved due to failure f from initiating machine j (upstream from η).

$E(\eta, 1)$ The efficiency of two-machine line (η,1).

Decomposition Derivation of p and r

$W^u(\eta, 1)$, the probability that $M^u(\eta, 1)$ is working on a part, is simply $E(\eta, 1)$. The reason for this is that the upstream pseudo-machine in the two-machine line cannot be starved, and thus it will always be working on a part when it is not down or blocked.

$$W^u(\eta, 1) = E(\eta, 1) \tag{2.9}$$

Virtual failures are introduced to mimic the effects of failures of non-local real machines in the two-machine line. The effects of such failures propagate as starvations or blockages. Therefore, it must be the case that there is a correspondence between virtual failures and starvations/blockages in neighboring two-machine lines. More specifically, the probability of a virtual failure in $M^u(\eta, 1)$ starting at time t, $X^u_{(j,f)}(\eta, 1)$, must be equal to the probability of starvation of $M^u(\eta - 1)$ starting at time t, $P_{s(j,f)}(\eta - 1, 1)$.

$$X^u_{(j,f)}(\eta,1) = P_{s(j,f)}(\eta-1,1) \tag{2.10}$$

The frequency of entering into a virtual failure mode must be equal to the frequency of leaving it. Essentially there are two ways in which a virtual failure mode could be entered or exited: (1) by real failures/repairs, and (2) by changes in failure-modes. In the context of virtual failure modes, this translates into:

$$\begin{aligned} X^u_{(j,f)}(\eta,1)\left[r^u_{(j,f)}(\eta,1) + \sum_{(j,f)'} z^u_{(j,f),(j,f)'}(\eta,1)\right] = \\ W^u(\eta,1)p^u_{j,f}(\eta,1) + \sum_{(j,f)''} X^u_{(j,f)''}(\eta,1) z^u_{(j,f)'',(j,f)}(\eta,1) \end{aligned}$$

where the sum $\sum_{(j,f)'} z^u_{(j,f),(j,f)'}$ is over $(j,f)' > (j,f)$. In other words, mode $(j,f)'$ corresponds to a real machine closer to η than (j,f). The sum $\sum_{(j,f)''}$ is over $(j,f)'' <$ (j,f), i.e. $(j,f)''$ is mode corresponding to a real machine farther away from η than (j,f).

Introducing the notation $\tilde{r}^u_{(j,f)}(\eta,1)$ as representing the sum of all probabilities of leaving down state (j,f):

$$\tilde{r}^u_{(j,f)}(\eta,1) = r^u_{(j,f)}(\eta,1) + \sum_{(j,f)'>(j,f)} z^u_{(j,f),(j,f)'}(\eta,1) \ ,$$

then,

$$X^u_{(j,f)}(\eta,1)\tilde{r}^u_{(j,f)}(\eta,1) = W^u(\eta,1)p^u_{j,f}(\eta,1) + \sum_{(j,f)''} X^u_{(j,f)''}(\eta,1) z^u_{(j,f)'',(j,f)}(\eta,1)$$

Using (2.9) and (2.10),

$$P_{s(j,f)}(\eta-1,1)\tilde{r}^{u}_{(j,f)}(\eta,1) = E(\eta,1)p^{u}_{j,f}(\eta,1) + \sum_{(j,f)''} Ps_{s(j,f)''}(\eta,1)z^{u}_{(j,f)'',(j,f)}(\eta,1)$$

Re-arranging terms,

$$p^{u}_{(j,f)}(\eta,1) = \frac{\tilde{r}^{u}_{(j,f)}(\eta,1)Ps_{(j,f)}(\eta-1,1) - \sum_{(j,f)'} Ps_{(j,f)'}(\eta-1,1)z^{u}_{(j,f)',(j,f)}(\eta,1)}{E(\eta,1)} \tag{2.11}$$

Since

$$E(1,1) = E(2,1) = ... = E(\eta,1) = ... = E(m,1)\ , \tag{2.12}$$

then by (2.11) and (2.12)

$$p^{u}_{(j,f)}(\eta,1) = \frac{\tilde{r}^{u}_{(j,f)}(\eta,1)Ps_{(j,f)}(\eta-1,1) - \sum_{(j,f)'} Ps_{(j,f)'}(\eta-1,1)z^{u}_{(j,f)',(j,f)}(\eta,1)}{E(\eta-1,1)} \tag{2.13}$$

Using a similar method and notation, $p^{d}_{j,f}(\eta,1)$ is found to be

$$p^{d}_{(j,f)}(\eta,1) = \frac{\tilde{r}^{d}_{(j,f)}(\eta,1)Pb_{(j,f)}(\eta+1,1) - \sum_{(j,f)'} Pb_{(j,f)'}(\eta+1,1)z^{d}_{(j,f)',(j,f)}(\eta,1)}{E(\eta+1,1)} \tag{2.14}$$

2.4 Part 2 Decomposition

This section develops the decomposition analysis for the part 2 behavior. Part 2 is the part with the lowest processing priority, and machines in the real line will only work on such part-type when blocked or starved for the higher priority part-types. For notation reasons, a *part c realm* for a given machine or two-machine line will be defined as the space in time when such machine or two-machine line is allowed to work on part-type c ($c = \{1, 2\}$).

2.4.1 Observable Part-2 Events

Part-2 observers see a different event space than the one seen by those standing in part-1 buffers. Since working on part 2 only occurs because a virtual failure occurred for some machine in the part-1 realm, then there are different ways in which part 2 production could start. This is important because the failure type which occurred in the part-1 realm (the initiating failure), determines the way in which the production in the part-2 realm could fail. Thus, for example, if $M^u(5,2)$ started production because of initiating failure mode 3 (M_3 failed), then $M^u(5,2)$ can only fail if either

- M_3 is repaired and $M^u(5,1)$ is repaired from the corresponding virtual failure.
- $M^u(5,2)$ fails in virtual mode 4 (i.e., due to the failure of M_4).

However, if $M^u(5,2)$ entered production because of initiating failure mode 1, then $M^u(5,2)$ can fail if

- M_1 is repaired and $M^u(5,1)$ is repaired from the corresponding virtual failure.
- $M^u(5,2)$ fails in virtual mode 4.
- $M^u(5,2)$ fails in virtual mode 3.
- $M^u(5,2)$ fails in virtual mode 2.

Thus, there are various up-states needed to have the memory required to account for the different failure modes corresponding to each initiating failure mode.

The ideal part-2 machine model would require new states to differentiate between the different down-modes on which part-2 could be. Indeed, each up state would have down states which correspond to identical failure modes. The reason for the different

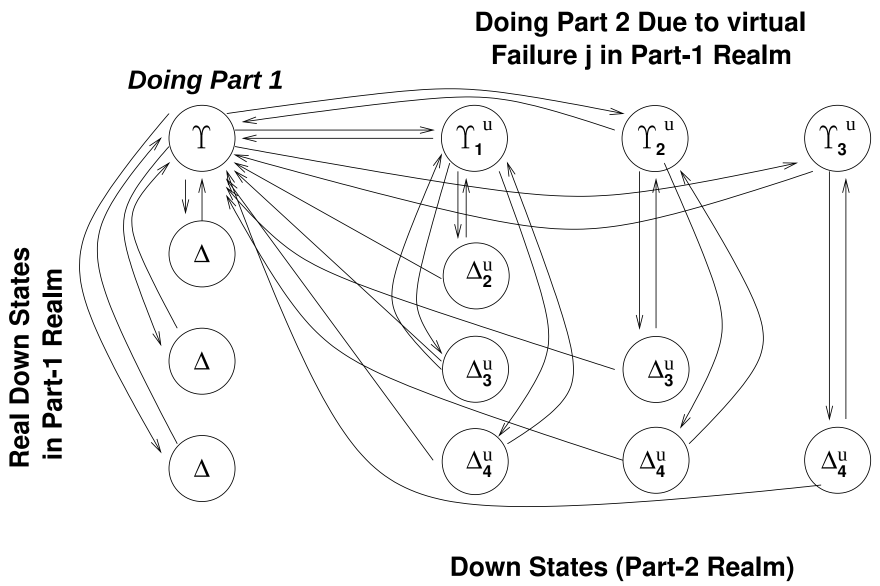

Figure 2-2: Ideal Machine Model for Typical $M^u(4,2)$

down-states is that when a virtual failure occurs in the part-2 realm, such could be repaired before the initiating failure. Because a virtual failure for a given up-mode could only return to doing part 2 in that same up-mode (or simply fail into doing part-1), then such down-mode could not be shared by multiple up-modes. Figure 2-2 depicts the ideal pseudo-machine model for part 2.

Unlike part-1 observers, part-2 observers would not see idleness failures. In addition, there would not be changes in failure mode since part 2 is the lowest priority part, and on blockage or starvation no other part would be processed. Because the set of possible states is extremely complex, simplification is necessary.

Simplifications

Several solvable models could be devised to capture some part of the behavior of part 2. Each model would sacrifice different details and events. The choice of model is hard, especially without apriori knowledge on what events are most important.

The simplification that is pursued here is one in which, for every pseudo-machine, there are multiple up-modes and one down-mode. Some of the reasons why such model seems reasonable are:

- It is more likely that after the initiating failure occurs, if a virtual failure occurs in the part-2 realm right after, the initiating failure will be repaired before the virtual failure does. Therefore, the required state transition in the part-2 realm would be from a virtual down mode to the *down doing type one parts* down mode. Thus, down states specific to each up mode are usually not required.

- Even with no up-state-specific failures, multiple up-modes will allow to have different probabilities of failing. Such probabilities could be adjusted to include all the different ways in which a failure could occur. Similarly the ways in which the up-mode could be entered could be adjusted to include intermediate failures. A multiple-down-mode model would not allow for a similar calculation to be done to adjust for the multiple up-state behavior.

The approximation is usually a realistic one in lines with similar machines. Potentially problematic cases would be the ones with machines which fail very often (in comparison to others in the line). However, if a machine fails and it is repaired very often, then its effects would usually not propagate to be felt as virtual failures. Also, if the machine failed very often, and was repaired slowly (comparatively speaking) then the whole analysis would be almost useless as the problems in production are due to that very unreliable machine.

2.4.2 Two-Machine Line Model and Parameters

The two-machine line model that will be used will thus be the one with multiple up-modes and one down-mode. In addition, there will neither be idleness-failures, nor up-mode change probabilities.

The complication with the model comes about from the fact that the line is to process part 2. To find the needed parameters (r's and p's) several assumptions and calculations must be performed.

Symmetry of $p_j(\eta, 1)$ and $r_l(\eta, 2)$

In order to calculate $r_l(\eta, 2)$ one must first recall the reasons why M_η would do part 2. part 2 is processed by M_η because, in the part-1 realm, $M^u(\eta, 1)$ went into some virtual failure. Thus, the probability of M_η starting to process part 2 is at least the same as the probability that the initiating failure occurred. The probability could be larger if adjustments were to be made to account for the fact that once in an up-mode, a virtual failure could occur and been repaired before the initiating failure was repaired. For simplification reasons, and since this was the original simplification motivation in the choice of the model, such probabilities will be ignored. Thus:

$$r_i^u(\eta, 2) = p_j^u(\eta, 1) \tag{2.15}$$

and

$$r_f^d(\eta, 2) = p_l^d(\eta, 1) \tag{2.16}$$

where the upstream-machine's down-mode j in part 1 corresponds to the same *real mode* as up-mode i in part 2, and equivalently between modes f and l for the downstream pseudo-machine.

Calculation of $p_i^u(\eta, 2)$ and $p_f^d(\eta, 2)$

The calculation of failure probabilities in part 2 pose a harder problem than the repair probability. A failure from a given upstate could occur because the initiating failure was repaired, or because another failure was felt within the part-2 production realm. This quantity will be solved by finding relationships between neighboring two-machine lines. It is noteworthy to mention, however, that such probability should be some function of the repair probability for the initiating machine failure.

2.4.3 Solution Cycle for Part 2

The decomposition process for part 2 cannot follow the Tolio's decomposition method exactly since the problem presented must deal with issues of multiple up-states instead of multiple down-states, in addition to incorporating part-1 events. The only variables that remain to be found for use in the two-machine lines are: $p_i^u(\eta,2)$, the probability that $M^u(\eta,2)$ fails while in in up-mode i, and $p_j^d(\eta,2)$, the probability that $M^d(\eta,2)$ fails while in up-mode j. There are three components that come into play in such probabilities; looking into $p_i^u(\eta,2)$ they are:

1. The probability that the initiating failure j was repaired for $M^u(\eta,1)$
2. The probability that a real failure occurs to M_η while it is working on part 2
3. The probability that a $M^u(\eta,2)$ experiences a virtual failure by means of starvation

The first component, the probability of the repair of the initiating failure, is $r_j(\eta)$, which is the r for the real machine to which j belongs.

The second component, a real failure of M_η while it is working on part 2, is also related to the real failure parameters of that machine. Given that the real machine is in the part-2 realm, and it is in up-state i, it can only fail if it is not starved or blocked (i.e., working on a part). Thus the probability of that happening is just the sum of all the probabilities of *real* failure $\sum_{g=1}^{G} p_g(\eta)$ (where G is the number of real failures for machine η) times the probability of that $M^u(\eta,2)$ is non-idle.

The final component can be related to failures (virtual or real) in the upstream neighboring two-machine-line. In order for these failures to be felt by $M^u(i,2)$, propagation by means of starvation must occur. Thus, looking at virtual failures for the upstream machine $M_i^u(\eta,2)$, where the i symbolizes that the machine is in up mode i, a virtual failure occurs after either:

- $M^u(\eta-1,2)$ is in mode i and fails while $B(\eta,2)=0$, and $M_i^u(\eta,2)$ does not fail.

- $M^u(\eta-1,2)$ is down and $B(\eta-1,2)=1$, and $M_i^u(\eta,2)$ does not fail.

Defining

$W^{u,0}(\eta-1,2,i)$ The probability that $M^u(\eta-1,2)$ is up in mode i, and $B(\eta-1,2)$ has zero parts in it

$D^{u,1}(\eta-1,2)$ The probability that $M^u(\eta-1,2)$ is down and $B(\eta-1,2)$ has only one part in it

where both terms can be calculated by the sum of probabilities of states in the two-machine lines, and defining $P_M(\eta)$ as the sum of the probabilities of all possible real failure types for $M(\eta)$,

$$P_M(\eta)=\sum_{g=1}^{G}p_g(\eta),$$

then, adding the probabilities for the aforesaid components:

$$\begin{aligned} p_i^u(\eta,2) &= r_j^u(\eta,1)+P_M(\eta) \\ &\quad +\left(1-P_M(\eta)\right)W^{u,0}(\eta-1,2,i)p_i^u(\eta-1,2) \\ &\quad +\left(1-P_M(\eta)\right)D^{u,1}(\eta-1,2)\left(1-\sum_{alli'}r_{i'}^u(\eta-1,2)\right) \end{aligned} \tag{2.17}$$

where mode i in part-2 pseudo-machines is equivalent to mode j in part-1 pseudo-machines.

Following a similar methodology for the downstream machine,

$$
\begin{aligned}
p_f^d(\eta,2) \quad &= \quad r_l^d(\eta,1) + P_M(\eta+1) \\
&\quad + (1 - P_M(\eta+1))\, W^{d,0}(\eta+1,2,f) p_f^d(\eta+1,2) \\
&\quad + (1 - P_M(\eta+1))\, D^{d,1}(\eta+1,2) \left(1 - \sum_{all\, f'} r_{f'}^d(\eta+1,2)\right) \qquad (2.18)
\end{aligned}
$$

Chapter 3

Part-1 Two-Machine Line

3.1 Introduction

This chapter presents the solution technique for the two-machine line chosen to represent the behavior of part 1. As discussed in Chapter 2, in order to analyze a long line it is necessary to decompose it into two-machine-lines. The Markovian model chosen for part 1 is one similar to Tolio's model [12] insofar as having one up-mode and multiple down-modes. However, because of changes in failure mode, it is now possible to go from a given down-mode to other down-modes. In addition, the introduction of idleness failures will further make the the solution technique differ at the boundary states.

Figure 3-1 depicts the transition space for a typical upstream machine in the two-machine line.

The introduction of changes of failure mode creates several conceptual and mathematical problems in the solution method pursued. Because of this difficulty, this chapter concentrates in developing the model that introduces idleness failures into the part-1 two-machine-line. The work done with the introduction of changes in failure mode is presented in Appendix B.

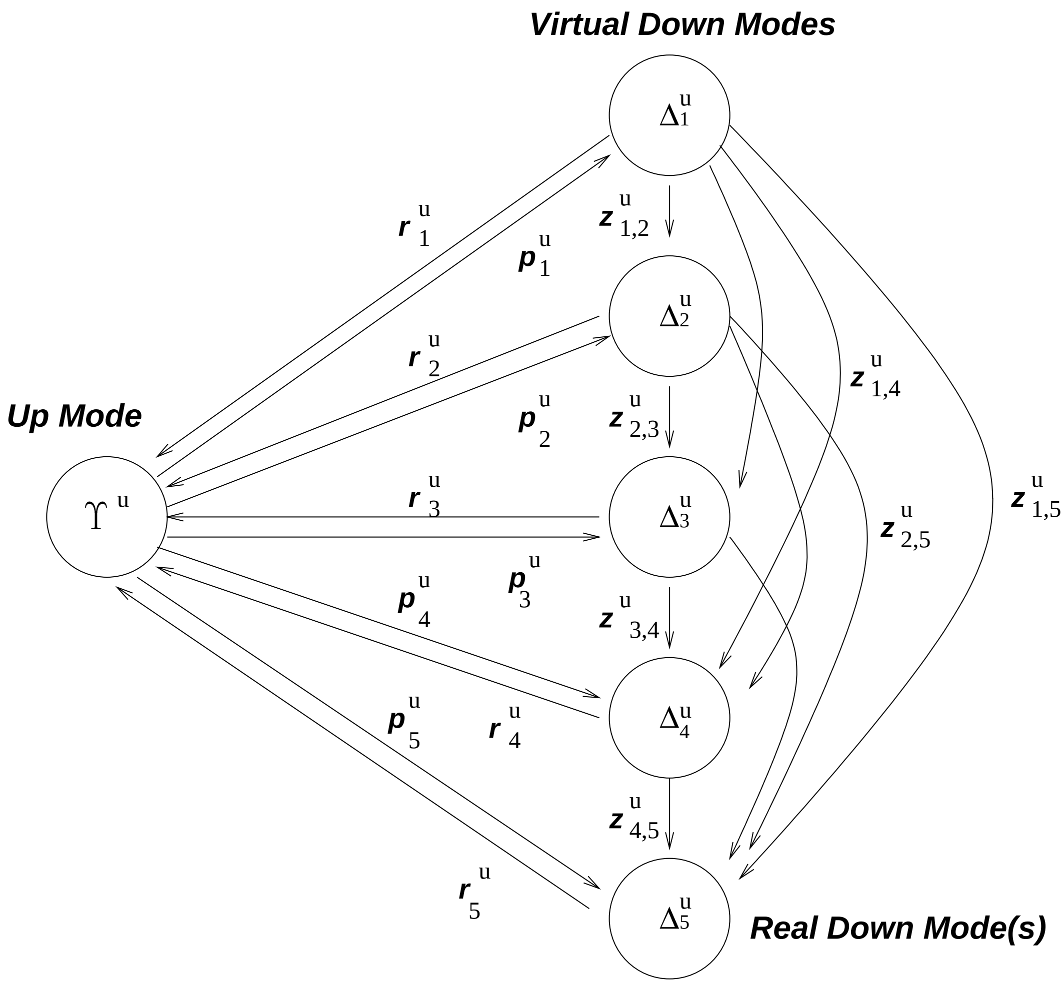

Figure 3-1: State Space for $M^u(5,1)$

3.2 Notation

The notation used will follow as much as possible that defined by Tolio [12]. Δ_j^u represents down mode j for the upstream machine, and Δ_l^d represents down mode l for the downstream machine. Because there is only one up state for each machine, Υ^u and Υ^d represent the up states of the upstream and downstream machines respectively.

