An Integrated Optimization Tool with Applications in Mining Using a Discrete Rate Stochastic Model

by

Asim Khan

Submitted in partial fulfilment of the requirements for the degree of Doctor of Philosophy at

Dalhousie University
Halifax, Nova Scotia
November 2011

© Copyright by Asim Khan, 2011
The undersigned hereby certify that they have read and recommend to the Faculty of Graduate Studies for acceptance a thesis entitled “An Integrated Optimization Tool with Applications in Mining Using a Discrete Rate Stochastic Model” by Asim Khan in partial fulfilment of the requirements for the degree of Doctor of Philosophy.

Dated: November 28, 2011

Supervisor: _________________________________
Readers: _________________________________

_________________________________

Departmental Representative: _________________________________
DATE: November 28, 2011

AUTHOR: Asim Khan

TITLE: An Integrated Optimization Tool with Applications in Mining using a Discrete Rate Stochastic Model

DEPARTMENT OR SCHOOL: Department of Civil and Resource Engineering

DEGREE: PhD CONVOCATION: May YEAR: 2012

Permission is herewith granted to Dalhousie University to circulate and to have copied for non-commercial purposes, at its discretion, the above title upon the request of individuals or institutions. I understand that my thesis will be electronically available to the public.

The author reserves other publication rights, and neither the thesis nor extensive extracts from it may be printed or otherwise reproduced without the author’s written permission.

The author attests that permission has been obtained for the use of any copyrighted material appearing in the thesis (other than the brief excerpts requiring only proper acknowledgement in scholarly writing), and that all such use is clearly acknowledged.

________________________________________________________
Signature of Author
# TABLE OF CONTENTS

LIST OF TABLES ........................................................................................................................................ viii

LIST OF FIGURES ....................................................................................................................................... ix

ABSTRACT .................................................................................................................................................. xii

LIST OF ABBREVIATIONS AND SYMBOLS USED ................................................................................ xiii

ACKNOWLEDGEMENTS .............................................................................................................................. xiv

CHAPTER 1: INTRODUCTION .................................................................................................................... 1

1.1 OVERVIEW ......................................................................................................................................... 1

1.2 BACKGROUND & MOTIVATION ....................................................................................................... 1

1.3 STATEMENT OF THE PROBLEM ................................................................................................. 1

1.3.1 Location And Severity Of Bottlenecks ..................................................................................... 2

1.3.2 Localized Optimization ................................................................................................................. 2

1.3.3 Discrepancy Between Optimization Levels .................................................................................. 2

1.4 OBJECTIVE OF THE THESIS ............................................................................................................ 3

1.5 METHODOLOGY ............................................................................................................................... 4

1.5.1 General ......................................................................................................................................... 4

1.5.2 Model ............................................................................................................................................ 4

1.5.3 Field Testing ................................................................................................................................. 5

1.6 STYLE, STRUCTURE, AND SCOPE OF THE THESIS ........................................................................ 6

1.6.1 Thesis Style And Format ............................................................................................................... 6

1.6.2 Structure And Organization ....................................................................................................... 7

1.6.3 Scope Of The Thesis ...................................................................................................................... 8

1.7 OPTIMIZATION LITERATURE RESEARCH ...................................................................................... 8

1.7.1 Introduction To Optimization ....................................................................................................... 8

1.7.2 History Of Optimization .............................................................................................................. 9

1.7.3 Optimization Techniques ............................................................................................................. 10

1.7.4 Applications Of Stochastic Optimization In Complex Integrated Mining Operations ................. 12
LIST OF TABLES

Table 4.1 – Scoring of assessed Simulation Software according to the evaluation criteria...24
LIST OF FIGURES