Probabilities of repair or failure are still represented by r_j^u, r_l^d, p_j^u, and p_l^d. New notation is introduced because of failure mode changes. A probability expressed as $z_{t,j}^u$ represents the probability of the upstream machine having a change from down mode t to down mode j. The expression $z_{x,l}^d$ represents the probability that the downstream machine has a change from down mode x to down mode l. Since this chapter will assume that there are no failure mode changes, all z's will be defined to be zero.

The expressions for idleness failures is represented by the letter q. Thus, q_j^u is the probability that the upstream machine has an idleness failure, and went into failure mode j, Δ_j^u.

Defining $\alpha^*(t)$ as the state (up state or down state) of a machine $*$ at time t (where $*$ is either upstream or downstream), then we can define r, p, q, and z as

$$r_{j,i}^u = prob(\alpha^u(t+1) = \Upsilon_i^u \mid \alpha^u(t) = \Delta_j^u)$$

$$r_{l,f}^d = prob(\alpha^d(t+1) = \Upsilon_f^d \mid \alpha^d(t) = \Delta_l^d)$$

$$p_{i,j}^u = prob(\alpha^u(t+1) = \Delta_j^u \mid \alpha^u(t) = \Upsilon_i^u \text{ and } n(t) < N)$$

$$p_{f,l}^d = prob(\alpha^d(t+1) = \Delta_l^d \mid \alpha^d(t) = \Upsilon_f^d \text{ and } n(t) > 0)$$

$$q_{i,j}^u = prob(\alpha^u(t+1) = \Delta_j^u \mid \alpha^u(t) = \Upsilon_i^u \text{ and } n(t) = N)$$

$$q^d_{f,l} = prob(\alpha^d(t+1) = \Delta^d_l \mid \alpha^d(t) = \Upsilon^d_f \text{ and } n(t) = 0)$$

$$z^u_{j,j'} = prob(\alpha^u(t+1) = \Delta^u_{j'} \mid \alpha^u(t) = \Delta^u_j)$$

$$z^d_{l,l'} = prob(\alpha^d(t+1) = \Delta^d_{l'} \mid \alpha^d(t) = \Delta^d_l)$$

For any given down mode, $\tilde{r}^u_j$ for the upstream machine, or $\tilde{r}^d_l$ for the downstream machine, will represent the probability of *leaving* a down mode. Because a down state can be left via repair or failure change, $\tilde{r}^u_j$ is the sum of the probabilities of all such possible events.

$$\tilde{r}^u_j = r^u_j + \sum_{t=j+1}^{J} z^u_{j,t}$$

and

$$\tilde{r}^d_l = r^d_l + \sum_{x=1}^{l-1} z^d_{l,x}$$

Since all z's are zero, $\tilde{r}$ reduces to r.

Similar to Tolio's method [12], P^u and P^d are

$$P^u = \sum_{j=1}^{J} p^u_j$$

and

$$P^d = \sum_{l=1}^{L} p^d_l$$

where J and L are the total number of failure modes for the upstream machine and downstream machine respectively.

The set of parameters p^u_j and p^d_l must be such that $P^u < 1$ and $P^d < 1$.

We define

$$Q^u = \sum_{j=1}^{J} q_j^u$$

and

$$Q^d = \sum_{l=1}^{L} q_l^d$$

and again, the set of parameters q_j^u and q_l^d must be such that $Q^u < 1$ and $Q^d < 1$.

3.3 Performance Measures

The performance measures for the two machine lines are

$$E^u = \sum_{n=0}^{N-1} \left[p(n, \Upsilon^u, \Upsilon^d) + \sum_{l=1}^{L} p(n, \Upsilon^u, \Delta_l^d) \right]$$

and

$$E^d = \sum_{n=1}^{N} \left[p(n, \Upsilon^u, \Upsilon^d) + \sum_{j=1}^{J} p(n, \Delta_j^u, \Upsilon^d) \right]$$

where E^u is the throughput of the upstream machine, and E^d is the throughput of the downstream machine. The solution technique must yield $E^u = E^d$

The average buffer level is given by

$$\overline{n} = \sum_{n=0}^{N} \sum_{j=1}^{J} \sum_{l=1}^{L} n \left[p(n, \Delta_j^u, \Delta_l^d) + p(n, \Upsilon^u, \Delta_l^d) + p(n, \Delta_j^u, \Upsilon^d) + p(n, \Upsilon^u, \Upsilon^d) \right]$$

3.4 Internal State Space

The *state* of a two-machine-line is defined as the set of parameters describing the the buffer level, and machine state at a given point of time. For example, a state at time t of $L(5,1)$ could be that $M^u(5,1)$ is up, $M^d(5,1)$ is down in failure mode 3, and $B(5,1)$ has 15 parts in it. The goal of the two-machine-line analysis is to find the probability of all possible states in a two machine line. There are three classifications for states: internal states, boundary states, and transient states. A *transient state* is on where, if the state changes, the two-machine line may never return again. Therefore, in the steady-state, the probability of a two-machine-line being on a transient state is zero.

An *internal state* is defined as one whose buffer level is $2 < n < N-2$. The set of transition equations for the internal states is defined by

$$\begin{aligned} p(n,\Delta_j^u,\Delta_l^d) &= p(n,\Delta_j^u,\Delta_l^d)(1-r_j^u)(1-r_l^d) \\ &\quad +p(n,\Delta_j^u,\Upsilon^d)(1-r_j^u)p_l^d \\ &\quad +p(n,\Upsilon^u,\Delta_l^d)(1-r_l^d)p_j^u \\ &\quad +p(n,\Upsilon^u,\Upsilon^d)p_j^u p_l^d \end{aligned} \tag{3.1}$$

$$\begin{aligned} p(n,\Delta_j^u,\Upsilon^d) &= \sum_{l=1}^{L} p(n+1,\Delta_j^u,\Delta_l^d)(1-r_j^u)r_l^d \\ &\quad +p(n+1,\Delta_j^u,\Upsilon^d)(1-r_j^u)(1-P^d) \\ &\quad +\sum_{l=1}^{L} p(n+1,\Upsilon^u,\Delta_l^d)p_j^u r_l^d \\ &\quad +p(n+1,\Upsilon^u,\Upsilon^d)p_j^u(1-P^d) \end{aligned} \tag{3.2}$$

$$\begin{aligned} p(n,\Upsilon^u,\Delta_l^d) &= \sum_{j=1}^{J} p(n-1,\Delta_j^u,\Delta_l^d)r_j^u(1-r_l^d) \\ &\quad +\sum_{j=1}^{J} p(n-1,\Delta_j^u,\Upsilon^d)r_j^u p_l^d \end{aligned}$$

$$
\begin{aligned}
&+p(n-1,\Upsilon^u,\Delta_l^d)(1-P^u)(1-r_l^d)\\
&+p(n-1,\Upsilon^u,\Upsilon^d)(1-P^u)p_l^d
\end{aligned}
\qquad (3.3)
$$

$$
\begin{aligned}
p(n,\Upsilon^u,\Upsilon^d) \;=\; & \sum_{j=1}^{J}\sum_{l=1}^{L} p(n,\Delta_j^u,\Delta_l^d)r_j^u r_l^d\\
&+\sum_{j=1}^{J} p(n,\Delta_j^u,\Upsilon^d)r_j^u(1-P^d)\\
&+\sum_{l=1}^{L} p(n,\Upsilon^u,\Delta_l^d)(1-P^u)r_l^d\\
&+p(n,\Upsilon^u,\Upsilon^d)(1-P^u)(1-P^d)
\end{aligned}
\qquad (3.4)
$$

To solve, a guess is taken by making the internal steady state probabilities assume the form

$$
\begin{aligned}
p(n,\Delta_j^u,\Delta_l^d) &= X^nU_jD_l\\
p(n,\Delta_j^u,\Upsilon^d) &= X^nU_j\\
p(n,\Upsilon^u,\Delta_l^d) &= X^nD_l\\
p(n,\Upsilon^u,\Upsilon^d) &= X^n\\
& j=1,...,J\\
& l=1,...,L\\
& 2\le n\le N-2
\end{aligned}
\qquad (3.5)
$$

where X, U_j, and D_l are $1+J+L$ constants to be evaluated. This solution structure is the one used by Tolio, and is expected to be appropriate for this adapted model.

By substituting (3.5) into (3.1)-(3.4)

$$
X^nU_jD_l \;=\; X^nU_jD_l(1-r_j^u)(1-r_l^d)
$$

$$
\begin{aligned}
&+X^n U_j (1-r_j^u) p_l^d \\
&+X^n D_l p_j^u (1-r_l^d) \\
&+X^n p_j^u p_l^d
\end{aligned} \tag{3.6}
$$

$$
\begin{aligned}
X^n U_j \;=\;& X^{n+1} p_j^u (1-P^d) \\
&+X^{n+1} U_j (1-r_j^u)(1-P^d) \\
&+\sum_{l=1}^{L} X^{n+1} D_l p_j^u r_l^d \\
&+\sum_{l=1}^{L} X^{n+1} U_j D_l (1-r_j^u) r_l^d
\end{aligned} \tag{3.7}
$$

$$
\begin{aligned}
X^n D_l \;=\;& X^{n-1} (1-P^u) p_l^d \\
&+\sum_{j=1}^{J} X^{n-1} U_j D_l r_j^u (1-r_l^d) \\
&+\sum_{j=1}^{J} X^{n-1} U_j r_j^u p_l^d \\
&+X^{n-1} D_l (1-P^u(1-r_l^d)
\end{aligned} \tag{3.8}
$$

$$
\begin{aligned}
X^n \;=\;& X^n (1-P^u)(1-P^d) \\
&+\sum_{j=1}^{J} X^n U_j r_j^u (1-P^d) \\
&+\sum_{l=1}^{L} X^n D_l (1-P^u) r_l^d \\
&+\sum_{j=1}^{J} \sum_{l=1}^{L} X^n U_j D_l r_j^u r_l^d
\end{aligned} \tag{3.9}
$$

After much simplification (3.6)-(3.9) reduce to

$$
U_j D_l \;=\; \left[U_j (1-r_j^u) + p_j^u \right] \left[D_l (1-r_l^d) + p_l^d \right] \tag{3.10}
$$

$$\frac{U_j}{X} = \left[1 - P^d + \sum_{l=1}^{L} D_l r_l^d\right] \left[U_j(1 - \tilde{r}_j^u) + p_j^u\right] \tag{3.11}$$

$$XD_l = \left[1 - P^u + \sum_{j=1}^{J} U_j r_j^u\right] \left[D_l(1 - r_l^d) + p_l^d\right] \tag{3.12}$$

$$1 = \left[1 - P^u + \sum_{j=1}^{J} U_j r_j^u\right] \left[1 - P^d + \sum_{l=1}^{L} D_l r_l^d\right] \tag{3.13}$$

Rearranging terms in (3.10)

$$1 = \left[\frac{U_j(1 - r_j^u) + p_j^u}{U_j}\right] \left[\frac{D_l(1 - r_l^d) + p_l^d}{D_l}\right]$$
$$\text{j = 1, ..., J; l = 1, ..., L.}$$

which means that for some constant K

$$\left[\frac{U_j(1 - r_j^u) + p_j^u}{U_j}\right] = K, \text{ j=1,...,J.}$$

and

$$\left[\frac{D_l(1 - r_l^d) + p_l^d}{D_l}\right] = \frac{1}{K}, \text{ l=1,...,L.}$$

Consequently,

$$U_j = \frac{p_j^u}{K - 1 + r_j^u}, \text{ j=1,...,J.} \tag{3.14}$$

and

$$D_l = \frac{p_l^d}{\frac{1}{K} - 1 + \tilde{r}_l^d}, \text{ l=1,...,L.} \tag{3.15}$$

By introducing (3.14) and (3.15) in (3.13)

$$1 = \left[1 - P^u + \sum_{j=1}^{J} \frac{p_j^u r_j^u}{K - 1 + r_j^u}\right] \left[1 - P^d + \sum_{l=1}^{L} \frac{p_l^d r_l^d}{\frac{1}{K} - 1 + r_l^d}\right] \quad (3.16)$$

This is not $R = JL$ order polynomial in K. Defining K_m to be the m^{th} root of the polynomial, the values of $U_{j,m,}$, $D_{l,m}$, and X_m can be found.

Using (3.12), (3.14), and (3.15), the equations needed for $U_{j,m,}$, $D_{l,m}$, and X_m are found to be

$$X_m = \left[1 - P^u + \sum_{j=1}^{J} \frac{p_j^u r_j^u}{K_m - 1 + r_j^u}\right] \frac{1}{K_m} \quad (3.17)$$
$$m = 1, ..., J + L.$$

$$U_{j,m} = \frac{p_j^u}{K_m - 1 + r_j^u}, \text{ j=1,...,J.} \quad (3.18)$$

$$D_{l,m} = \frac{p_l^d}{\frac{1}{K_m} - 1 + r_l^d}, \text{ l=1,...,L.} \quad (3.19)$$

Efficient Root Search

The key to solving the internal state solutions is to efficiently find all the roots of equation (3.16). Defining $F(K)$ as:

$$F(K) = \left[1 - P^u + \sum_{j=1}^{J} \frac{p_j^u r_j^u}{K - 1 + r_j^u}\right] \left[1 - P^d + \sum_{l=1}^{L} \frac{p_l^d r_l^d}{\frac{1}{K} - 1 + r_l^d}\right] - 1 \quad (3.20)$$

then root-finding is equivalent as finding the Ks that make $F(K) = 0$. $F(K)$ has poles because its denominator can go to 0 for some K. For upstream failure modes, a pole is encountered when

$$K = 1 - r_j^u$$

whereas for downstream modes, poles are encountered when

$$K = \frac{1}{1 - r_l^d}$$

Because $0 < r_j^u < 1$ and $0 < r_l^d < 1$, then it must be the case that the only way there would be repeated poles is if there is a repeated r among either the upstream modes, or the downstream modes. However, no pair of r_j^u and r_l^d would ever yield a repeated pole.

Since $0 < r_j^u < 1$ and $0 < r_l^d < 1$, then all poles caused by upstream modes must be at $0 < K < 1$; whereas all poles caused by downstream modes must be at $1 < K$.

The key usefulness of such insight comes from realizing that, apart from $K = 1$, for every pole, there is a root, and repeated poles imply repeated roots. In fact, apart from $K = 1$, there must be only one root between every pair of poles. This is because the the line going from one pole to the next crosses the zero axis only once. With such information available, one is able to limit the search space for the roots, knowing exactly how many roots are repeated,

A typical example of the graph of $F(K)$ is shown in Figure 3-2. This figure shows the expected behavior of roots being between poles. Simple search algorithms can be devised to find the roots by scanning between between poles.

Dealing with Repeated Roots

Dealing with repeated roots is a sensitive issue because in long lines, it is often seen that machines have repeated repair probabilities. From a two-machine line perspective, dealing with repeated roots is done by realizing that a machine that has two different failure modes with the same repair probability (different failure probability), those failure modes can be considered the exact same failure type. In other words,

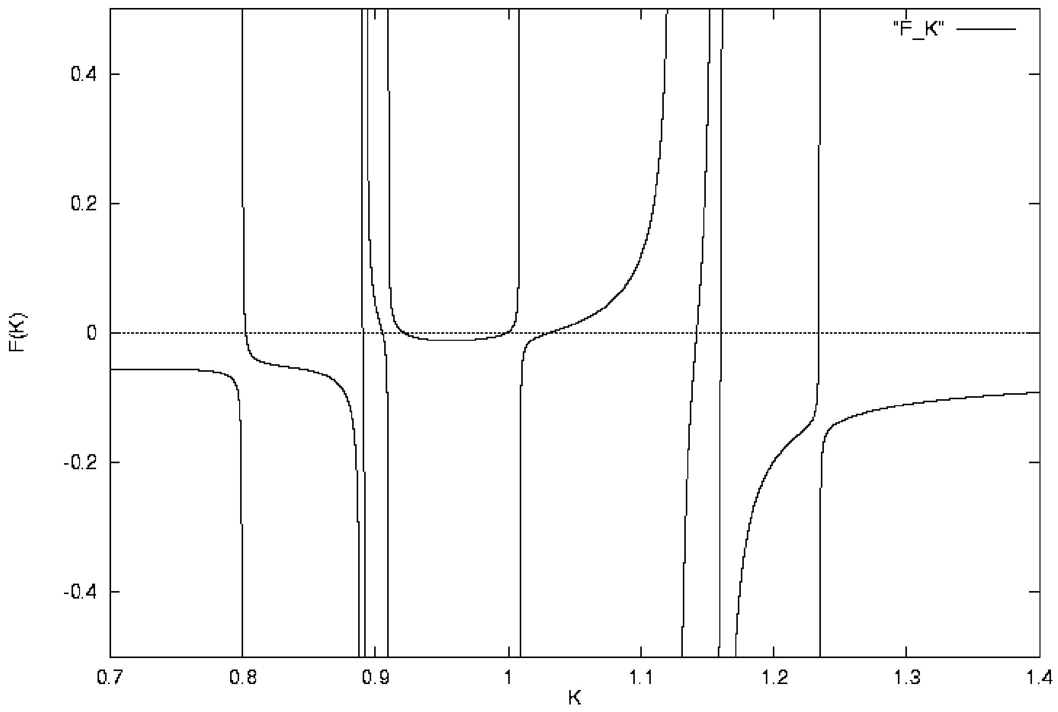

Figure 3-2: F(K) for a two-machine line with nine down modes.

two failure modes that have the exact same repair characteristics are the same failure mode. Thus, combining such modes into one single mode, but with a failure probability equal to the addition of all the initial failure probabilities, takes care of the problem.

The pooling of modes in the two-machine lines, however, creates a potential problem for the long-line decomposition analysis. This is because the decomposition needs to receive from the two-machine lines the probability of starvation and blockage due to *each specific failure mode*. Thus, by pooling the probabilities, it is not clear how such probability can be decomposed. The method we propose to decompose is to take the weighted fraction of the desired quantity in terms of the pooled probabilities of failure. For example, if downstream modes a, b, and c have the same r and were pooled into failure mode m; since

$$r_m^d = r_a^d = r_b^d = r_c^d$$

and

$$p_m^d = p_a^d + p_b^d + p_c^d$$

then, after solving the two machine line, we can decompose P_{sm}, the probability of starvation due to mode m, as

$$P_{s(a)} = P_{sm}\frac{p_a^d}{p_m^d}$$
$$P_{s(b)} = P_{sm}\frac{p_b^d}{p_m^d}$$
$$P_{s(c)} = P_{sm}\frac{p_c^d}{p_m^d}$$

3.5 Boundary States

Because of idleness failures, the states considered transient are fewer in number than in the original Tolio decomposition version. The states that remain transient are $p(0, \Upsilon^u, \Upsilon^d)$, $p(0, \Upsilon^u, \Delta_l^d)$, $p(N, \Upsilon^u, \Upsilon^d)$, and $p(N, \Delta_j^u, \Upsilon^d)$.

The transition equations for the boundary state probabilities are

$$\begin{aligned} p(0, \Delta_j^u, \Delta_l^d) &= p(0, \Delta_j^u, \Delta_l^d)(1 - r_j^u)(1 - r_l^d) \\ &\quad + p(0, \Delta_j^u, \Upsilon^d)(1 - r_j^u) q_l^d \end{aligned} \tag{3.21}$$

$$\begin{aligned} p(0, \Delta_j^u, \Upsilon^d) &= \sum_{l=1}^{L} p(0, \Delta_j^u, \Delta_l^d)(1 - r_j^u) r_l^d \\ &\quad + \sum_{l=1}^{L} p(1, \Delta_j^u, \Delta_l^d)(1 - r_j^u) r_l^d \\ &\quad + p(0, \Delta_j^u, \Upsilon^d)(1 - r_j^u)(1 - Q^d) \\ &\quad + p(1, \Delta_j^u, \Upsilon^d)(1 - r_j^u)(1 - P^d) \\ &\quad + \sum_{l=1}^{L} p(1, \Upsilon^u, \Delta_l^d) p_j^u r_l^d \\ &\quad + p(1, \Upsilon^u, \Upsilon^d) p_j^u (1 - P^d) \end{aligned} \tag{3.22}$$

$$p(0, \Upsilon^u, \Delta_l^d) = 0 \tag{3.23}$$

$$p(0, \Upsilon^u, \Upsilon^d) = 0 \tag{3.24}$$

$$\begin{aligned} p(1, \Delta_j^u, \Delta_l^d) &= p(1, \Delta_j^u, \Delta_l^d)(1 - r_j^u)(1 - r_l^d) \\ &\quad + p(1, \Delta_j^u, \Upsilon^d)(1 - r_j^u) p_l^d \\ &\quad + p(1, \Upsilon^u, \Delta_l^d) p_j^u (1 - r_l^d) \end{aligned}$$

$$+p(1,\Upsilon^u,\Upsilon^d)p^u_j p^d_l \tag{3.25}$$

$$\begin{aligned}
p(1,\Delta^u_j,\Upsilon^d) &= \sum_{l=1}^{L} p(2,\Delta^u_j,\Delta^d_l)(1-r^u_j)r^d_l \\
&+p(2,\Delta^u_j,\Upsilon^d)(1-r^u_j)(1-P^d) \\
&+\sum_{l=1}^{L} p(2,\Upsilon^u,\Delta^d_l)p^u_j r^d_l \\
&+p(2,\Upsilon^u,\Upsilon^d)p^u_j(1-P^d)
\end{aligned} \tag{3.26}$$

$$\begin{aligned}
p(1,\Upsilon^u,\Delta^d_l) &= \sum_{j=1}^{J} p(0,\Delta^u_j,\Delta^d_l)r^u_j(1-r^d_l) \\
&+\sum_{j=1}^{J} p(0,\Delta^u_j,\Upsilon^d)r^u_j q^d_l
\end{aligned} \tag{3.27}$$

$$\begin{aligned}
p(1,\Upsilon^u,\Upsilon^d) &= \sum_{j=1}^{J}\sum_{l=1}^{L} p(0,\Delta^u_j,\Delta^d_l)r^u_j r^d_l \\
&+\sum_{j=1}^{J}\sum_{l=1}^{L} p(1,\Delta^u_j,\Delta^d_l)r^u_j r^d_l \\
&+\sum_{j=1}^{J} p(0,\Delta^u_j,\Upsilon^d)r^u_j(1-Q^d) \\
&+\sum_{j=1}^{J} p(1,\Delta^u_j,\Upsilon^d)r^u_j(1-P^d) \\
&+\sum_{l=1}^{L} p(1,\Upsilon^u,\Delta^d_l)(1-P^u)r^d_l \\
&+p(1,\Upsilon^u,\Upsilon^d)(1-P^u)(1-P^d)
\end{aligned} \tag{3.28}$$

$$\begin{aligned}
p(2,\Upsilon^u,\Delta^d_l) &= \sum_{j=1}^{J} p(1,\Delta^u_j,\Delta^d_l)r^u_j(1-r^d_l) \\
&+\sum_{j=1}^{J} p(1,\Delta^u_j,\Upsilon^d)r^u_j p^d_l \\
&+p(1,\Upsilon^u,\Delta^d_l)(1-P^u)(1-r^d_l) \\
&+p(1,\Upsilon^u,\Upsilon^d)(1-P^u)p^d_l
\end{aligned} \tag{3.29}$$

$$\begin{aligned} p(N-2,\Delta_j^u,\Upsilon^d) &= \sum_{l=1}^{L} p(N-1,\Delta_j^u,\Delta_l^d)(1-r_j^u)r_l^d \\ &\quad + p(N-1,\Delta_j^u,\Upsilon^d)(1-r_j^u)(1-P^d) \\ &\quad + \sum_{l=1}^{L} p(N-1,\Upsilon^u,\Delta_l^d)p_j^u r_l^d \\ &\quad + p(N-1,\Upsilon^u,\Upsilon^d)p_j^u(1-P^d) \end{aligned} \tag{3.30}$$

$$\begin{aligned} p(N-1,\Delta_j^u,\Delta_l^d) &= p(N-1,\Delta_j^u,\Delta_l^d)(1-r_j^u)(1-r_l^d) \\ &\quad + p(N-1,\Delta_j^u,\Upsilon^d)(1-r_j^u)p_l^d \\ &\quad + p(N-1,\Upsilon^u,\Delta_l^d)p_j^u(1-r_l^d) \\ &\quad + p(N-1,\Upsilon^u,\Upsilon^d)p_j^u p_l^d \end{aligned} \tag{3.31}$$

$$\begin{aligned} p(N-1,\Delta_j^u,\Upsilon^d) &= \sum_{l=1}^{L} p(N,\Delta_j^u,\Delta_l^d)(1-r_j^u)r_l^d \\ &\quad + \sum_{l=1}^{L} p(N,\Upsilon^u,\Delta_l^d)q_j^u r_l^d \end{aligned} \tag{3.32}$$

$$\begin{aligned} p(N-1,\Upsilon^u,\Delta_l^d) &= \sum_{j=1}^{J} p(N-2,\Delta_j^u,\Delta_l^d)r_j^u(1-r_l^d) \\ &\quad + \sum_{j=1}^{J} p(N-2,\Delta_j^u,\Upsilon^d)r_j^u p_l^d \\ &\quad + p(N-2,\Upsilon^u,\Delta_l^d)(1-P^u)(1-r_l^d) \\ &\quad + p(N-2,\Upsilon^u,\Upsilon^d)(1-P^u)p_l^d \end{aligned} \tag{3.33}$$

$$\begin{aligned} p(N-1,\Upsilon^u,\Upsilon^d) &= \sum_{j=1}^{J}\sum_{l=1}^{L} p(N-1,\Delta_j^u,\Delta_l^d)r_j^u r_l^d \\ &\quad + \sum_{l=1}^{L} p(N-1,\Upsilon^u,\Delta_l^d)(1-P^u)r_l^d \\ &\quad + \sum_{l=1}^{L} p(N,\Upsilon^u,\Delta_l^d)(1-Q^u)r_l^d \end{aligned}$$

$$+p(N-1,\Upsilon^u,\Upsilon^d)(1-P^u)(1-P^d) \tag{3.34}$$

$$\begin{aligned} p(N,\Delta_j^u,\Delta_l^d) &= p(N,\Delta_j^u,\Delta_l^d)(1-r_j^u)(1-r_l^d) \\ &\quad +p(N,\Upsilon^u,\Delta_l^d)q_j^u(1-r_l^d) \end{aligned} \tag{3.35}$$

$$p(N,\Delta_j^u,\Upsilon^d) = 0 \tag{3.36}$$

$$\begin{aligned} p(N,\Upsilon^u,\Delta_l^d) &= \sum_{j=1}^{J} p(N-1,\Delta_j^u,\Delta_l^d)r_l^u(1-r_l^u) \\ &\quad +\sum_{j=1}^{J} p(N,\Delta_j^u,\Delta_l^d)r_j^u(1-r_l^d) \\ &\quad +\sum_{j=1}^{J} p(N-1,\Delta_j^u,\Upsilon^d)r_j^u p_l^d \\ &\quad +p(N-1,\Upsilon^u,\Delta_l^d)(1-P^u)(1-r_l^d) \\ &\quad +p(N,\Upsilon^u,\Delta_l^d)(1-Q^u)(1-r_l^d) \\ &\quad +p(N-1,\Upsilon^u,\Upsilon^d)(1-P^u)p_l^d \end{aligned} \tag{3.37}$$

$$p(N,\Upsilon^u,\Upsilon^d) = 0 \tag{3.38}$$

3.5.1 Solution Technique for Boundary Equations

In trying to simplify the lower boundary solutions via the solutions found by Tolio, we propose that $P(state) = P_T(state) + F(state)$, where $P_T(state)$ is Tolio's solution for the state [12], and $F(state)$ is an unknown. Because $p(0,\Delta_j^u,\Delta_l^d)$ and $p(1,\Upsilon^u,\Delta_l^d)$ were transient states in the original Tolio work, $P_T(state) = 0$. In addition, the assumption that $F(state) = 0$ will be made for all states that Tolio found to have an internal form probability. Thus, the solutions for the non-transient lower boundary states are

$$p(0, \Delta_j^u, \Delta_l^d) = \sum_{m=1}^{R} C_m F_m(0, \Delta_j^u, \Delta_l^d) \tag{3.39}$$

$$p(1, \Upsilon^u, \Delta_l^d) = \sum_{m=1}^{R} C_m F_m(1, \Upsilon^u, \Delta_l^d) \tag{3.40}$$

$$p(1, \Delta_j^u, \Upsilon^d) = \sum_{m=1}^{R} C_m X_m U_{j,m} \tag{3.41}$$

$$p(1, \Delta_j^u, \Delta_l^d) = \sum_{m=1}^{R} C_m X_m U_{j,m} D_{l,m} \tag{3.42}$$

$$p(2, \Upsilon^u, \Delta_l^d) = \sum_{m=1}^{R} C_m X_m^2 D_{l,m} \tag{3.43}$$

and

$$\begin{aligned} p(0, \Delta_j^u, \Upsilon^d) = & \sum_{m=1}^{R} C_m \frac{1 - r_j^u}{r_j^u} \frac{U_{j,m}}{K_m} \\ & + \frac{p_j^u}{p_l^d r_j^u}(1 - P^d) \sum_{m=1}^{R} C_m X_m \frac{D_{l,m}}{K_m} \\ & + \sum_{m=1}^{R} C_m F_m(0, \Delta_j^u, \Upsilon^d) \end{aligned} \tag{3.44}$$

$$\begin{aligned} p(1, \Upsilon^u, \Upsilon^d) = & \sum_{m=1}^{R} C_m \frac{1}{p_l^d} X_m \frac{D_{l,m}}{K_m} \\ & + \sum_{m=1}^{R} C_m F_m(1, \Upsilon^u, \Upsilon^d) \end{aligned} \tag{3.45}$$

We know the $P_T(state)$s from Tolio's decomposition, however, the $F_m(state)$s are unknown. In total, there are RJ unknown $F_m(0, \Delta_j^u, \Upsilon^d)$ terms (m=1,..,R; j=1,...,J), and R unknown $F_m(1, \Upsilon^u, \Upsilon^d)$ terms. In order to find the $R(J+1)$ unknowns, a set of

$R(J+1)$ equations must be found. The derivation of the required equations follows.

Working with the equations for $p(0,\Delta_j^u,\Delta_l^d)$ and $p(1,\Upsilon^u,\Delta_l^d)$ from (3.21) and (3.47),

$$\begin{aligned}
p(0,\Delta_j^u,\Delta_l^d) &= p(0,\Delta_j^u,\Delta_l^d)(1-r_j^u)(1-r_l^d) \\
&\quad + p(0,\Delta_j^u,\Upsilon^d)(1-r_j^u)q_l^d \\
&= \frac{p(0,\Delta_j^u,\Upsilon^d)(1-r_j^u)q_l^d}{\left[1-(1-r_j^u)(1-r_l^d)\right]} \\
p(1,\Upsilon^u,\Delta_l^d) &= \sum_{j'=1}^{J} p(0,\Delta_{j'}^u,\Delta_l^d) r_{j'}^u (1-r_l^d) \\
&\quad + \sum_{j'=1}^{J} p(0,\Delta_{j'}^u,\Upsilon^d) r_{j'}^u q_l^d \\
&= \sum_{j'=1}^{J} \frac{r_{j'}^u (1-r_{j'}^u)(1-r_l^d) q_l^d p(0,\Delta_{j'}^u,\Upsilon^d)}{\left[1-(1-r_{j'}^u)(1-r_l^d)\right]} \\
&\quad + \sum_{j'=1}^{J} p(0,\Delta_{j'}^u,\Upsilon^d) r_{j'}^u q_l^d \\
&= \sum_{j'=1}^{J} \frac{q_l^d r_{j'}^u p(0,\Delta_{j'}^u,\Upsilon^d)}{\left[1-(1-r_{j'}^u)(1-r_l^d)\right]}
\end{aligned}$$

Substituting the assumed solution form for $p(0,\Delta_j^u,\Upsilon^d)$ from (3.45),

$$\begin{aligned}
&p(0,\Delta_j^u,\Delta_l^d) = \\
&\quad \frac{(1-r^u)^2 q_l^d}{r_j^u\left[1-(1-r_j^u)(1-r_l^d)\right]} \sum_{m=1}^{R} C_m \frac{U_{j,m}}{K_m} \\
&\quad + \frac{(1-r^u) q_l^d p_j^u (1-P^d)}{p_l^d r_j^u \left[1-(1-r_j^u)(1-r_l^d)\right]} \sum_{m=1}^{R} C_m X_m \frac{D_{l,m}}{K_m}
\end{aligned}$$

$$+\frac{(1-r^u)q_l^d}{\left[1-(1-r_j^u)(1-r_l^d)\right]}\sum_{m=1}^{R} C_m F_m(0,\Delta_j^u,\Upsilon^d) \tag{3.46}$$

and

$$\begin{aligned}
&p(1,\Upsilon^u,\Delta_l^d) = \\
&\quad \sum_{j'=1}^{J} \frac{q_l^d(1-r_{j'}^u)}{\left[1-(1-r_{j'}^u)(1-r_l^d)\right]}\sum_{m=1}^{R} C_m \frac{U_{j',m}}{K_m} \\
&\quad +\sum_{j'=1}^{J} \frac{q_l^d p_{j'}^u(1-P^d)}{p_l^d\left[1-(1-r_{j'}^u)(1-r_l^d)\right]}\sum_{m=1}^{R} C_m X_m \frac{D_{l,m}}{K_m} \\
&\quad +\sum_{j'=1}^{J} \frac{q_l^d r_{j'}^u}{\left[1-(1-r_{j'}^u)(1-r_l^d)\right]}\sum_{m=1}^{R} C_m F_m(0,\Delta_{j'}^u,\Upsilon^d)
\end{aligned} \tag{3.47}$$

Remembering that

$$\begin{aligned}
p(2,\Upsilon^u,\Delta_l^d) &= \sum_{j=1}^{J} p(1,\Delta_j^u,\Delta_l^d) r_j^u (1-r_l^d) \\
&\quad +\sum_{j=1}^{J} p(1,\Delta_j^u,\Upsilon^d) r_j^u p_l^d \\
&\quad +p(1,\Upsilon^u,\Upsilon^d)(1-P^u)p_l^d \\
&\quad +p(1,\Upsilon^u,\Delta_l^d)(1-P^u)(1-r_l^d)
\end{aligned}$$

and P_T for such state (as derived by Tolio, and discussed in Appendix A)

$$P_T(2,\Upsilon^u,\Delta_l^d) = \sum_{m=1}^{R} C_m X_m^2 D_{l,m}$$

one can substitute and rearrange terms to find

$$\begin{aligned}
P_T(2, \Upsilon^u, \Delta_l^d) &= P_T(1, \Delta_j^u, \Delta_l^d) r_j^u (1 - r_l^d) \\
&\quad + P_T(1, \Delta_j^u, \Upsilon^d) r_j^u p_l^d \\
&\quad + P_T(1, \Upsilon^u, \Upsilon^d)(1 - P^u) p_l^d \\
&\quad + \sum_{m=1}^{R} C_m F_m(1, \Upsilon^u, \Upsilon^d)(1 - P^u) p_l^d \\
&\quad + p(1, \Upsilon^u, \Delta_l^d)(1 - P^u)(1 - r_l^d)
\end{aligned}$$

All the P_T's cancel out. Therefore, it must be the case that

$$\begin{aligned}
0 &= p(1, \Upsilon^u, \Delta^d)(1 - P^u)(1 - r_l^d) \\
&\quad + \sum_{m=1}^{R} C_m F_m(1, \Upsilon^u, \Upsilon^d)(1 - P^u) p_l^d
\end{aligned} \tag{3.48}$$

where $p(1, \Upsilon^u, \Delta_l^d)$ is a function of $p(0, \Delta_j^u, \Upsilon^d)$ as found in (3.47). Therefore,

$$\begin{aligned}
0 &= \sum_{j'=1}^{J} \frac{q_l^d (1 - r_{j'}^u)(1 - P^u)(1 - r_l^d)}{\left[1 - (1 - r_{j'}^u)(1 - r_l^d)\right]} \sum_{m=1}^{R} C_m \frac{U_{j',m}}{K_m} \\
&\quad + \sum_{j'=1}^{J} \frac{q_l^d p_{j'}^u (1 - P^d)(1 - P^u)(1 - r_l^d)}{p_l^d \left[1 - (1 - r_{j'}^u)(1 - r_l^d)\right]} \sum_{m=1}^{R} C_m X_m \frac{D_{l,m}}{K_m} \\
&\quad + \sum_{j'=1}^{J} \frac{q_l^d r_{j'}^u (1 - P^u)(1 - r_l^d)}{\left[1 - (1 - r_{j'}^u)(1 - r_l^d)\right]} \sum_{m=1}^{R} C_m F_m(0, \Delta_{j'}^u, \Upsilon^d) \\
&\quad + \sum_{m=1}^{R} C_m F_m(1, \Upsilon^u, \Upsilon^d)(1 - P^u) p_l^d
\end{aligned} \tag{3.49}$$

Following a similar process for $p(1, \Delta_j^u, \Delta_l^d)$,

$$
\begin{aligned}
p(1,\Delta_j^u,\Delta_l^d) &= \sum_{m=1}^{R} C_m X_m U_{j,m} D_{l,m} \\
&= p(1,\Delta_j^u,\Delta_l^d)(1-r_j^u)(1-r_l^d) \\
&\quad + p(1,\Delta_j^u,\Upsilon^d)(1-r_j^u)p_l^d \\
&\quad + p(1,\Upsilon^u,\Delta_l^d)p_j^u(1-r_l^d) \\
&\quad + p(1,\Upsilon^u,\Upsilon^d)p_j^u p_l^d
\end{aligned}
$$

Substituting and rearranging terms,

$$
\begin{aligned}
P_T(1,\Delta_j^u,\Delta_l^d)[1-(1-r_j^u)(1-r_l^d)] &= +P_T(1,\Delta_j^u,\Upsilon^d)(1-r_j^u)p_l^d \\
&\quad + p(1,\Upsilon^u,\Delta_l^d)p_j^u(1-r_l^d) \\
&\quad + P_T(1,\Upsilon^u,\Upsilon^d)p_j^u p_l^d \\
&\quad + \sum_{m=1}^{R} C_m F_m(1,\Upsilon^u,\Upsilon^d)p_j^u p_l^d
\end{aligned}
$$

where once again, all P_T terms cancel out, so

$$
\begin{aligned}
0 &= \frac{p_j^u p_l^d \sum_{m=1}^{R} C_m F_m(1,\Upsilon^u,\Upsilon^d)}{\left[1-(1-r_j^u)(1-r_l^d)\right]} \\
&\quad + \frac{p_j^u(1-r_l^d)p(1,\Upsilon^u,\Delta_l^d)}{\left[1-(1-r_j^u)(1-r_l^d)\right]}
\end{aligned} \tag{3.50}
$$

where $p(1,\Upsilon^u,\Delta_l^d)$, as found in (3.47), is a function of $p(0,\Delta_j^u,\Upsilon^d)$. This result, however, is redundant, as it is the same as equation (3.48) scaled by a constant.