2.1 Sources of Variations present within any process 14
2.2 Illustration of day to day common cause variation 14
2.3 Effect of Special Cause Variation to a process outcome; within a day, and day to day 15
3.1 Illustration of the components and flow of information within the SIPOC system 19
3.2 An example of modeling processes in series using SIPOC 20
3.3 Illustration of lateral and causal mapping of a smelter facility 22
5.1 Illustration of Amorphous inputs into the model 33
5.2 An illustration of Defined inputs into the model 34
5.3 An illustration of Empirical inputs into the model 35
5.4 An illustration of multi model distribution 36
5.5 Production Output: a histogram depicting sh. tons skipped to a mill 37
5.6 Bar Chart Output: an illustration of day to day constraint of various processes. 38
5.7 Scatter Chart Output: an illustration of sh. tons constrained by various process over a duration of 1 year (365 days). 39
5.8 Column Chart Output: an illustration of the severity of the constrained processes. 40
5.9 Capacity Constraint Output: an illustration of ore bins backing up representing process in front being a constraint. 41
5.10 Cumulative Bottleneck Output: a fictional illustration of cumulative bottlenecks of various processes. 42
5.11 A flowchart of steps involved in building and simulating the model 45
<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.16</td>
<td>Cumulative Bottlenecks output as identified by the model (Base Case)</td>
<td>72</td>
</tr>
<tr>
<td>6.17</td>
<td>Throughput Histogram of the Mill (Grinding Mills Initiative)</td>
<td>73</td>
</tr>
<tr>
<td>6.18</td>
<td>Severity of Bottlenecks within the mill as identified by the model (Grinding mills initiative)</td>
<td>73</td>
</tr>
<tr>
<td>6.19</td>
<td>Process Map of a Vale Inco Refinery Facility</td>
<td>76</td>
</tr>
<tr>
<td>6.20</td>
<td>Throughput Histogram of the Refinery (Base Case)</td>
<td>77</td>
</tr>
<tr>
<td>6.21</td>
<td>Throughput Histogram of the Refinery (Tank house Lean initiative)</td>
<td>78</td>
</tr>
<tr>
<td>6.22</td>
<td>Throughput Histogram of Mine (Tram initiative)</td>
<td>79</td>
</tr>
<tr>
<td>6.23</td>
<td>Severity of bottlenecks within the mine (Tram initiative)</td>
<td>80</td>
</tr>
<tr>
<td>6.24</td>
<td>Throughput Histogram of the Mill (Tram initiative)</td>
<td>81</td>
</tr>
<tr>
<td>6.25</td>
<td>Throughput Histogram of the Smelter (Tram initiative)</td>
<td>82</td>
</tr>
<tr>
<td>6.26</td>
<td>Cumulative Bottlenecks within Smelter as identified by the model (Base Case)</td>
<td>83</td>
</tr>
<tr>
<td>6.27</td>
<td>A snapshot of casual mapping of convertor process in ExtendSIM</td>
<td>84</td>
</tr>
<tr>
<td>7.1</td>
<td>An example of Scenario Analysis and Robustness of the model</td>
<td>87</td>
</tr>
<tr>
<td>7.2</td>
<td>Filtering of Data: an illustration of before &amp; after variation affecting actual performance of a process.</td>
<td>88</td>
</tr>
</tbody>
</table>
ABSTRACT

The simulation as a stand alone optimization tool of a complex system such as a vertical integrated mining operation, significantly over simplifies the actual picture of the system processes involved resulting in an unaccountable effort and resources being spent on optimizing Non Value Added (NA) processes.

This study purposed to develop a discrete stochastic simulation-optimization model to accurately capture the dynamics of the system and to provide a structured way to optimize the Value Added (VA) processes.

The mine operation model to be simulated for this study is designed as a hybrid level throughput model to identify the VA processes in a mining operation. This study also allows a better understanding of the impact of variation on the likelihood of achieving any given overall result.

The proposed discrete stochastic simulation- optimization model provides the ability for a process manager to gain realistic understanding of what a process can do if some factors constraining the process were to be optimized i.e. to conduct what-if analysis. Another benefit of this approached technique is to be able to estimate dependable and reasonable returns on a large optimization related expenditure.

The inputs into the model are the capability of the processes which are entered using various variables depending on how much information is available; simple inputs for least amount of information to detailed inputs for well known process to combinational inputs for somewhere in between. The process bottlenecks are identified and measured using the outputs of the model which include production output, severity of constraints, capacity constraints and cumulative bottleneck plots. Once a base case has been identified and documented then the inputs can be modified to represent the business initiatives and the outputs can be compared to the base case to evaluate the true value of the initiative.
# LIST OF ABBREVIATIONS AND SYMBOLS USED

<table>
<thead>
<tr>
<th>Symbols</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>5M+E</td>
<td>Used to represent the six sources for process variation.</td>
</tr>
<tr>
<td>A</td>
<td>Used in scoring criteria as &quot;Average&quot;</td>
</tr>
<tr>
<td>ARENA</td>
<td>A simulation software by Rockwell Automation Inc.</td>
</tr>
<tr>
<td>CI</td>
<td>Confidence Interval</td>
</tr>
<tr>
<td>ExtendSIM</td>
<td>A simulation software by Imagine That Inc.</td>
</tr>
<tr>
<td>FGS</td>
<td>Faculty of Graduate Students</td>
</tr>
<tr>
<td>G</td>
<td>Used in scoring criteria as &quot;Good&quot;</td>
</tr>
<tr>
<td>LHD</td>
<td>Load-Haul-Dump underground mining vehicle</td>
</tr>
<tr>
<td>P</td>
<td>Used in scoring criteria as &quot;Poor&quot;</td>
</tr>
<tr>
<td>PERT</td>
<td>Performance Evaluation Review Technique</td>
</tr>
<tr>
<td>SIMUL8</td>
<td>A simulation software by SIMUL8 Corp.</td>
</tr>
<tr>
<td>SIPOC</td>
<td>Supplier-Input-Process-Output-Customer</td>
</tr>
<tr>
<td>Six Sigma</td>
<td>A business management strategy</td>
</tr>
<tr>
<td>TOC</td>
<td>Theory of Constraints</td>
</tr>
</tbody>
</table>
ACKNOWLEDGEMENTS

Acknowledgements go to my supervisor, Dr. Maria Rockwell; I want to thank her for her invaluable advice, enormous time and effort. There were times and hurdles where without her motivational guidance and resourcefulness, this thesis would not have been accomplished. For that, I want to be on record that I am unreservedly grateful to her for being my guiding light in the end of a very long tunnel.