Doing the same procedure on $p(0,\Delta_j^u,\Upsilon^d)$,

$$\begin{aligned}
p(0,\Delta_j^u,\Upsilon^d) &= \sum_{m=1}^{R} C_m \frac{1-r_j^u}{r_j^u}\frac{U_{j,m}}{K_m} \\
&\quad + \frac{p_j^u}{p_l^d r_j^u}(1-P^d)\sum_{m=1}^{R} C_m X_m \frac{D_{l,m}}{K_m} \\
&\quad + \sum_{m=1}^{R} C_m F_m(0,\Delta_j^u,\Upsilon^d) \\
&= \sum_{l=1}^{L} p(0,\Delta_j^u,\Delta_l^d)(1-r_j^u)r_l^d \\
&\quad + \sum_{l=1}^{L} p(1,\Delta_j^u,\Delta_l^d)(1-r_j^u)r_l^d \\
&\quad + p(0,\Delta_j^u,\Upsilon^d)(1-r_j^u)(1-Q^d) \\
&\quad + p(1,\Delta_j^u,\Upsilon^d)(1-r_j^u)(1-P^d) \\
&\quad + \sum_{l=1}^{L} p(1,\Upsilon^u,\Delta_l^d)p_j^u r_l^d \\
&\quad + p(1,\Upsilon^u,\Upsilon^d)p_j^u(1-P^d) \\
&= p(0,\Delta_j^u,\Upsilon^d)(1-r_j^u) \\
&\quad + \sum_{l=1}^{L} p(1,\Delta_j^u,\Delta_l^d)(1-r_j^u)r_l^d \\
&\quad + p(1,\Delta_j^u,\Upsilon^d)(1-r_j^u)(1-P^d) \\
&\quad + p(1,\Upsilon^u,\Upsilon^d)p_j^u(1-P^d) \\
&\quad - p(0,\Delta_j^u,\Upsilon^d)(1-r_j^u)Q^d \\
&\quad + \sum_{l=1}^{L} p(0,\Delta_j^u,\Delta_l^d)(1-r_j^u)r_l^d \\
&\quad + \sum_{l=1}^{L} p(1,\Upsilon^u,\Delta_l^d)p_j^u r_l^d
\end{aligned}$$

Thus,

$$
\begin{aligned}
&\left[P_T(0,\Delta_j^u,\Upsilon^d)+\sum_{m=1}^{R}C_mF_m(0,\Delta_j^u,\Upsilon^d)\right]r_j^u = \\
&\quad \sum_{l=1}^{L}P_T(1,\Delta_j^u,\Delta_l^d)(1-r_j^u)r_l^d \\
&\quad +P_T(1,\Delta_j^u,\Upsilon^d)(1-r_j^u)(1-P^d) \\
&\quad +\left[P_T(1,\Upsilon^u,\Upsilon^d)+\sum_{m=1}^{R}C_mF_m(1,\Upsilon^u,\Upsilon^d)\right]p_j^u(1-P^d) \\
&\quad -p(0,\Delta_j^u,\Upsilon^d)(1-r_j^u)Q^d \\
&\quad +\sum_{l=1}^{L}p(0,\Delta_j^u,\Delta_l^d)(1-r_j^u)r_l^d \\
&\quad +\sum_{l=1}^{L}p(1,\Upsilon^u,\Delta_l^d)p_j^u r_l^d
\end{aligned}
$$

The P_T terms cancel out, therefore

$$
\begin{aligned}
\sum_{m=1}^{R}C_mF_m(0,\Delta_j^u,\Upsilon^d) &= -p(0,\Delta_j^u,\Upsilon^d)\frac{(1-r_j^u)}{r_j^u}Q^d \\
&\quad +\sum_{l=1}^{L}p(0,\Delta_j^u,\Delta_l^d)\frac{(1-r_j^u)}{r_j^u}r_l^d \\
&\quad +\sum_{l=1}^{L}p(1,\Upsilon^u,\Delta_l^d)\frac{p_j^u r_l^d}{r_j^u} \\
&\quad +\sum_{m=1}^{R}C_mF_m(1,\Upsilon^u,\Upsilon^d)\frac{p_j^u}{r_j^u}(1-P^d)
\end{aligned}
$$

Re-substituting for $p(0,\Delta_j^u,\Upsilon^d)$ the assumed solution form,

$$
\sum_{m=1}^{R}C_mF_m(0,\Delta_j^u,\Upsilon^d) = -\frac{(1-r_j^u)^2}{r_j^{u2}}Q^d\sum_{m=1}^{R}C_m\frac{U_{j,m}}{K_m}
$$

$$
\begin{aligned}
&-\frac{(1-r_j^u)}{r_j^{u2}} Q^d \frac{p_j^u}{p_l^d}(1-P^d) \sum_{m=1}^{R} C_m X_m \frac{D_{l,m}}{K_m} \\
&-\frac{(1-r_j^u)}{r_j^u} Q^d \sum_{m=1}^{R} C_m F_m(0,\Delta_j^u,\Upsilon^d) \\
&+\sum_{l=1}^{L} p(0,\Delta_j^u,\Delta_l^d) \frac{(1-r_j^u)}{r_j^u} r_l^d \\
&+\sum_{l=1}^{L} p(1,\Upsilon^u,\Delta_l^d) \frac{p_j^u r_l^d}{r_j^u} \\
&+\sum_{m=1}^{R} C_m F_m(1,\Upsilon^u,\Upsilon^d) \frac{p_j^u}{r_j^u}(1-P^d)
\end{aligned}
$$

Thus,

$$
\begin{aligned}
\sum_{m=1}^{R} & C_m F_m(0,\Delta_j^u,\Upsilon^d)\left[r_j^u+(1-r_j^u)Q^d\right] = \\
= \quad & -\frac{(1-r_j^u)^2 Q^d}{r_j^u} \sum_{m=1}^{R} C_m \frac{U_{j,m}}{K_m} \\
& -(1-r_j^u) Q^d \frac{p_j^u}{p_l^d r_j^u}(1-P^d) \sum_{m=1}^{R} C_m X_m \frac{D_{l,m}}{K_m} \\
& +\sum_{l=1}^{L} p(0,\Delta_j^u,\Delta_l^d)(1-r_j^u) r_l^d \\
& +\sum_{l=1}^{L} p(1,\Upsilon^u,\Delta_l^d) p_j^u r_l^d \\
& +\sum_{m=1}^{R} C_m F_m(1,\Upsilon^u,\Upsilon^d) p_j^u (1-P^d)
\end{aligned}
$$

Substituting in with equations (3.46) and (3.47),

$$
\begin{aligned}
&\sum_{m=1}^{R} C_m F_m(0,\Delta_j^u,\Upsilon^d)\left[r_j^u+(1-r_j^u)Q^d\right] = \\
&\quad -\frac{(1-r_j^u)^2 Q^d}{r_j^u} \sum_{m=1}^{R} C_m \frac{U_{j,m}}{K_m}
\end{aligned}
$$

$$
\begin{aligned}
&-(1-r_j^u)Q^d\frac{p_j^u}{p_l^d r_j^u}(1-P^d)\sum_{m=1}^{R} C_m X_m \frac{D_{l,m}}{K_m}\\
&+\sum_{l=1}^{L}\frac{(1-r_j^u)^3 r_l^d q_l^d}{r_j^u\left[1-(1-r_j^u)(1-r_l^d)\right]}\sum_{m=1}^{R} C_m \frac{U_{j,m}}{K_m}\\
&+\sum_{l=1}^{L}\frac{(1-r^{u_j})^2 q_l^d r_l^d p_j^u(1-P^d)}{p_l^d r_j^u\left[1-(1-r_j^u)(1-r_l^d)\right]}\sum_{m=1}^{R} C_m X_m \frac{D_{l,m}}{K_m}\\
&+\sum_{l=1}^{L}\frac{(1-r_j^u)^2 r_l^d q_l^d}{\left[1-(1-r_j^u)(1-r_l^d)\right]}\sum_{m=1}^{R} C_m F_m(0,\Delta_j^u,\Upsilon^d)\\
&+\sum_{l=1}^{L}\sum_{j'=1}^{J}\frac{q_l^d p_j^u r_l^d(1-r_{j'}^u)}{\left[1-(1-r_{j'}^u)(1-r_l^d)\right]}\sum_{m=1}^{R} C_m \frac{U_{j',m}}{K_m}\\
&+\sum_{l=1}^{L}\sum_{j'=1}^{J}\frac{q_l^d p_{j'}^u p_j^u r_l^d(1-P^d)}{p_l^d\left[1-(1-r_{j'}^u)(1-r_l^d)\right]}\sum_{m=1}^{R} C_m X_m \frac{D_{l,m}}{K_m}\\
&+\sum_{l=1}^{L}\sum_{j'=1}^{J}\frac{q_l^d p_j^u r_l^d r_{j'}^u}{\left[1-(1-r_{j'}^u)(1-r_l^d)\right]}\sum_{m=1}^{R} C_m F_m(0,\Delta_{j'}^u,\Upsilon^d)\\
&+\sum_{m=1}^{R} C_m F_m(1,\Upsilon^u,\Upsilon^d)p_j^u(1-P^d)
\end{aligned}
\tag{3.51}
$$

from Equation (3.49) we see that

$$
\begin{aligned}
&\sum_{m=1}^{R} C_m F_m(1,\Upsilon^u,\Upsilon^d)(1-P^u)p_l^d =\\
&\quad-\sum_{j'=1}^{J}\frac{q_l^d(1-r_{j'}^u)(1-P^u)(1-r_l^d)}{\left[1-(1-r_{j'}^u)(1-r_l^d)\right]}\sum_{m=1}^{R} C_m \frac{U_{j',m}}{K_m}\\
&\quad-\sum_{j'=1}^{J}\frac{q_l^d p_{j'}^u(1-P^d)(1-P^u)(1-r_l^d)}{p_l^d\left[1-(1-r_{j'}^u)(1-r_l^d)\right]}\sum_{m=1}^{R} C_m X_m \frac{D_{l,m}}{K_m}\\
&\quad-\sum_{j'=1}^{J}\frac{q_l^d r_{j'}^u(1-P^u)(1-r_l^d)}{\left[1-(1-r_{j'}^u)(1-r_l^d)\right]}\sum_{m=1}^{R} C_m F_m(0,\Delta_{j'}^u,\Upsilon^d)
\end{aligned}
$$

which is the last term of Equation (3.51) scaled. Substituting,

$$
\begin{aligned}
&\sum_{m=1}^{R} C_m F_m(0,\Delta_j^u,\Upsilon^d)\left[r_j^u + (1-r_j^u)Q^d\right] = \\
&\quad -\frac{(1-r_j^u)^2 Q^d}{r_j^u}\sum_{m=1}^{R} C_m \frac{U_{j,m}}{K_m} \\
&\quad -(1-r_j^u)Q^d \frac{p_j^u}{p_l^d r_j^u}(1-P^d)\sum_{m=1}^{R} C_m X_m \frac{D_{l,m}}{K_m} \\
&\quad +\sum_{l=1}^{L}\frac{(1-r_j^u)^3 r_l^d q_l^d}{r_j^u\left[1-(1-r_j^u)(1-r_l^d)\right]}\sum_{m=1}^{R} C_m \frac{U_{j,m}}{K_m} \\
&\quad +\sum_{l=1}^{L}\frac{(1-r_j^u)^2 q_l^d r_l^d p_j^u (1-P^d)}{p_l^d r_j^u\left[1-(1-r_j^u)(1-r_l^d)\right]}\sum_{m=1}^{R} C_m X_m \frac{D_{l,m}}{K_m} \\
&\quad +\sum_{l=1}^{L}\frac{(1-r_j^u)^2 r_l^d q_l^d}{\left[1-(1-r_j^u)(1-r_l^d)\right]}\sum_{m=1}^{R} C_m F_m(0,\Delta_j^u,\Upsilon^d) \\
&\quad +\sum_{l=1}^{L}\sum_{j'=1}^{J}\frac{q_l^d p_j^u r_l^d (1-r_{j'}^u)}{\left[1-(1-r_{j'}^u)(1-r_l^d)\right]}\sum_{m=1}^{R} C_m \frac{U_{j',m}}{K_m} \\
&\quad +\sum_{l=1}^{L}\sum_{j'=1}^{J}\frac{q_l^d p_{j'}^u p_j^u r_l^d (1-P^d)}{p_l^d\left[1-(1-r_{j'}^u)(1-r_l^d)\right]}\sum_{m=1}^{R} C_m X_m \frac{D_{l,m}}{K_m} \\
&\quad +\sum_{l=1}^{L}\sum_{j'=1}^{J}\frac{q_l^d p_j^u r_l^d r_{j'}^u}{\left[1-(1-r_{j'}^u)(1-r_l^d)\right]}\sum_{m=1}^{R} C_m F_m(0,\Delta_{j'}^u,\Upsilon^d) \\
&\quad -\sum_{j'=1}^{J}\frac{q_l^d (1-r_{j'}^u)(1-P^d)(1-r_l^d) p_j^u}{p_l^d\left[1-(1-r_{j'}^u)(1-r_l^d)\right]}\sum_{m=1}^{R} C_m \frac{U_{j',m}}{K_m} \\
&\quad -\sum_{j'=1}^{J}\frac{q_l^d p_{j'}^u p_j^u (1-P^d)^2 (1-r_l^d)}{p_l^{d2}\left[1-(1-r_{j'}^u)(1-r_l^d)\right]}\sum_{m=1}^{R} C_m X_m \frac{D_{l,m}}{K_m} \\
&\quad -\sum_{j'=1}^{J}\frac{q_l^d r_{j'}^u (1-P^d)(1-r_l^d) p_j^u}{p_l^d\left[1-(1-r_{j'}^u)(1-r_l^d)\right]}\sum_{m=1}^{R} C_m F_m(0,\Delta_{j'}^u,\Upsilon^d)
\end{aligned}
\tag{3.52}
$$

Equation (3.52) must hold true for all m, and thus it sets up a system of RJ equations in RJ unknowns (one $F_m(0,\Delta_j^u,\Upsilon^d)$ for every (j,m) pair). Once all such unknowns are found, they can be plugged into equation (3.49), to find all the $F_m(0,\Upsilon^u,\Upsilon^d)$ (R of them).

A similar process for the upstream equations would yield the same solution process for all $F_m(N, \Upsilon^u, \Delta^d_l)$ and $F_m(N-1, \Upsilon^u, \Upsilon^d)$.

The solution form for the upper boundary internal states is

$$p(N-1, \Delta^u_j, \Upsilon^d) = \sum_{m=1}^{R} C_m F_m(N-1, \Delta^u_j, \Upsilon^d) \tag{3.53}$$

$$p(N, \Delta^u_j, \Delta^d_l) = \sum_{m=1}^{R} C_m F_m(N, \Delta^u_j, \Delta^d_l) \tag{3.54}$$

$$p(N-2, \Delta^u_j, \Upsilon^d) = \sum_{m=1}^{R} C_m X_m^{N-2} U_{j,m} \tag{3.55}$$

$$p(N-1, \Delta^u_j, \Delta^d_l) = \sum_{m=1}^{R} C_m X_m^{N-1} U_{j,m} D_{l,m} \tag{3.56}$$

$$p(N-1, \Upsilon^u, \Delta^d_l) = \sum_{m=1}^{R} C_m X_m^{N-1} D_{l,m} \tag{3.57}$$

and

$$\begin{aligned} p(N-1, \Upsilon^u, \Upsilon^d) &= \frac{1}{p^u_j} \sum_{m=1}^{R} C_m X_m^{N-1} U_{j,m} K_m \\ &\quad + \sum_{m=1}^{R} C_m F_m(N-1, \Upsilon^u, \Upsilon^d) \end{aligned} \tag{3.58}$$

$$\begin{aligned} p(N, \Upsilon^u, \Delta^d_l) &= \frac{(1-r^d_l)}{r^d_l} \sum_{m=1}^{R} C_m X_m^N D_{l,m} K_m \\ &\quad + \frac{p^d_l}{p^u_j r^d_l}(1-P^u) \sum_{m=1}^{R} C_m X^{N-1} U_{j,m} K_m \\ &\quad + \sum_{m=1}^{R} C_m F_m(N, \Upsilon^u, \Delta^d_l) \end{aligned} \tag{3.59}$$

Working with equations (3.32) and (3.35), one can find that

$$\begin{aligned}
p(N,\Delta_j^u,\Delta_l^d) &= p(N,\Delta_j^u,\Delta_l^d)(1-r_j^u)(1-r_l^d) \\
&\quad + p(N,\Upsilon^u,\Delta_l^d)q_j^u(1-r_l^d) \\
&= \frac{q_j^u(1-r_l^d)p(N,\Upsilon^u,\Delta_l^d)}{\left[1-(1-r_j^u)(1-r_l^d)\right]} \\
p(N-1,\Delta_j^u,\Upsilon^d) &= \sum_{l'=1}^{L} p(N,\Delta_j^u,\Delta_{l'}^d)(1-r_j^u)r_{l'}^d \\
&\quad + \sum_{l'=1}^{L} p(N,\Upsilon^u,\Delta_{l'}^d)q_j^u r_{l'}^d \\
&= \sum_{l'=1}^{L} \frac{q_j^u(1-r_j^u)r_{l'}^d(1-r_l^d)p(N,\Upsilon^u,\Delta_{l'}^d)}{\left[1-(1-r_j^u)(1-r_{l'}^d)\right]} \\
&\quad + \sum_{l'=1}^{L} p(N,\Upsilon^u,\Delta_{l'}^d)q_j^u r_{l'}^d \\
&= \sum_{l'=1}^{L} \frac{q_j^u r_{l'}^d p(N,\Upsilon^u,\Delta_{l'}^d)}{\left[1-(1-r_j^u)(1-r_{l'}^d)\right]}
\end{aligned}$$

Substituting (3.59)

$$\begin{aligned}
&p(N,\Delta_j^u,\Delta_l^d) = \\
&\quad \frac{q_j^u(1-r_l^d)^2}{r_l^d\left[1-(1-r_j^u)(1-r_l^d)\right]}\sum_{m=1}^{R} C_m X_m^N D_{l',m}K_m \\
&\quad + \frac{q_j^u p_l^d(1-r_l^d)(1-P^u)}{p_j^u r_l^d\left[1-(1-r_j^u)(1-r_l^d)\right]}\sum_{m=1}^{R} C_m X^{N-1}U_{j,m}K_m \\
&\quad + \frac{q_j^u(1-r_l^d)}{\left[1-(1-r_j^u)(1-r_l^d)\right]}\sum_{m=1}^{R} C_m F_m(N,\Upsilon^u,\Delta_l^d)
\end{aligned} \tag{3.60}$$

$$p(N-1,\Delta_j^u,\Upsilon^d) =$$

$$\sum_{l'=1}^{L} \frac{(1-r_{l'}^d)q_j^u}{\left[1-(1-r_j^u)(1-r_{l'}^d)\right]} \sum_{m=1}^{R} C_m X_m^N D_{l',m} K_m$$
$$+\sum_{l'=1}^{L} \frac{q_j^u p_{l'}^d (1-P^u)}{p_j^u \left[1-(1-r_j^u)(1-r_{l'}^d)\right]} \sum_{m=1}^{R} C_m X^{N-1} U_{j,m} K_m$$
$$+\sum_{l'=1}^{L} \frac{q_j^u r_{l'}^d}{\left[1-(1-r_j^u)(1-r_{l'}^d)\right]} \sum_{m=1}^{R} C_m F_m(N, \Upsilon^u, \Delta_{l'}^d) \tag{3.61}$$

Remembering that

$$\begin{aligned} p(N-2, \Delta_j^u, \Upsilon^d) &= \sum_{l=1}^{L} p(N-1, \Delta_j^u, \Delta_l^d)(1-r_j^u) r_l^d \\ &+ p(N-1, \Delta_j^u, \Upsilon^d)(1-r_j^u)(1-P^d) \\ &+ \sum_{l=1}^{L} p(N-1, \Upsilon^u, \Delta_l^d) p_j^u r_l^d \\ &+ p(N-1, \Upsilon^u, \Upsilon^d) p_j^u (1-P^d) \\ &= \sum_{m=1}^{R} C_m X_m^{N-2} U_{j,m} \end{aligned}$$

one can substitute and rearrange terms to find

$$\begin{aligned} P_T(N-2, \Delta_j^u, \Upsilon^d) &= \sum_{l=1}^{L} P_T(N-1, \Delta_j^u, \Delta_l^d)(1-r_j^u) r_l^d \\ &+ p(N-1, \Delta_j^u, \Upsilon^d)(1-r_j^u)(1-P^d) \\ &+ \sum_{l=1}^{L} P_T(N-1, \Upsilon^u, \Delta_l^d) p_j^u r_l^d \\ &+ P_T(N-1, \Upsilon^u, \Upsilon^d) p_j^u (1-P^d) \\ &+ \sum_{m=1}^{R} C_m F_m(N-1, \Upsilon^u, \Upsilon^d) p_j^u (1-P^d) \end{aligned}$$

All the P_T's cancel out. Therefore, it must be the case that

$$\begin{aligned} 0 &= p(N-1,\Delta_j^u,\Upsilon^d)(1-r_j^u)(1-P^d) \\ &\quad + \sum_{m=1}^{R} C_m F_m(N-1,\Upsilon^u,\Upsilon^d)p_j^u(1-P^d) \end{aligned} \tag{3.62}$$

Where $p(N-1,\Delta_j^u,\Upsilon^d)$ is a function of $p(N,\Upsilon^u,\Delta_l^d)$. Thus, substituting equation (3.61),

$$\begin{aligned} 0 &= \sum_{l'=1}^{L} \frac{(1-r_{l'}^d)(1-r_j^u)(1-P^d)q_j^u}{\left[1-(1-r_j^u)(1-r_{l'}^d)\right]} \sum_{m=1}^{R} C_m X_m^N D_{l',m} K_m \\ &\quad + \sum_{l'=1}^{L} \frac{q_j^u p_{l'}^d (1-r_j^u)(1-P^d)(1-P^u)}{p_j^u \left[1-(1-r_j^u)(1-r_{l'}^d)\right]} \sum_{m=1}^{R} C_m X^{N-1} U_{j,m} K_m \\ &\quad + \sum_{l'=1}^{L} \frac{q_j^u (1-r_j^u)(1-P^d) r_{l'}^d}{\left[1-(1-r_j^u)(1-r_{l'}^d)\right]} \sum_{m=1}^{R} C_m F_m(N,\Upsilon^u,\Delta_{l'}^d) \\ &\quad + \sum_{m=1}^{R} C_m F_m(N-1,\Upsilon^u,\Upsilon^d)p_j^u(1-P^d) \end{aligned} \tag{3.63}$$

Doing the same procedure on $p(N,\Upsilon^u,\Delta_l^d)$,

$$\begin{aligned} p(N,\Upsilon^u,\Delta_l^d) &= \frac{(1-r_l^d)}{r_l^d} \sum_{m=1}^{R} C_m X_m^N D_{j,m} K_m \\ &\quad + \frac{p_l^d}{p_j^u r_l^d}(1-P^u) \sum_{m=1}^{R} C_m X^{N-1} U_{j,m} K_m \\ &\quad + \sum_{m=1}^{R} C_m F_m(N,\Upsilon^u,\Delta_l^d) \\ &= \sum_{j=1}^{J} p(N-1,\Delta_j^u,\Delta_l^d) r_l^u (1-r_l^u) \\ &\quad + \sum_{j=1}^{J} p(N,\Delta_j^u,\Delta_l^d) r_j^u (1-r_l^d) \end{aligned}$$

$$
\begin{aligned}
& +\sum_{j=1}^{J} p(N-1,\Delta_j^u,\Upsilon^d) r_j^u p_l^d \\
& +p(N-1,\Upsilon^u,\Delta_l^d)(1-P^u)(1-r_l^d) \\
& +p(N,\Upsilon^u,\Delta_l^d)(1-Q^u)(1-r_l^d) \\
& +p(N-1,\Upsilon^u,\Upsilon^d)(1-P^u) p_l^d
\end{aligned}
$$

$$
\begin{aligned}
= \quad & p(N-1,\Upsilon^u,\Delta_l^d)(1-P^u)(1-r_l^d) \\
& +\sum_{j=1}^{J} p(N-1,\Delta_j^u,\Delta_l^d) r_l^u (1-r_l^u) \\
& +p(N-1,\Upsilon^u,\Upsilon^d)(1-P^u) p_l^d \\
& +p(N,\Upsilon^u,\Delta_l^d)(1-r_l^d) \\
& +\sum_{j=1}^{J} p(N,\Delta_j^u,\Delta_l^d) r_j^u (1-r_l^d) \\
& +\sum_{j=1}^{J} p(N-1,\Delta_j^u,\Upsilon^d) r_j^u p_l^d \\
& -p(N,\Upsilon^u,\Delta_l^d) Q^u (1-r_l^d)
\end{aligned}
$$

Thus

$$
\begin{aligned}
& \left[P_T(N,\Upsilon^u,\Delta_l^d) + \sum_{m=1}^{R} C_m F_m(N,\Upsilon^u,\Delta_l^d) \right] r_l^d = \\
= \quad & P_T(N-1,\Upsilon^u,\Delta_l^d)(1-P^u)(1-r_l^d) \\
& +\sum_{j=1}^{J} P_T(N-1,\Delta_j^u,\Delta_l^d) r_l^u (1-r_l^u) \\
& +P_T(N-1,\Upsilon^u,\Upsilon^d)(1-P^u) p_l^d \\
& +\sum_{m=1}^{R} C_m F_m(N-1,\Upsilon^u,\Upsilon^d)(1-P^u) p_l^d \\
& +\sum_{j=1}^{J} p(N,\Delta_j^u,\Delta_l^d) r_j^u (1-r_l^d)
\end{aligned}
$$

$$
\begin{aligned}
&+\sum_{j=1}^{J} p(N-1,\Delta_j^u,\Upsilon^d) r_j^u p_l^d \\
&-p(N,\Upsilon^u,\Delta_l^d) Q^u (1-r_l^d)
\end{aligned}
$$

The P_T terms cancel out, therefore

$$
\begin{aligned}
&\sum_{m=1}^{R} C_m F_m(N,\Upsilon^u,\Delta_l^d) = \\
&\quad +\sum_{m=1}^{R} C_m F_m(N-1,\Upsilon^u,\Upsilon^d)\frac{(1-P^u)p_l^d}{r_l^d} \\
&\quad +\sum_{j=1}^{J} p(N,\Delta_j^u,\Delta_l^d)\frac{r_j^u(1-r_l^d)}{r_l^d} \\
&\quad +\sum_{j=1}^{J} p(N-1,\Delta_j^u,\Upsilon^d)\frac{r_j^u p_l^d}{r_l^d} \\
&\quad -p(N,\Upsilon^u,\Delta_l^d)\frac{Q^u(1-r_l^d)}{r_l^d}
\end{aligned}
$$

Re-substituting for $p(N,\Upsilon^u,\Delta_l^d)$ the assumed solution form,

$$
\begin{aligned}
&\sum_{m=1}^{R} C_m F_m(N,\Upsilon^u,\Delta_l^d) = \\
&\quad +\sum_{m=1}^{R} C_m F_m(N-1,\Upsilon^u,\Upsilon^d)\frac{(1-P^u)p_l^d}{r_l^d} \\
&\quad +\sum_{j=1}^{J} p(N,\Delta_j^u,\Delta_l^d)\frac{r_j^u(1-r_l^d)}{r_l^d} \\
&\quad +\sum_{j=1}^{J} p(N-1,\Delta_j^u,\Upsilon^d)\frac{r_j^u p_l^d}{r_l^d} \\
&\quad -\frac{Q^u(1-r_l^d)^2}{r_l^{d2}}\sum_{m=1}^{R} C_m X_m^N D_{j,m} K_m \\
&\quad -\frac{Q^u(1-r_l^d)p_l^d}{p_j^u r_l^{d2}}(1-P^u)\sum_{m=1}^{R} C_m X^{N-1} U_{j,m} K_m
\end{aligned}
$$

$$- \sum_{m=1}^{R} \frac{Q^u(1-r_l^d)}{r_l^d} C_m F_m(N, \Upsilon^u, \Delta_l^d)$$

Thus

$$\begin{aligned}
\sum_{m=1}^{R} C_m F_m(N, \Upsilon^u, \Delta_l^d) \left[r_l^d + Q^u(1-r_l^d) \right] = \\
&+ \sum_{m=1}^{R} C_m F_m(N-1, \Upsilon^u, \Upsilon^d)(1-P^u) p_l^d \\
&+ \sum_{j=1}^{J} p(N, \Delta_j^u, \Delta_l^d) r_j^u (1-r_l^d) \\
&+ \sum_{j=1}^{J} p(N-1, \Delta_j^u, \Upsilon^d) r_j^u p_l^d \\
&- \frac{Q^u(1-r_l^d)^2}{r_l^d} \sum_{m=1}^{R} C_m X_m^N D_{j,m} K_m \\
&- \frac{Q^u(1-r_l^d) p_l^d}{p_j^u r_l^d} (1-P^u) \sum_{m=1}^{R} C_m X^{N-1} U_{j,m} K_m
\end{aligned}$$

Substituting in with equations (3.60) and (3.61)

$$\begin{aligned}
\sum_{m=1}^{R} C_m F_m(N, \Upsilon^u, \Delta_l^d) \left[r_l^d + Q^u(1-r_l^d) \right] = \\
&+ \sum_{m=1}^{R} C_m F_m(N-1, \Upsilon^u, \Upsilon^d)(1-P^u) p_l^d \\
&+ \sum_{j=1}^{J} \frac{q_j^u (1-r_l^d)^3 r_j^u}{r_l^d \left[1-(1-r_j^u)(1-r_l^d) \right]} \sum_{m=1}^{R} C_m X_m^N D_{l',m} K_m \\
&+ \sum_{j=1}^{J} \frac{q_j^u p_l^d (1-r_l^d)^2 r_j^u (1-P^u)}{p_j^u r_l^d \left[1-(1-r_j^u)(1-r_l^d) \right]} \sum_{m=1}^{R} C_m X^{N-1} U_{j,m} K_m \\
&+ \sum_{j=1}^{J} \frac{q_j^u (1-r_l^d)^2 r_j^u}{\left[1-(1-r_j^u)(1-r_l^d) \right]} \sum_{m=1}^{R} C_m F_m(N, \Upsilon^u, \Delta_l^d) \\
&+ \sum_{j=1}^{J} \sum_{l'=1}^{L} \frac{(1-r_{l'}^d) q_j^u r_j^u p_l^d}{\left[1-(1-r_j^u)(1-r_{l'}^d) \right]} \sum_{m=1}^{R} C_m X_m^N D_{l',m} K_m
\end{aligned}$$

$$
\begin{aligned}
&+\sum_{j=1}^{J}\sum_{l'=1}^{L}\frac{q_j^u p_{l'}^d(1-P^u)r_j^u p_l^d}{p_j^u\left[1-(1-r_j^u)(1-r_{l'}^d)\right]}\sum_{m=1}^{R}C_m X^{N-1}U_{j,m}K_m\\
&+\sum_{j=1}^{J}\sum_{l'=1}^{L}\frac{q_j^u r_{l'}^d r_j^u p_l^d}{\left[1-(1-r_j^u)(1-r_{l'}^d)\right]}\sum_{m=1}^{R}C_m F_m(N,\Upsilon^u,\Delta_{l'}^d)\\
&-\frac{Q^u(1-r_l^d)^2}{r_l^d}\sum_{m=1}^{R}C_m X_m^N D_{j,m}K_m\\
&-\frac{Q^u(1-r_l^d)p_l^d}{p_j^u r_l^d}(1-P^u)\sum_{m=1}^{R}C_m X^{N-1}U_{j,m}K_m
\end{aligned}
\tag{3.64}
$$

from Equation (3.63) we see that

$$
\begin{aligned}
&\sum_{m=1}^{R}C_m F_m(N-1,\Upsilon^u,\Upsilon^d)p_j^u(1-P^d)=\\
&\quad-\sum_{l'=1}^{L}\frac{(1-r_{l'}^d)(1-r_j^u)(1-P^d)q_j^u}{\left[1-(1-r_j^u)(1-r_{l'}^d)\right]}\sum_{m=1}^{R}C_m X_m^N D_{l',m}K_m\\
&\quad-\sum_{l'=1}^{L}\frac{q_j^u p_{l'}^d(1-r_j^u)(1-P^d)(1-P^u)}{p_j^u\left[1-(1-r_j^u)(1-r_{l'}^d)\right]}\sum_{m=1}^{R}C_m X^{N-1}U_{j,m}K_m\\
&\quad-\sum_{l'=1}^{L}\frac{q_j^u(1-r_j^u)(1-P^d)r_{l'}^d}{\left[1-(1-r_j^u)(1-r_{l'}^d)\right]}\sum_{m=1}^{R}C_m F_m(N,\Upsilon^u,\Delta_{l'}^d)
\end{aligned}
$$

which is the first term of Equation (3.64) scaled. Substituting,

$$
\begin{aligned}
&\sum_{m=1}^{R}C_m F_m(N,\Upsilon^u,\Delta_l^d)\left[r_l^d+Q^u(1-r_l^d)\right]=\\
&\quad-\sum_{l'=1}^{L}\frac{(1-r_{l'}^d)(1-r_j^u)(1-P^u)p_l^d q_j^u}{p_j^u\left[1-(1-r_j^u)(1-r_{l'}^d)\right]}\sum_{m=1}^{R}C_m X_m^N D_{l',m}K_m\\
&\quad-\sum_{l'=1}^{L}\frac{q_j^u p_{l'}^d(1-r_j^u)(1-P^u)^2 p_l^d}{p_j^{u2}\left[1-(1-r_j^u)(1-r_{l'}^d)\right]}\sum_{m=1}^{R}C_m X^{N-1}U_{j,m}K_m\\
&\quad-\sum_{l'=1}^{L}\frac{q_j^u(1-r_j^u)(1-P^u)p_l^d r_{l'}^d}{p_j^u\left[1-(1-r_j^u)(1-r_{l'}^d)\right]}\sum_{m=1}^{R}C_m F_m(N,\Upsilon^u,\Delta_{l'}^d)
\end{aligned}
$$

$$
\begin{aligned}
&+\sum_{j=1}^{J} \frac{q_j^u(1-r_l^d)^3 r_j^u}{r_l^d\left[1-(1-r_j^u)(1-r_l^d)\right]} \sum_{m=1}^{R} C_m X_m^N D_{l',m} K_m \\
&+\sum_{j=1}^{J} \frac{q_j^u p_l^d(1-r_l^d)^2 r_j^u(1-P^u)}{p_j^u r_l^d\left[1-(1-r_j^u)(1-r_l^d)\right]} \sum_{m=1}^{R} C_m X^{N-1} U_{j,m} K_m \\
&+\sum_{j=1}^{J} \frac{q_j^u(1-r_l^d)^2 r_j^u}{\left[1-(1-r_j^u)(1-r_l^d)\right]} \sum_{m=1}^{R} C_m F_m(N,\Upsilon^u,\Delta_l^d) \\
&+\sum_{j=1}^{J}\sum_{l'=1}^{L} \frac{(1-r_{l'}^d) q_j^u r_j^u p_l^d}{\left[1-(1-r_j^u)(1-r_{l'}^d)\right]} \sum_{m=1}^{R} C_m X_m^N D_{l',m} K_m \\
&+\sum_{j=1}^{J}\sum_{l'=1}^{L} \frac{q_j^u p_{l'}^d(1-P^u) r_j^u p_l^d}{p_j^u\left[1-(1-r_j^u)(1-r_{l'}^d)\right]} \sum_{m=1}^{R} C_m X^{N-1} U_{j,m} K_m \\
&+\sum_{j=1}^{J}\sum_{l'=1}^{L} \frac{q_j^u r_{l'}^d r_j^u p_l^d}{\left[1-(1-r_j^u)(1-r_{l'}^d)\right]} \sum_{m=1}^{R} C_m F_m(N,\Upsilon^u,\Delta_{l'}^d) \\
&-\frac{Q^u(1-r_l^d)^2}{r_l^d} \sum_{m=1}^{R} C_m X_m^N D_{j,m} K_m \\
&-\frac{Q^u(1-r_l^d) p_l^d}{p_j^u r_l^d}(1-P^u) \sum_{m=1}^{R} C_m X^{N-1} U_{j,m} K_m
\end{aligned}
\tag{3.65}
$$

Equation (3.65) must hold true for all m, and thus it sets up a system of RL equations in RL unknowns (one $F_m(N,\Upsilon^u,\Delta_l^d)$ for every (l,m) pair). Once all such unknowns are found, they can be plugged into equation (3.63), to find all the $F_m(N-1,\Upsilon^u,\Upsilon^d)$ (R of them).

With all the internal solutions found, the last terms to find are the C_m's. Because the sum of all probabilities must be unity, setting up a system of simultaneous equations, and then scaling would allow to find all the state probabilities.

Chapter 4

Part-2 Two-Machine Line

4.1 Introduction

This chapter presents the model and solution technique for the two-machine line chosen to represent the behavior of part 2. The model is one with one down mode, and multiple up modes. As opposed to part 1 lines, part-2 observers do not observe changes in up mode or idleness failures. Figure 4-1 depicts the transition space for a typical upstream machine in the two-machine line.

4.2 Notation

The model has multiple up states, and only one down state, therefore the notation used will be Tolio's method but referring to the complementary states. Thus, Υ_i^u represents up mode i for the upstream machine. Υ_f^d represents up mode f for the downstream machine. r_i^u, r_f^d, p_i^u, and p_f^d will have the same meaning.

Because the down mode is a unique state for a machine, Δ^u and Δ^d represent the states of being down for the upstream and downstream machines respectively. For example, $p(n, \Delta^u, \Delta^d)$ represents the probability of the 2-Machine line having both machines down. When being down, there are several repairs with which the machine

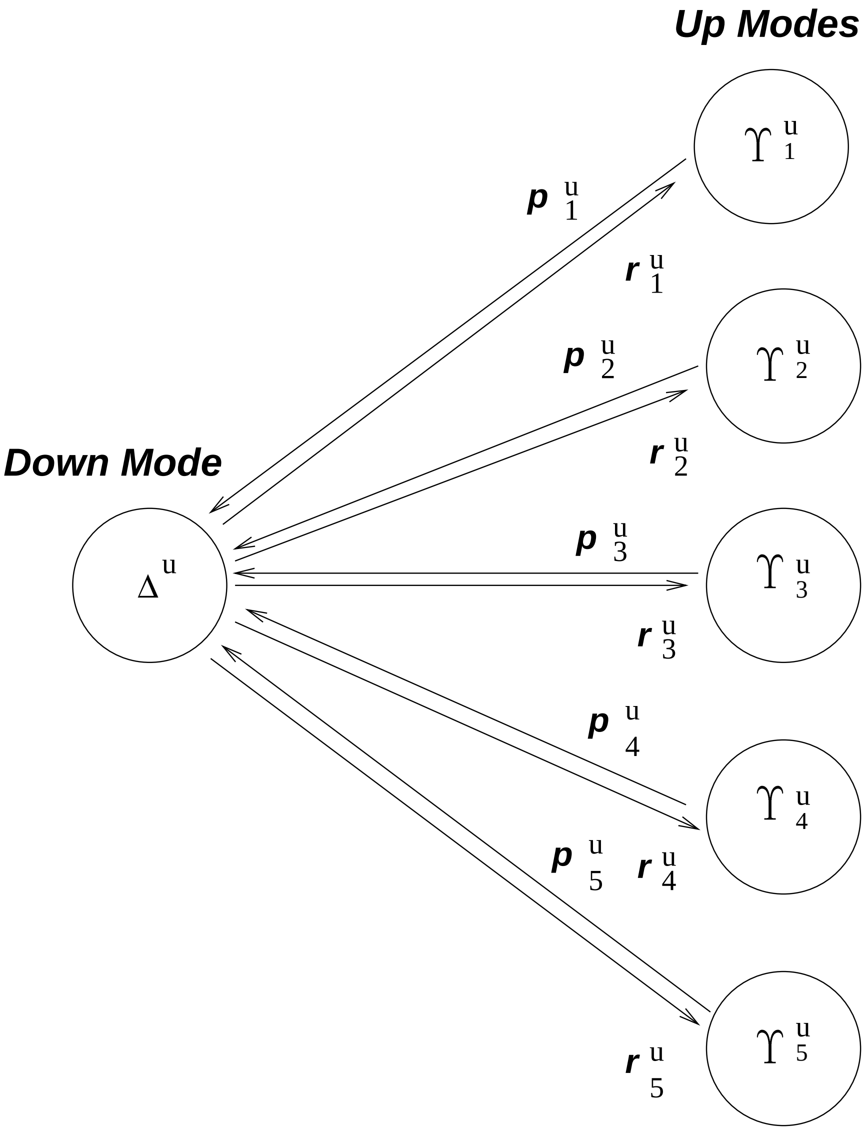

Figure 4-1: State Space for $M^u(6,2)$

can reach the different up modes. We define

$$R^u = \sum_{i=1}^{I} r_i^u$$

and

$$R^d = \sum_{f=1}^{F} r_f^d$$

where I and F are the numbers of up modes for the upstream and downstream machines respectively. We require that $R^u < 1$ and $R^d < 1$.

4.3 Performance Measures

The performance measures for the two machine lines are E^u, E^d, and $\overline{n}$. The first two are defined as

$$E^u = \sum_{n=0}^{N-1} \sum_{i=1}^{I} \left[p(n, \Upsilon_i^u, \Delta^d) + \sum_{f=1}^{F} p(n, \Upsilon_i^u, \Upsilon_f^d) \right]$$

and

$$E^d = \sum_{n=1}^{N} \sum_{f=1}^{F} \left[p(n, \Delta^u, \Upsilon_f^d) + \sum_{i=1}^{J} p(n, \Upsilon_i^u, \Upsilon_f^d) \right]$$

where E^u is the throughput of the upstream machine, and E^d is the throughput of the downstream machine. The solution technique must yield that $E^u = E^d$.

The average buffer level is given by

$$\overline{n} = \sum_{n=0}^{N} n \left[\sum_{i=1}^{I} \sum_{f=1}^{F} p(n, \Upsilon_i^u, \Upsilon_f^d) + \sum_{i=1}^{I} p(n, \Upsilon_i^u, \Delta^d) + \sum_{f=1}^{F} p(n, \Delta^u, \Upsilon_f^d) + p(n, \Delta^u, \Delta^d) \right]$$

4.4 Internal Transition Equations

The transitions equations for the internal states are

$$
\begin{aligned}
p(n,\Delta^u,\Delta^d) &= p(n,\Delta^u,\Delta^d)(1-R^u)(1-R^d) \\
&+\sum_{f=1}^{F} p(n,\Delta^u,\Upsilon_f^d)(1-R^u)p_f^d \\
&+\sum_{i=1}^{I} p(n,\Upsilon_i^u,\Delta^d)(1-R^d)p_i^u \\
&+\sum_{i=1}^{I}\sum_{f=1}^{F} p(n,\Upsilon_i^u,\Upsilon_f^d)p_i^u p_f^d
\end{aligned} \tag{4.1}
$$

$$
\begin{aligned}
p(n,\Delta^u,\Upsilon_f^d) &= p(n+1,\Delta^u,\Delta^d)(1-R^u)r_f^d \\
&+p(n+1,\Delta^u,\Upsilon_f^d)(1-R^u)(1-p_f^d) \\
&+\sum_{i=1}^{I} p(n+1,\Upsilon_i^u,\Delta^d)p_i^u r_f^d \\
&+\sum_{i=1}^{I} p(n+1,\Upsilon_i^u,\Upsilon_f^d)p_i^u(1-p_f^d)
\end{aligned} \tag{4.2}
$$

$$
\begin{aligned}
p(n,\Upsilon_i^u,\Delta^d) &= p(n-1,\Delta^u,\Delta^d)r_i^u(1-R^d) \\
&+\sum_{f=1}^{F} p(n-1,\Delta^u,\Upsilon_f^d)r_i^u p_f^d \\
&+p(n-1,\Upsilon_i^u,\Delta^d)(1-p_i^u)(1-R^d) \\
&+\sum_{f=1}^{F} p(n-1,\Upsilon_i^u,\Upsilon_f^d)(1-p_i^u)p_f^d
\end{aligned} \tag{4.3}
$$

$$
\begin{aligned}
p(n,\Upsilon_i^u,\Upsilon_f^d) &= p(n,\Delta^u,\Delta^d)r_i^u r_f^d \\
&+p(n,\Delta^u,\Upsilon_f^d)r_i^u(1-p_f^d) \\
&+p(n,\Upsilon_i^u,\Delta^d)(1-p_i^u)r_f^d \\
&+p(n,\Upsilon_i^u,\Upsilon_f^d)(1-p_i^u)(1-p_f^d)
\end{aligned} \tag{4.4}
$$

In order to solve for the internal state probabilities, we guess that the solutions take the form

$$\begin{aligned}
p(n, \Delta^u, \Delta^d) &= X^n \\
p(n, \Delta^u, \Upsilon_f^d) &= X^n D_f \\
p(n, \Upsilon_i^u, \Delta^d) &= X^n U_i \\
p(n, \Upsilon_i^u, \Upsilon_f^d) &= X^n U_i D_f \\
& \quad i = 1, ..., I; \\
& \quad f = 1, ..., F; \\
& \quad 2 \leq n \leq N - 2
\end{aligned} \tag{4.5}$$

where X, U_i, and D_f are $1 + I + F$ constants to be evaluated. By substituting (4.5) into (4.1)-(4.4)

$$\begin{aligned}
X^n &= X^n(1 - R^u)(1 - R^d) \\
&+ \sum_{f=1}^{F} X^n D_f (1 - R^u) p_f^d \\
&+ \sum_{i=1}^{I} X^n U_i p_i^u (1 - R^d) \\
&+ \sum_{i=1}^{I} \sum_{f=1}^{F} X^n U_i D_f p_i^u p_f^d
\end{aligned} \tag{4.6}$$

$$\begin{aligned}
X^n D_f &= X^{n+1}(1 - R^u) r_f^d \\
&+ \sum_{i=1}^{I} X^{n+1} U_i p_i^u r_f^d \\
&+ X^{n+1} D_f (1 - R^u)(1 - p_f^d) \\
&+ \sum_{i=1}^{I} X^{n+1} U_i D_f p_i^u (1 - p_f^d)
\end{aligned} \tag{4.7}$$

$$
\begin{aligned}
X^n U_i &= X^{n-1} r_i^u (1 - R^d) \\
&+ \sum_{f=1}^{F} X^{n-1} U_i D_f (1 - p_i^u) p_f^d \\
&+ X^{n-1} U_i (1 - p_i^u)(1 - R^d) \\
&+ \sum_{f=1}^{F} X^{n-1} D_f r_i^u p_f^d
\end{aligned} \tag{4.8}
$$

$$
\begin{aligned}
X^n U_i D_f &= X^n U_i D_f (1 - p_i^u)(1 - p_f^d) \\
&+ X^n D_f r_i^u (1 - p_f^d) \\
&+ X^n U_i (1 - p_i^u) r_f^d \\
&+ X^n r_i^u r_f^d
\end{aligned} \tag{4.9}
$$

After simplification and rearranging, (4.6)-(4.9) result in

$$1 = \left[1 - R^u + \sum_{i=1}^{I} U_i p_i^u\right] \left[1 - R^d + \sum_{f=1}^{F} D_f p_f^d\right] \tag{4.10}$$

$$\frac{D_f}{X} = \left[1 - R^u + \sum_{i=1}^{I} U_i p_i^u\right] \left[(1 - p_f^d) D_f + r_f^d\right] \tag{4.11}$$

$$X U_i = \left[1 - R^d + \sum_{f=1}^{F} D_f p_f^d\right] \left[U_i (1 - p_i^u) + r_i^u\right] \tag{4.12}$$

$$1 = \left[\frac{U_i (1 - p_i^u) + r_i^u}{U_i}\right] \left[\frac{D_f (1 - p_f^d) + r_f^d}{D_f}\right] \tag{4.13}$$

(4.13) implies that there is a constant K such that

$$\left[\frac{U_i (1 - p_i^u) + r_i^u}{U_i}\right] = K$$

and

$$\left[\frac{D_f(1-p_f^d)+r_f^d}{D_f}\right]=\frac{1}{K}\ .$$

Rearranging terms,

$$U_i=\frac{r_i^u}{K-1+p_i^u} \tag{4.14}$$

and

$$D_f=\frac{r_f^d}{\frac{1}{K}-1+p_f^d} \tag{4.15}$$

By using (4.10), (4.14), and (4.15) one gets

$$1 \ = \ \left[1-R^u+\sum_{i=1}^{I}\frac{p_i^u r_i^u}{K-1+p_i^u}\right]\left[1-R^d+\sum_{f=1}^{F}\frac{p_f^d r_f^d}{\frac{1}{K}-1+p_f^d}\right] \tag{4.16}$$

which is the equation which will yield the $m=I+F$ roots, the K_m's that make (4.16) true.

Using (4.12), (4.14), (4.15) and the definition of K in terms of U_i, we get

$$X_m \ = \ \left[1-R^d+\sum_{f=1}^{F}\frac{p_f^d r_f^d}{\frac{1}{K_m}-1+p_f^d}\right]K_m \tag{4.17}$$

$$U_{i,m} \ = \ \frac{r_i^u}{K_m-1+p_i^u} \tag{4.18}$$

$$D_{j,m} \ = \ \frac{r_f^d}{\frac{1}{K_m}-1+p_f^d} \tag{4.19}$$

4.5 Boundary States

The transient states are the same as those in Tolio's 2-Machine line. The equations for the boundary states are

$$\begin{aligned}
p(0,\Delta^u,\Upsilon_f^d) &= p(1,\Delta^u,\Delta^d)(1-R^u)r_f^d \\
&+p(0,\Delta^u,\Upsilon_f^d)(1-R^u) \\
&+p(1,\Delta^u,\Upsilon_f^d)(1-R^u)(1-p_f^d) \\
&+\sum_{i=1}^{I} p(1,\Upsilon_i^u,\Upsilon_f^d)p_i^u(1-p_f^d)
\end{aligned} \tag{4.20}$$

$$\begin{aligned}
p(1,\Delta^u,\Delta^d) &= p(1,\Delta^u,\Delta^d)(1-R^u)(1-R^d) \\
&+\sum_{f=1}^{F} p(1,\Delta^u,\Upsilon_f^d)(1-R^u)p_f^d \\
&+\sum_{i=1}^{I}\sum_{f=1}^{F} p(1,\Upsilon_i^u,\Upsilon_f^d)p_i^u p_f^d
\end{aligned} \tag{4.21}$$

$$\begin{aligned}
p(1,\Delta^u,\Upsilon_f^d) &= p(2,\Delta^u,\Delta^d)(1-R^u)r_f^d \\
&+p(2,\Delta^u,\Upsilon_f^d)(1-R^u)(1-p_f^d) \\
&+\sum_{i=1}^{I} p(2,\Upsilon_i^u,\Delta^d)p_i^u r_f^d \\
&+\sum_{i=1}^{I} p(2,\Upsilon_i^u,\Upsilon_f^d)p_i^u(1-p_f^d)
\end{aligned} \tag{4.22}$$

$$\begin{aligned}
p(1,\Upsilon_i^u,\Upsilon_f^d) &= p(1,\Delta^u,\Delta^d)r_i^u r_f^d \\
&+p(0,\Delta^u,\Upsilon_f^d)r_i^u \\
&+p(1,\Delta^u,\Upsilon_f^d)r_i^u(1-p_f^d) \\
&+p(1,\Upsilon_i^u,\Upsilon_f^d)(1-p_i^u)(1-p_f^d)
\end{aligned} \tag{4.23}$$

$$\begin{aligned} p(2, \Upsilon_i^u, \Delta^d) &= p(1, \Delta^u, \Delta^d)(1 - R^d) r_i^u \\ &+ \sum_{f=1}^{F} p(1, \Delta^u, \Upsilon_f^d) r_i^u p_f^d \\ &+ \sum_{f=1}^{F} p(1, \Upsilon_i^u, \Upsilon_f^d)(1 - p_i^u) p_f^d \end{aligned} \tag{4.24}$$

$$\begin{aligned} p(N-2, \Delta^u, \Upsilon_f^d) &= p(N-1, \Delta^u, \Delta^d)(1 - R^u) r_f^d \\ &+ \sum_{i=1}^{I} p(N-1, \Upsilon_i^u, \Delta^d) p_i^u r_f^d \\ &+ \sum_{i=1}^{I} p(N-1, \Upsilon_i^u, \Upsilon_f^d) p_i^u (1 - p_f^d) \end{aligned} \tag{4.25}$$

$$\begin{aligned} p(N-1, \Delta^u, \Delta^d) &= p(N-1, \Delta^u, \Delta^d)(1 - R^u)(1 - R^d) \\ &+ \sum_{i=1}^{I} p(N-1, \Upsilon_i^u, \Delta^d) p_i^u (1 - R^d) \\ &+ \sum_{i=1}^{I} \sum_{f=1}^{F} p(N-1, \Upsilon_i^u, \Upsilon_f^d) p_i^u p_f^d \end{aligned} \tag{4.26}$$

$$\begin{aligned} p(N-1, \Upsilon_i^u, \Delta^d) &= p(N-2, \Delta^u, \Delta^d) r_i^u (1 - R^d) \\ &+ \sum_{f=1}^{F} p(N-2, \Delta^u, \Upsilon_f^d) r_i^u p_f^d \\ &+ p(N-2, \Upsilon_i^u, \Delta^d)(1 - p_i^u)(1 - R^d) \\ &+ \sum_{f=1}^{F} p(N-2, \Upsilon_i^u, \Upsilon_f^d)(1 - p_i^u) p_f^d \end{aligned} \tag{4.27}$$

$$\begin{aligned} p(N-1, \Upsilon_i^u, \Upsilon_f^d) &= p(N-1, \Delta^u, \Delta^d) r_i^u r_f^d \\ &+ p(N-1, \Upsilon_i^u, \Delta^d)(1 - p_i^u) r_f^d \\ &+ p(N, \Upsilon_i^u, \Delta^u, r_f^d) \\ &+ p(N-1, \Upsilon_i^u, \Upsilon_f^d)(1 - p_i^u)(1 - p_f^d) \end{aligned} \tag{4.28}$$

$$
\begin{aligned}
p(N, \Upsilon_i^u, \Delta^d) \quad = \quad & p(N-1, \Delta^u, \Delta^d) r_i^u (1 - R^d) \\
& + p(N-1, \Upsilon_i^u, \Delta^d)(1 - p_i^u)(1 - R^d) \\
& + p(N, \Upsilon_i^u, \Delta^d)(1 - R^d) \\
& + \sum_{f=1}^{F} p(N-1, \Upsilon_i^u, \Upsilon_f^d)(1 - p_i^u) p_f^d
\end{aligned} \tag{4.29}
$$

4.5.1 Solution to Boundary State Equations

Because of the existing symmetry between the multiple-down-state model of Tolio and the multiple-up-state model developed here, we expect that the solution technique for the boundary state equations be very similar among them. This thesis will not develop the solution technique for the part-2 boundary equations. However, it is noteworthy to mention the states for which the solution is of internal form:

- $p(2, \Upsilon_i^u, \Delta^d)$ and $p(N-2, \Delta^u, \Upsilon_f^d)$ are internal states.
- $p(1, \Delta^u, \Upsilon_f^d)$ and $p(N-1, \Upsilon_i^u, \Delta^d)$ are of internal form and functions of internal states

Therefore, the solution form for such states is

$$p(1, \Delta^u, \Upsilon_f^d) \quad = \quad \sum_{m=1}^{R} C_m X_m D_{f,m} \tag{4.30}$$

$$p(2, \Upsilon_i^u, \Delta^d) \quad = \quad \sum_{m=1}^{R} C_m X_m^2 U_{i,m} \tag{4.31}$$

$$p(N-2, \Delta^u, \Upsilon_f^d) \quad = \quad \sum_{m=1}^{R} C_m X_m^{N-2} D_{f,m} \tag{4.32}$$

$$p(N-1, \Upsilon_i^u, \Delta^d) \quad = \quad \sum_{m=1}^{R} C_m X_m^{N-1} U_{i,m} \tag{4.33}$$

Because of the aforesaid symmetry with Tolio's model, we expect that $p(1, \Upsilon_i^u, \Upsilon_f^d)$ and $p(N-1, \Upsilon_i^u, \Upsilon_f^d)$ be of internal form as well.

Chapter 5

Conclusions and New Research

An analytic method has been proposed to evaluate the performance of a deterministic, two-part-type processing line, with finite buffer capacity, and unreliable machines which can fail in different modes. The method decomposes the long line into multiple part-homogeneous two-machine lines, which are analytically tractable. Using the relationship expected from neighboring two-machine lines, a recursive algorithm similar to Dallery's [3] can thus be developed. Superior time performance is expected when compared to simulation exercises on the same lines, and optimization algorithms of line parameters is thus possible.

The two-part-type line developed here has yet to be completed. A finalized two-part-type line is a final step of the task of developing multiple-part-type lines with multiple-failure modes. The required extension would use the 2-part system of this thesis as a basis for the behavior that larger systems would require for some of their parts. Specifically, the part-1's behavior described here should be consistent with the highest priority part-type in any line. The part-2's behavior described here should be consistent with the lowest priority part-type. The two-machine lines for parts with priorities which are not the highest or lowest are expected to be captured by a single new two-machine line model. Such a model will require the possibility of both multiple up-modes and multiple down-modes, as well as transitions from either and all modes to any other mode.

The development of the two-part-type method required the sacrifice of various characteristics intrinsic to the behavior of the real line. The two-machine line for part-type two was the most affected. In addition to extending the line to more than two part-types, future work would be useful in trying to develop more comprehensive methods for capturing the behavior of the part-2 two-machine line. As seen in the Chapter 2, the solution for internal states presented a new challenge in the form of recursive equations. The developed equations for U and D are recursive on K_m as is the equation used to find the roots. For such reason, the part 2 two-machine line was evaluated assuming that changes in failure mode were not significant. Future research should concentrate in determining if such assumption is valid, and in developing a solution algorithm that deals with the recursive nature of the solution equations.

Appendix A

Lower Boundary $P_T(state)$

$$
\begin{aligned}
p(2, \Upsilon^u, \Delta_l^d) &= \sum_{j=1}^{J} p(1, \Delta_j^u, \Delta_l^d) r_j^u (1 - r_l^d) \\
&\quad + \sum_{j=1}^{J} p(1, \Delta_j^u, \Upsilon^d) r_j^u p_l^d \\
&\quad + p(1, \Upsilon^u, \Upsilon^d)(1 - P^u) p_l^d
\end{aligned} \tag{A.1}
$$

We can substitute Tolio's internal equation solutions to the equation above and rearrange terms to find:

$$
\begin{aligned}
\sum_{m=1}^{R} C_m X_m^2 D_{l,m} &= \sum_{j=1}^{J} \sum_{m=1}^{R} C_m X_m U_{j,m} D_{l,m} r_j^u (1 - r_l^d) \\
&\quad + \sum_{j=1}^{J} \sum_{m=1}^{R} C_m X_m U_{j,m} r_j^u p_l^d \\
&\quad + \frac{1}{p_l^d} \sum_{m=1}^{R} C_m \left[X_m \frac{D_{l,m}}{K_m} \right] (1 - P^u) p_l^d \\
&= \sum_{j=1}^{J} \sum_{m=1}^{R} C_m X_m U_{j,m} D_{l,m} r_j^u (1 - r_l^d)
\end{aligned}
$$

$$
\begin{aligned}
&+\sum_{j=1}^{J}\sum_{m=1}^{R} C_m X_m U_{j,m} r_j^u p_l^d \\
&+\sum_{m=1}^{R} C_m X_m \frac{D_{l,m}}{K_m}(1-P^u) \\
= &\sum_{j=1}^{J}\sum_{m=1}^{R} C_m X_m U_{j,m} r_j^u \left[p_l^d + D_{l,m}(1-r_l^d)\right] \\
&+\sum_{m=1}^{R} C_m X_m \frac{D_{l,m}}{K_m}(1-P^u)
\end{aligned}
\tag{A.2}
$$

It can be shown that

$$
\frac{D_{l,m}}{K_m} = p_l^d + D_{l,m}(1-r_l^d) \ ,
$$

therefore

$$
\begin{aligned}
\sum_{m=1}^{R} C_m X_m^2 D_{l,m} &= \sum_{j=1}^{J}\sum_{m=1}^{R} C_m X_m U_{j,m} r_j^u \frac{D_{l,m}}{K_m} \\
&\quad +\sum_{m=1}^{R} C_m X_m \frac{D_{l,m}}{K_m}(1-P^u) \\
&= \sum_{m=1}^{R} C_m X_m \frac{D_{l,m}}{K_m}\left[1-P^u+\sum_{j=1}^{J} r_j^u U_{j,m}\right]
\end{aligned}
\tag{A.3}
$$

Since

$$
X_m K_m = 1 - P^u + \sum_{j=1}^{J} r_j^u U_{j,m} \ ,
$$

then

$$p(2, \Upsilon^u, \Delta_l^d) = \sum_{m=1}^{R} C_m X_m^2 D_{l,m} \tag{A.4}$$

Following a similar process for $p(1, \Delta_j^u, \Delta_l^d)$,

$$\begin{aligned} p(1, \Delta_j^u, \Delta_l^d) &= p(1, \Delta_j^u, \Delta_l^d)(1 - r_j^u)(1 - r_l^d) \\ &\quad + p(1, \Delta_j^u, \Upsilon^d)(1 - r_j^u)p_l^d \\ &\quad + p(1, \Upsilon^u, \Upsilon^d)p_j^u p_l^d \end{aligned}$$

Re-arranging terms,

$$\begin{aligned} p(1, \Delta_j^u, \Delta_l^d)\left[1 - (1 - r_j^u)(1 - r_l^d)\right] &= p(1, \Delta_j^u, \Upsilon^d)(1 - r_j^u)p_l^d \\ &\quad + p(1, \Upsilon^u, \Upsilon^d)p_j^u p_l^d \end{aligned}$$

so

$$\begin{aligned} p(1, \Delta_j^u, \Delta_l^d)\left[1 - (1 - r_j^u)(1 - r_l^d)\right] &= \sum_{m=1}^{R} C_m X_m U_{j,m}(1 - r_j^u)p_l^d \\ &\quad + p(1, \Upsilon^u, \Upsilon^d)p_j^u p_l^d \\ &= \sum_{m=1}^{R} C_m X_m U_{j,m}(1 - r_j^u)p_l^d \\ &\quad + \sum_{m=1}^{R} C_m X_m \frac{D_{l,m}}{K_m} p_j^u \\ &= \sum_{m=1}^{R} C_m X_m \left[U_{j,m}(1 - r_j^u)p_l^d + \frac{D_{l,m}}{K_m} p_j^u\right] \end{aligned}$$

It can be shown that

$$\frac{D_{l,m}}{K_m} = D_{l,m}(1 - r_l^d) + p_l^d$$

so

$$\begin{aligned}
U_{j,m}(1 - r_j^u)p_l^d + \frac{D_{l,m}}{K_m}p_j^u &= U_{j,m}p_j^d(1 - r_j^u) + D_{l,m}(1 - r_l^d)p_j^u + p_j^u p_l^d \\
&= p_l^d\left[U_{j,m}(1 - r_j^u) + p_j^u\right] + D_{l,m}(1 - r_l^d)p_j^u \\
&= \left[U_{j,m}(1 - r_j^u) + p_j^u\right] \times \left[p_l^d + D_{l,m}(1 - r_l^d)\right] \\
&\quad - D_{l,m}U_{j,m}(1 - r_l^d)(1 - r_j^u) \\
&= U_{j,m}D_{l,m} - D_{l,m}U_{j,m}(1 - r_l^d)(1 - r_j^u) \\
&= U_{j,m}D_{l,m}\left[1 - (1 - r_j^u)(1 - r_l^d)\right]
\end{aligned}$$

Thus,

$$p(1, \Delta_j^u, \Delta_l^d) = \sum_{m=1}^{R} C_m X_m U_{j,m} D_{l,m} \tag{A.5}$$

Doing the same procedure on $p(0, \Delta_j^u, \Upsilon^d)$,

$$p(0, \Delta_j^u, \Upsilon^d) = \sum_{l=1}^{L} p(0, \Delta_j^u, \Delta_l^d)(1 - r_j^u)$$

$$
\begin{aligned}
&+\sum_{l=1}^{L} p(1,\Delta_j^u,\Delta_l^d)(1-r_j^u)r_l^d \\
&+p(1,\Delta_j^u,\Upsilon^d)(1-r_j^u)(1-P^d) \\
&+p(1,\Upsilon^u,\Upsilon^d)p_j^u(1-P^d)
\end{aligned}
$$

Thus,

$$
\begin{aligned}
p(0,\Delta_j^u,\Upsilon^d)r_j^u &= \sum_{l=1}^{L}\sum_{m=1}^{R} C_m X_m U_{j,m} D_{l,m}(1-r_j^u)r_l^d \\
&+\sum_{m=1}^{R} C_m X_m U_{j,m}(1-r_j^u)(1-P^d) \\
&+\frac{1}{p_l^d}\sum_{m=1}^{R} C_m X_m \frac{D_{l,m}}{K_m} p_j^u(1-P^d)
\end{aligned}
$$

Re-arranging terms,

$$
\begin{aligned}
p(0,\Delta_j^u,\Upsilon^d) &= \sum_{l=1}^{L}\sum_{m=1}^{R} C_m X_m U_{j,m} D_{l,m}\frac{(1-r_j^u)}{r_j^u}r_l^d \\
&+\sum_{m=1}^{R} C_m X_m U_{j,m}\frac{(1-r_j^u)}{r_j^u}(1-P^d) \\
&+\frac{1}{r_j^u p_l^d}\sum_{m=1}^{R} C_m X_m \frac{D_{l,m}}{K_m} p_j^u(1-P^d) \\
&= \frac{1-r_j^u}{r_j^u}\sum_{m=1}^{R} C_m X_m U_{j,m}\left[1-P^d+\sum_{l=1}^{L} D_{l,m} r_l^d\right] \\
&+\frac{p_j^u}{r_j^u p_l^d}(1-P^d)\sum_{m=1}^{R} C_m X_m \frac{D_{l,m}}{K_m}
\end{aligned}
$$

It can be shown that

$$
\frac{1}{X_m K_m} = 1 - P^d + \sum_{l=1}^{L} D_{l,m} r_l^d \ ,
$$

thus

$$
\begin{aligned}
p(0, \Delta_j^u, \Upsilon^d) &= \frac{1 - r_j^u}{r_j^u} \sum_{m=1}^{R} C_m \frac{U_{j,m}}{K_m} \\
&+ \frac{p_j^u}{r_j^u p_l^d}(1 - P^d) \sum_{m=1}^{R} C_m X_m \frac{D_{l,m}}{K_m} \qquad \text{(A.6)}
\end{aligned}
$$

Appendix B

Part-2 Two-Machine-Line ($z \neq 0$)

B.1 Motivation

Chapter 3 discussed the the part-2 two-machine-line model assuming that the failure mode change probabilities were zero ($z = 0$). The reason why such assumption is made is that the introduction of z increases the complexity of the solution search for the two-machine-line's U, D and X parameters. This appendix discusses the version of the model with $z \neq 0$.

B.2 Notation

The notation used is identical as the one introduced in Chapter 3. However, since $z \neq 0$, remember that

$$\tilde{r}_j^u = r_j^u + \sum_{t=j+1}^{J} z_{j,t}^u$$

and

$$\tilde{r}_l^d = r_l^d + \sum_{x=1}^{l-1} z_{l,x}^d$$

B.3 Internal States

The set of transition equations for the internal states is defined by

$$\begin{aligned}
p(n, \Delta_j^u, \Delta_l^d) = {} & p(n, \Delta_j^u, \Delta_l^d)(1-\tilde{r}_j^u)(1-\tilde{r}_l^d) \\
& + \sum_{t=1}^{j-1} p(n, \Delta_t^u, \Delta_l^d) z_{t,j}^u (1-\tilde{r}_l^d) \\
& + \sum_{x=1}^{l-1} p(n, \Delta_j^u, \Delta_x^d)(1-\tilde{r}_j^u) z_{x,l}^d \\
& + \sum_{t=1}^{j-1}\sum_{x=1}^{l-1} p(n, \Delta_t^u, \Delta_x^d) z_{t,j}^u z_{x,l}^d \\
& + p(n, \Delta_j^u, \Upsilon^d)(1-\tilde{r}_j^u) p_l^d \\
& + \sum_{t=1}^{j-1} p(n, \Delta_t^u, \Upsilon^d) z_{t,j}^u p_l^d \\
& + p(n, \Upsilon^u, \Delta_l^d)(1-\tilde{r}_l^d) p_j^u \\
& + \sum_{x=1}^{l-1} p(n, \Upsilon^u, \Delta_x^d) p_j^u z_{x,l}^d \\
& + p(n, \Upsilon^u, \Upsilon^d) p_j^u p_l^d
\end{aligned} \tag{B.1}$$

$$\begin{aligned}
p(n, \Delta_j^u, \Upsilon^d) = {} & \sum_{l=1}^{L}\sum_{t=1}^{j-1} p(n+1, \Delta_t^u, \Delta_l^d) z_{t,j}^u r_l^d \\
& + \sum_{l=1}^{L} p(n+1, \Delta_j^u, \Delta_l^d)(1-\tilde{r}_j^u) r_l^d \\
& + p(n+1, \Delta_j^u, \Upsilon^d)(1-\tilde{r}_j^u)(1-P^d) \\
& + \sum_{t=1}^{j-1} p(n+1, \Delta_t^u, \Upsilon^d) z_{t,j}^u (1-P^d) \\
& + \sum_{l=1}^{L} p(n+1, \Upsilon^u, \Delta_l^d) p_j^u r_l^d \\
& + p(n+1, \Upsilon^u, \Upsilon^d) p_j^u (1-P^d)
\end{aligned} \tag{B.2}$$

$$
\begin{aligned}
p(n, \Upsilon^u, \Delta_l^d) &= \sum_{j=1}^{J} p(n-1, \Delta_j^u, \Delta_l^d) r_j^u (1 - \tilde{r}_l^d) \\
&+ \sum_{j=1}^{J} \sum_{x=1}^{l-1} p(n-1, \Delta_j^u, \Delta_x^d) r_j^u z_{x,l}^d \\
&+ \sum_{j=1}^{J} p(n-1, \Delta_j^u, \Upsilon^d) r_j^u p_l^d \\
&+ p(n-1, \Upsilon^u, \Delta_l^d)(1 - P^u)(1 - \tilde{r}_l^d) \\
&+ \sum_{x=1}^{l-1} p(n-1, \Upsilon^u, \Delta_x^d)(1 - P^u) z_{x,l}^d \\
&+ p(n-1, \Upsilon^u, \Upsilon^d)(1 - P^u) p_l^d
\end{aligned} \tag{B.3}
$$

$$
\begin{aligned}
p(n, \Upsilon^u, \Upsilon^d) &= \sum_{j=1}^{J} \sum_{l=1}^{L} p(n, \Delta_j^u, \Delta_l^d) r_j^u r_l^d \\
&+ \sum_{j=1}^{J} p(n, \Delta_j^u, \Upsilon^d) r_j^u (1 - P^d) \\
&+ \sum_{l=1}^{L} p(n, \Upsilon^u, \Delta_l^d)(1 - P^u) r_l^d \\
&+ p(n, \Upsilon^u, \Upsilon^d)(1 - P^u)(1 - P^d)
\end{aligned} \tag{B.4}
$$

To solve, a guess is taken by making the internal steady state probabilities assume the form

$$
\begin{aligned}
p(n, \Delta_j^u, \Delta_l^d) &= X^n U_j D_l \\
p(n, \Delta_j^u, \Upsilon^d) &= X^n U_j \\
p(n, \Upsilon^u, \Delta_l^d) &= X^n D_l \\
p(n, \Upsilon^u, \Upsilon^d) &= X^n \\
& \quad j = 1, ..., J \\
& \quad l = 1, ..., L \\
& \quad 2 \leq n \leq N-2
\end{aligned} \tag{B.5}
$$

where X, U_j, and D_l are $1+J+L$ constants to be evaluated. This solution structure is the one used by Tolio, and is expected to be appropriate for this adapted model.

By substituting (B.5) into (B.1)-(B.4)

$$
\begin{aligned}
X^n U_j D_l = {} & X^n U_j D_l (1-\tilde{r}^u_j)(1-\tilde{r}^d_l) \\
& + X^n U_j (1-\tilde{r}^u_j) p^d_l \\
& + X^n D_l p^u_j (1-\tilde{r}^d_l) \\
& + X^n p^u_j p^d_l \\
& + \sum_{t=1}^{j-1} X^n U_t D_l z^u_{t,j} (1-\tilde{r}^d_l) \\
& + \sum_{x=1}^{l-1} X^n U_j D_x (1-\tilde{r}^u_j) z^d_{x,l} \\
& + \sum_{t=1}^{j-1} \sum_{x=1}^{l-1} X^n U_t D_x z^u_{t,j} z^d_{x,l} \\
& + \sum_{t=1}^{j-1} X^n U_t z^u_{t,j} p^d_l \\
& + \sum_{x=1}^{l-1} X^n D_x p^u_j z^d_{x,l}
\end{aligned} \tag{B.6}
$$

$$
\begin{aligned}
X^n U_j = {} & X^{n+1} p^u_j (1-P^d) \\
& + X^{n+1} U_j (1-\tilde{r}^u_j)(1-P^d) \\
& + \sum_{l=1}^{L} X^{n+1} D_l p^u_j r^d_l \\
& + \sum_{l=1}^{L} X^{n+1} U_j D_l (1-\tilde{r}^u_j) r^d_l \\
& + \sum_{t=1}^{j-1} X^{n+1} U_t z^u_{t,j} (1-P^d) \\
& + \sum_{l=1}^{L} \sum_{t=1}^{j-1} X^{n+1} U_t D_l z^u_{t,j} r^d_l
\end{aligned} \tag{B.7}
$$

$$
\begin{aligned}
X^n D_l = \; & X^{n-1}(1-P^u)p_l^d \\
& + \sum_{j=1}^{J} X^{n-1} U_j D_l r_j^u (1-\tilde{r}_l^d) \\
& + \sum_{j=1}^{J} X^{n-1} U_j r_j^u p_l^d \\
& + X^{n-1} D_l (1 - P^u(1-\tilde{r}_l^d) \\
& + \sum_{x=1}^{l-1} X^{n-1} U_j D_x (1-P^u) z_{x,l}^d \\
& + \sum_{j=1}^{J} \sum_{x=1}^{l-1} X^{n-1} D_x r_j^u z_{x,l}^d
\end{aligned} \tag{B.8}
$$

$$
\begin{aligned}
X^n = \; & X^n(1-P^u)(1-P^d) \\
& + \sum_{j=1}^{J} X^n U_j r_j^u (1-P^d) \\
& + \sum_{l=1}^{L} X^n D_l (1-P^u) r_l^d \\
& + \sum_{j=1}^{J} \sum_{l=1}^{L} X^n U_j D_l r_j^u r_l^d
\end{aligned} \tag{B.9}
$$

After much simplification (B.6)-(B.9) reduce to

$$U_j D_l = \left[U_j(1-\tilde{r}_j^u) + p_j^u + \sum_{t=1}^{j-1} U_t z_{t,j}^u \right] \left[D_l(1-\tilde{r}_l^d) + p_l^d + \sum_{x=1}^{l-1} D_x z_{x,l}^d \right] \tag{B.10}$$

$$\frac{U_j}{X} = \left[1 - P^d + \sum_{l=1}^{L} D_l r_l^d \right] \left[U_j(1-\tilde{r}_j^u) + p_j^u + \sum_{t=1}^{j-1} U_t z_{t,j}^u \right] \tag{B.11}$$

$$X D_l = \left[1 - P^u + \sum_{j=1}^{J} U_j r_j^u \right] \left[D_l(1-\tilde{r}_l^d) + p_l^d + \sum_{x=1}^{l-1} D_x z_{x,l}^d \right] \tag{B.12}$$

$$1 = \left[1 - P^u + \sum_{j=1}^{J} U_j r_j^u \right] \left[1 - P^d + \sum_{l=1}^{L} D_l r_l^d \right] \tag{B.13}$$

Rearranging terms in (B.10)

$$
\begin{aligned}
1 \quad = \quad & \left[\frac{U_j(1-\tilde{r}_j^u) + p_j^u + \sum_{t=1}^{j-1} U_t z_{t,j}^u}{U_j}\right]\left[\frac{D_l(1-\tilde{r}_l^d) + p_l^d + \sum_{x=1}^{l-1} D_x z_{x,l}^d}{D_l}\right] \\
& \text{j = 1, ..., J; l = 1, ..., L.}
\end{aligned}
$$

which means that for some constant K

$$\left[\frac{U_j(1-\tilde{r}_j^u) + p_j^u + \sum_{t=1}^{j-1} U_t z_{t,j}^u}{U_j}\right] = K, \text{ j=1,...,J.}$$

and

$$\left[\frac{D_l(1-\tilde{r}_l^d) + p_l^d + \sum_{x=1}^{l-1} U_x z_{x,l}^d}{D_l}\right] = \frac{1}{K}, \text{ l=1,...,L.}$$

Consequently,

$$U_j = \frac{p_j^u + \sum_{t=1}^{j-1} U_t z_{t,j}^u}{K - 1 + \tilde{r}_j^u}, \text{ j=1,...,J.} \tag{B.14}$$

and

$$D_l = \frac{p_l^d + \sum_{x=1}^{l-1} D_x z_{x,l}^d}{\frac{1}{K} - 1 + \tilde{r}_l^d}, \text{ l=1,...,L.} \tag{B.15}$$

Since U_1 and D_1 are the first in their respective sequences, then all U_j and D_l can be solved recursively once U_1 and D_1 are found.

By introducing (B.14) and (B.15) in (B.13)

$$
\begin{aligned}
1 \quad = \quad & \left[1 - P^u + \sum_{j=1}^{J} \frac{(p_j^u + \sum_{t=1}^{j-1} U_t z_{t,j}^u) r_j^u}{K - 1 + \tilde{r}_j^u}\right] \\
& \left[1 - P^d + \sum_{l=1}^{L} \frac{(p_l^d + \sum_{x=1}^{l-1} D_x z_{x,l}^d) r_l^d}{\frac{1}{K} - 1 + \tilde{r}_l^d}\right]
\end{aligned} \tag{B.16}
$$

Unfortunately, this is not $R = JL$ order polynomial in K as in the original Tolio case [12]. Finding the roots of this equation is not a trivial task, and the required solution technique will not be explored here. It is not clear what the new roots mean, or if all the roots are real.

Defining K_m to be the m^{th} root of the polynomial, if all the roots are found, the values of $U_{j,m,}$, $D_{l,m}$, and X_m can also be found.

Using (B.12), (B.14), and (B.15), the equations needed for $U_{j,m,}$, $D_{l,m}$, and X_m are found to be

$$X_m = \left[1 - P^u + \sum_{j=1}^{J} \frac{(p_j^u + \sum_{t=1}^{j-1} U_t z_{t,j}^u) r_j^u}{K_m - 1 + \tilde{r}_j^u}\right] \frac{1}{K_m} \qquad m = 1, ..., J + L. \tag{B.17}$$

$$U_{j,m} = \frac{p_j^u + \sum_{t=1}^{j-1} U_t z_{t,j}^u}{K_m - 1 + \tilde{r}_j^u}, \text{ j=1,...,J.} \tag{B.18}$$

$$D_{l,m} = \frac{p_l^d + \sum_{x=1}^{l-1} D_x z_{x,l}^d}{\frac{1}{K_m} - 1 + \tilde{r}_l^d}, \text{ l=1,...,L.} \tag{B.19}$$

B.4 Boundary States

Because of idleness failures, the states considered transient are fewer in number than in the original Tolio decomposition version. The states that remain transient are $p(0, \Upsilon^u, \Upsilon^d)$, $p(0, \Upsilon^u, \Delta_l^d)$, $p(N, \Upsilon^u, \Upsilon^d)$, and $p(N, \Delta_j^u, \Upsilon^d)$.

The transition equations for the boundary state probabilities are

$$p(0, \Delta_j^u, \Delta_l^d) = p(0, \Delta_j^u, \Delta_l^d)(1 - \tilde{r}_j^u)(1 - \tilde{r}_l^d) + \sum_{t=1}^{j-1} p(0, \Delta_t^u, \Delta_l^d) z_{t,j}^u (1 - \tilde{r}_l^d)$$

$$
\begin{aligned}
&+\sum_{x=1}^{l-1} p(0,\Delta_j^u,\Delta_x^d)(1-\tilde{r}_j^u)z_{x,l}^d \\
&+\sum_{t=1}^{j-1}\sum_{x=1}^{l-1} p(0,\Delta_t^u,\Delta_x^d)z_{t,j}^u z_{x,l}^d \\
&+p(0,\Delta_j^u,\Upsilon^d)(1-\tilde{r}_j^u)q_l^d \\
&+\sum_{t=1}^{j-1} p(0,\Delta_t^u,\Upsilon^d)z_{t,j}^u q_l^d
\end{aligned} \tag{B.20}
$$

$$
\begin{aligned}
p(0,\Delta_j^u,\Upsilon^d) \;=\; &\sum_{l=1}^{L} p(0,\Delta_j^u,\Delta_l^d)(1-\tilde{r}_j^u)r_l^d \\
&+\sum_{l=1}^{L}\sum_{t=1}^{j-1} p(0,\Delta_t^u,\Delta_l^d)z_{t,j}^u r_l^d \\
&+\sum_{l=1}^{L} p(1,\Delta_j^u,\Delta_l^d)(1-\tilde{r}_j^u)r_l^d \\
&+\sum_{l=1}^{L}\sum_{t=1}^{j-1} p(1,\Delta_t^u,\Delta_l^d)z_{t,j}^u r_l^d \\
&+p(0,\Delta_j^u,\Upsilon^d)(1-\tilde{r}_j^u)(1-Q^d) \\
&+\sum_{t=1}^{j-1} p(0,\Delta_t^u,\Upsilon^d)z_{t,j}^u(1-Q^d) \\
&+p(1,\Delta_j^u,\Upsilon^d)(1-\tilde{r}_j^u)(1-P^d) \\
&+\sum_{t=1}^{j-1} p(1,\Delta_t^u,\Upsilon^d)z_{t,j}^u(1-P^d) \\
&+\sum_{l=1}^{L} p(1,\Upsilon^u,\Delta_l^d)p_j^u r_l^d \\
&+p(1,\Upsilon^u,\Upsilon^d)p_j^u(1-P^d)
\end{aligned} \tag{B.21}
$$

$$
p(0,\Upsilon^u,\Delta_l^d) \;=\; 0 \tag{B.22}
$$

$$
p(0,\Upsilon^u,\Upsilon^d) \;=\; 0 \tag{B.23}
$$

$$
p(1,\Delta_j^u,\Delta_l^d) \;=\; p(1,\Delta_j^u,\Delta_l^d)(1-\tilde{r}_j^u)(1-\tilde{r}_l^d)
$$

$$
\begin{aligned}
&+\sum_{t=1}^{j-1} p(1, \Delta_t^u, \Delta_l^d) z_{t,j}^u (1-\tilde{r}_l^d) \\
&+\sum_{x=1}^{l-1} p(1, \Delta_j^u, \Delta_x^d)(1-\tilde{r}_j^u) z_{x,l}^d \\
&+\sum_{t=1}^{j-1}\sum_{x=1}^{l-1} p(1, \Delta_t^u, \Delta_x^d) z_{t,j}^u z_{x,l}^d \\
&+p(1, \Delta_j^u, \Upsilon^d)(1-\tilde{r}_j^u) p_l^d \\
&+\sum_{t=1}^{j-1} p(1, \Delta_t^u, \Upsilon^d) z_{t,j}^u p_l^d \\
&+p(1, \Upsilon^u, \Delta_l^d) p_j^u (1-\tilde{r}_l^d) \\
&+\sum_{x=1}^{l-1} p(1, \Upsilon^u, \Delta_x^d) p_j^u z_{x,l}^d \\
&+p(1, \Upsilon^u, \Upsilon^d) p_j^u p_l^d
\end{aligned} \tag{B.24}
$$

$$
\begin{aligned}
p(1, \Delta_j^u, \Upsilon^d) \;=\; &\sum_{l=1}^{L} p(2, \Delta_j^u, \Delta_l^d)(1-\tilde{r}_j^u) r_l^d \\
&+\sum_{l=1}^{L}\sum_{t=1}^{j-1} p(2, \Delta_t^u, \Delta_l^d) z_{t,j}^u r_l^d \\
&+p(2, \Delta_j^u, \Upsilon^d)(1-\tilde{r}_j^u)(1-P^d) \\
&+\sum_{t=1}^{j-1} p(2, \Delta_t^u, \Upsilon^d) z_{t,j}^u (1-P^d) \\
&+\sum_{l=1}^{L} p(2, \Upsilon^u, \Delta_l^d) p_j^u r_l^d \\
&+p(2, \Upsilon^u, \Upsilon^d) p_j^u (1-P^d)
\end{aligned} \tag{B.25}
$$

$$
\begin{aligned}
p(1, \Upsilon^u, \Delta_l^d) \;=\; &\sum_{j=1}^{J} p(0, \Delta_j^u, \Delta_l^d) r_j^u (1-\tilde{r}_l^d) \\
&+\sum_{j=1}^{J}\sum_{x=1}^{l-1} p(0, \Delta_j^u, \Delta_x^d) r_j^u z_{x,l}^d \\
&+\sum_{j=1}^{J} p(0, \Delta_j^u, \Upsilon^d) r_j^u q_l^d
\end{aligned} \tag{B.26}
$$

$$
\begin{aligned}
p(1,\Upsilon^u,\Upsilon^d) &= \sum_{j=1}^{J}\sum_{l=1}^{L} p(0,\Delta_j^u,\Delta_l^d) r_j^u r_l^d \\
&+ \sum_{j=1}^{J}\sum_{l=1}^{L} p(1,\Delta_j^u,\Delta_l^d) r_j^u r_l^d \\
&+ \sum_{j=1}^{J} p(0,\Delta_j^u,\Upsilon^d) r_j^u (1-Q^d) \\
&+ \sum_{j=1}^{J} p(1,\Delta_j^u,\Upsilon^d) r_j^u (1-P^d) \\
&+ \sum_{l=1}^{L} p(1,\Upsilon^u,\Delta_l^d)(1-P^u) r_l^d \\
&+ p(1,\Upsilon^u,\Upsilon^d)(1-P^u)(1-P^d)
\end{aligned} \tag{B.27}
$$

$$
\begin{aligned}
p(2,\Upsilon^u,\Delta_l^d) &= \sum_{j=1}^{J} p(1,\Delta_j^u,\Delta_l^d) r_j^u (1-\tilde{r}_l^d) \\
&+ \sum_{j=1}^{J}\sum_{x=1}^{l-1} p(1,\Delta_j^u,\Delta_x^d) r_j^u z_{x,l}^d \\
&+ \sum_{j=1}^{J} p(1,\Delta_j^u,\Upsilon^d) r_j^u p_l^d \\
&+ p(1,\Upsilon^u,\Delta_l^d)(1-P^u)(1-\tilde{r}_l^d) \\
&+ \sum_{x=1}^{l-1} p(1,\Upsilon^u,\Delta_x^d)(1-P^u) z_{x,l}^d \\
&+ p(1,\Upsilon^u,\Upsilon^d)(1-P^u) p_l^d
\end{aligned} \tag{B.28}
$$

$$
\begin{aligned}
p(N-2,\Delta_j^u,\Upsilon^d) &= \sum_{l=1}^{L} p(N-1,\Delta_j^u,\Delta_l^d)(1-\tilde{r}_j^u) r_l^d \\
&+ \sum_{l=1}^{L}\sum_{t=1}^{j-1} p(N-1,\Delta_t^u,\Delta_l^d) z_{t,j}^u r_l^d \\
&+ p(N-1,\Delta_j^u,\Upsilon^d)(1-\tilde{r}_j^u)(1-P^d) \\
&+ \sum_{t=1}^{j-1} p(N-1,\Delta_t^u,\Upsilon^d) z_{t,j}^u (1-P^d) \\
&+ \sum_{l=1}^{L} p(N-1,\Upsilon^u,\Delta_l^d) p_j^u r_l^d
\end{aligned}
$$

$$+p(N-1,\Upsilon^u,\Upsilon^d)p_j^u(1-P^d) \tag{B.29}$$

$$\begin{aligned}
p(N-1,\Delta_j^u,\Delta_l^d) &= p(N-1,\Delta_j^u,\Delta_l^d)(1-\tilde{r}_j^u)(1-\tilde{r}_l^d) \\
&+\sum_{t=1}^{j-1} p(N-1,\Delta_t^u,\Delta_l^d)z_{t,j}^u(1-\tilde{r}_l^d) \\
&+\sum_{x=1}^{l-1} p(N-1,\Delta_j^u,\Delta_x^d)(1-\tilde{r}_j^u)z_{x,l}^d \\
&+\sum_{t=1}^{j-1}\sum_{x=1}^{l-1} p(N-1,\Delta_t^u,\Delta_x^d)z_{t,j}^u z_{x,l}^d \\
&+p(N-1,\Delta_j^u,\Upsilon^d)(1-\tilde{r}_j^u)p_l^d \\
&+p(N-1,\Delta_t^u,\Upsilon^d)z_{t,j}^u p_l^d \\
&+p(N-1,\Upsilon^u,\Delta_l^d)p_j^u(1-\tilde{r}_l^d) \\
&+\sum_{x=1}^{l-1} p(N-1,\Upsilon^u,\Delta_x^d)p_j^u z_{x,l}^d \\
&+p(N-1,\Upsilon^u,\Upsilon^d)p_j^u p_l^d
\end{aligned} \tag{B.30}$$

$$\begin{aligned}
p(N-1,\Delta_j^u,\Upsilon^d) &= \sum_{l=1}^{L} p(N,\Delta_j^u,\Delta_l^d)(1-\tilde{r}_j^u)r_l^d \\
&+\sum_{l=1}^{L}\sum_{t=1}^{j-1} p(N,\Delta_t^u,\Delta_l^d)z_{t,j}^u r_l^d \\
&+\sum_{l=1}^{L} p(N,\Upsilon^u,\Delta_l^d)q_j^u r_l^d
\end{aligned} \tag{B.31}$$

$$\begin{aligned}
p(N-1,\Upsilon^u,\Delta_l^d) &= \sum_{j=1}^{J} p(N-2,\Delta_j^u,\Delta_l^d)r_j^u(1-\tilde{r}_l^d) \\
&+\sum_{j=1}^{J}\sum_{x=1}^{l-1} p(N-2,\Delta_j^u,\Delta_x^d)r_l^u z_{x,l}^d \\
&+\sum_{j=1}^{J} p(N-2,\Delta_j^u,\Upsilon^d)r_j^u p_l^d \\
&+p(N-2,\Upsilon^u,\Delta_l^d)(1-P^u)(1-\tilde{r}_l^d) \\
&+\sum_{x=1}^{l-1} p(N-2,\Upsilon^u,\Delta_x^d)(1-P^u)z_{x,l}^d
\end{aligned}$$

$$+p(N-2,\Upsilon^u,\Upsilon^d)(1-P^u)p_l^d \tag{B.32}$$

$$\begin{aligned} p(N-1,\Upsilon^u,\Upsilon^d) &= \sum_{j=1}^{J}\sum_{l=1}^{L} p(N-1,\Delta_j^u,\Delta_l^d)r_j^u r_l^d \\ &+\sum_{l=1}^{L} p(N-1,\Upsilon^u,\Delta_l^d)(1-P^u)r_l^d \\ &+\sum_{l=1}^{L} p(N,\Upsilon^u,\Delta_l^d)(1-Q^u)r_l^d \\ &+p(N-1,\Upsilon^u,\Upsilon^d)(1-P^u)(1-P^d) \end{aligned} \tag{B.33}$$

$$\begin{aligned} p(N,\Delta_j^u,\Delta_l^d) &= p(N,\Delta_j^u,\Delta_l^d)(1-\tilde{r}_j^u)(1-\tilde{r}_l^d) \\ &+\sum_{t=1}^{j-1} p(N,\Delta_t^u,\Delta_l^d)z_{t,j}^u(1-\tilde{r}_l^d) \\ &+\sum_{x=1}^{l-1} p(N,\Delta_j^u,\Delta_x^d)(1-\tilde{r}_j^u)z_{x,l}^d \\ &+\sum_{t=1}^{j-1}\sum_{x=1}^{l-1} p(N,\Delta_t^u,\Delta_x^d)z_{t,j}^u z_{x,l}^d \\ &+p(N,\Upsilon^u,\Delta_l^d)q_j^u(1-\tilde{r}_l^d) \\ &+\sum_{x=1}^{l-1} p(N,\Upsilon^u,\Delta_x^d)q_j^u z_{x,l}^d \end{aligned} \tag{B.34}$$

$$p(N,\Delta_j^u,\Upsilon^d) = 0 \tag{B.35}$$

$$\begin{aligned} p(N,\Upsilon^u,\Delta_l^d) &= \sum_{j=1}^{J} p(N-1,\Delta_j^u,\Delta_l^d)r_l^u(1-\tilde{r}_l^u) \\ &+\sum_{j=1}^{J}\sum_{x=1}^{l-1} p(N-1,\Delta_j^u,\Delta_x^d)r_j^u z_{x,l}^d \\ &+\sum_{j=1}^{J} p(N,\Delta_j^u,\Delta_l^d)r_j^u(1-\tilde{r}_l^d) \\ &+\sum_{j=1}^{J}\sum_{x=1}^{l-1} p(N,\Delta_j^u,\Delta_x^d)r_j^u z_{x,l}^d \end{aligned}$$

$$\begin{aligned}
&+\sum_{j=1}^{J} p(N-1,\Delta_j^u,\Upsilon^d) r_j^u p_l^d \\
&+p(N-1,\Upsilon^u,\Delta_l^d)(1-P^u)(1-\tilde{r}_l^d) \\
&+\sum_{x=1}^{l-1} p(N-1,\Upsilon^u,\Delta_x^d)(1-P^u) z_{x,l}^d \\
&+p(N,\Upsilon^u,\Delta_l^d)(1-Q^u)(1-\tilde{r}_l^d) \\
&+\sum_{x=1}^{l-1} p(N,\Upsilon^u,\Delta_x^d)(1-Q^u) z_{x,l}^d \\
&+p(N-1,\Upsilon^u,\Upsilon^d)(1-P^u) p_l^d \qquad (B.36)
\end{aligned}$$

$$p(N,\Upsilon^u,\Upsilon^d) = 0 \qquad (B.37)$$

B.4.1 Solution to Boundary State Equations

The complete solution technique for the boundary state equations will not be pursued, however it is noteworthy to state that we expect the solutions to the ones found in Chapter 3 plus a linear combination of terms dependent on the z's. In other words, $P_A(state) = P_3(state) + \sum_{m=1}^{R} C_m F_m(state)$, where P_A represents the solution to the states as proposed in this appendix, P_3 is the found solution as explained in Chapter 3, and F_m is an unknown function.

Since $p(2,\Upsilon^u,\Delta_l^d)$ and $p(N-2,\Delta_j^u,\Upsilon^d)$ are internal states, their probabilities can be expressed as part of the internal solution. Also $p(1,\Delta_j^u,\Upsilon^d)$ and $p(1,\Delta_j^u,\Delta_l^d)$, being the right side members of (1), (3) and (4) which are composed of only internal steady state probabilities, can be expressed as internal solutions as well . Therefore, it is possible, by using the boundary and internal state equations, to express the probabilities of the boundary states as a function of the constants $C_1,...,C_m$ as shown below:

$$p(0,\Upsilon^u,\Upsilon^d) = 0 \qquad (B.38)$$

$$p(0, \Upsilon^u, \Delta_l^d) = 0 \tag{B.39}$$

$$p(1, \Delta_j^u, \Upsilon^d) = \sum_{m=1}^{R} C_m X_m U_{j,m} \tag{B.40}$$

$$p(1, \Delta_j^u, \Delta_l^d) = \sum_{m=1}^{R} C_m X_m U_{j,m} D_{l,m} \tag{B.41}$$

$$p(2, \Upsilon^u, \Delta_l^d) = \sum_{m=1}^{R} C_m X_m^2 D_{l,m} \tag{B.42}$$

Similarly to the lower boundary states, some of the upper boundary states have an internal form, like are $p(N-1, \Delta_j^u, \Delta_l^d)$, and $p(N-1, \Upsilon^u, \Delta_l^d)$.

$$p(N-2, \Delta_j^u, \Upsilon^d) = \sum_{m=1}^{R} C_m X_m^{N-2} U_{j,m} \tag{B.43}$$

$$p(N-1, \Delta_j^u, \Delta_l^d) = \sum_{m=1}^{R} C_m X_m^{N-1} U_{j,m} D_{l,m} \tag{B.44}$$

$$p(N-1, \Upsilon^u, \Delta_l^d) = \sum_{m=1}^{R} C_m X_m^{N-1} D_{l,m} \tag{B.45}$$

$$p(N, \Upsilon^u, \Upsilon^d) = 0 \tag{B.46}$$

$$p(N, \Delta_j^u, \Upsilon^d) = 0 \tag{B.47}$$

The missing equations remain to be derived.

Bibliography

[1] A. M. Bonvik. *Performance Analysis of Manufacturing Systems Under Hybrid Control Policies*. PhD thesis, Massachusetts Institute of Technology, 1996.

[2] Mitchell H. Burman. *New Results in Flow Line Analysis*. PhD thesis, Massachusetts Institute of Technology, 1995.

[3] John A. Buzacott and J. George Shanthikumar. *Stochastic Models of Manufacturing Systems*. Prentice Hall, 1993.

[4] Y. Dallery, R. David, and X.L. Xie. An efficient algorithm for analysis of transfer lines with unreliable machines and finite buffers. *IIE Transactions*, 20(3), 1988.

[5] Y. Dallery, R. David, and X.L. Xie. Approximate analysis of transfer lines with unreliable machines and finite buffers. *IEEE Transactions on Automatic Control*, 34, 1989.

[6] Y. Dallery and S. B. Gershwin. Manufacturing flow line systems: A review of models and analytical results. *Queuing Systems Theory and Applications, Special Issue on Queuing Models and Manufacturing Systems*, 12, 1992.

[7] Alan Drake. *Fundamentals of Applied Probability Theory*. McGraw-Hill Publishing Company, Inc., 1988.

[8] Yannick Frein, Christian Commault, and Yves Dallery. Modeling and analysis of closed-loop production lines with unreliable machines and finite buffers. Technical report, Laboratoire d'Automatique de Grenoble, 1992.

[9] Stanley B. Gershwin. An efficient decomposition method for the approximate evaluation of tandem queues with finite storage space and blocking. *Operations Research*, March 1987.

[10] Stanley B. Gershwin. Representation analysis of transfer lines with machines that have different processing rates. *Annals of Operations Research*, 9, 1987.

[11] Stanley B. Gershwin. Assembly/disassembly systems: An efficient decomposition algorithm for tree-structured networks. *IIE Transactions*, 23(4), 1991.

[12] Stanley B. Gershwin. *Manufacturing Systems Engineering*. PTR Prentice Hall, 1994.

[13] Stanley B. Gershwin. Design and operation of manufacturing systems - control and system-theoretical models and issues. Technical report, Massachusetts Institute of Technology, 1997.

[14] Stanley B. Gershwin and O. Berman. Analysis of transfer lines consisting of two unreliable machines with random processing times and finite buffers. *AIIE Transactions*, 13(1), 1981.

[15] Stanley B. Gershwin and I. Schick. Continuous model of an unreliable two-state material flow system with a finite interstage buffer. Technical report, Massachusetts Institute of Technology, 1980.

[16] Stanley B. Gershwin and James E. Schor. Efficient algorithms for buffer space allocation. Technical report, Massachusetts Institute of Technology, 1997.

[17] Joseph Nemec. *Diffusion and Decomposition Approximations of Stochastic Models of Multiclass Processing Networks.* PhD thesis, Massachusetts Institute of Technology, 1999.

[18] T. Tolio, S. B. Gershwin, and A. Matta. Analysis of two-machine lines with multiple failure modes. Technical report, Politecnico di Milano, 1998.

[19] T. Tolio and A. Matta. A method for performance evaluation of automated flow lines. Technical report, Politecnico di Milano, 1998.