I would also like to acknowledge my co-supervisor, Dr. Steve Zou, and the committee member, Dr. George Jarjoura for their invaluable technical discussions and suggestions. I would also like to thank Vale Limited and specifically, Kyle Gimpl and Carmine Ciriello for providing me with the vision and the opportunity to conduct this research in an industrial setting.

A number of financial scholarships were utilized to achieve the goals of this thesis for which I am thankful to the Department of Civil and Mineral Resource, Dalhousie University and specifically the Dean of Engineering, Dr. Joshua Leon.

I would also like to acknowledge the members, staff, students and lecturers in the mining engineering department for their support and to Dalhousie University for providing me the opportunity to study at the university.

I am also grateful to my parents, family, and friends for their unconditional support and help in making this thesis a success.

Finally, the greatest appreciation is to the only SUPREME ALMIGHTY GOD who is most graceful and merciful. What has been accomplished in this research is just by HIS grace and mercy.
CHAPTER 1: INTRODUCTION

1.1 Overview

This chapter gives the background and motivation of this thesis and specifically defines the problems that lead to the research. It presents the specific problems that were solved. It also outlines the objectives of the thesis and explains the methodology involved in building the tool used to achieve the objectives of this thesis. The methodology underlying the field testing of the tool is also given. The style, outline, organization and scope of the thesis are also discussed.

1.2 Background & Motivation

This research was motivated by the need to better evaluate and to develop better business cases intended to optimize mine production processes. In past and even today, many mine operations face the predicament of whether to implement an optimization initiative or not as most of the initiatives do not deliver on the promised optimization.

A tool is needed to first identify where the optimization efforts should be focused and then to adequately gauge the return that an optimization initiative will deliver.

1.3 Statement of the Problem

The main problem deals with spending capital/resources on non-optimum process initiatives. In other words, projects are undertaken to optimize processes which do not work towards achieving the goal (i.e. make money\(^1\)). This problem can be related into three main issues:

Myths / beliefs about location and severity of bottleneck
Localized optimization
Various levels of optimization

1.3.1 Location and Severity of bottlenecks

This problem exists in many industry operations today\textsuperscript{2}. Usually, operations chase the bottlenecks instead of identifying them. An optimization solution is generally implemented for the perceived issue; bottlenecks moves within the system and the end result is no net optimization.

1.3.2 Localized Optimization

This problem usually exists in an integrated operation. Usually a part of the integrated operation will conduct a local optimization without realizing the affect of this optimization on the rest of the system. Problem arises when a business initiative which promises to increase the throughput (money) fails to do so due to no understanding of the bottlenecks within the integrated system\textsuperscript{3}.

1.3.3 Discrepancy between Optimization Levels

This problem is very common in modeling industry today\textsuperscript{4}. Problem occurs as various processes are modeled with different scientific methodologies. The problem is not in modeling itself but lies in the communication between various methodologies. It becomes very complex and time consuming to connect the model inputs/outputs built on different levels such as communication between models built with chemical parameters to a model built with thermodynamic parameters.
1.4 Objective of the thesis

The main objective of this research is to build a tool/model to identify the bottlenecks in an integrated mining operation. Following are the sub objectives intended to be attained by the tool develop in this research:

- Validate improvement initiatives to address process bottleneck
- Analyze the flow behavior of bottlenecks in the models by removing/optimizing the constraints.
- Provide the ability for a process manager to gain a realistic understanding of what a process is capable of producing by regulating the capacity and/or the variation in a certain area i.e. to conduct what-if analysis.
- Analyze the return of large throughput related capital expenditures.
- Provide a tool to evaluate the return of a business initiative in terms of a business goal.
- Observe the performance of throughput processes by regulating variability of the inputs
- The ability to identify the hot spots where further analysis is needed which can then be achieved by constructing more detailed models of the relative areas.
- To identify any process capacity waste in a system.
1.5 Methodology

1.5.1 General

The research objectives were achieved by a combination of (i) literature review of the principals behind the stated problems researched such as Process Variation and Theory of Constraints; (ii) discussions and consultation with six-sigma and continuous improvement experts on resolving these problems and modeling techniques; (iii) creation and utilization of discrete rate simulation modeling; and (iv) field testing of the model built. The research strategy investigated the fundamental principles underlying the difficulties faced by mining operations and techniques currently employed by industry experts. The specific methodologies are given in the following section.

1.5.2 Model

The model is designed as discrete rate flow model using ExtendSIM simulation software. Simulation is one of the most widely applied techniques of management science. Simulation tools can evaluate the efficiency and possible drawbacks of certain options before they are actually implemented in practice, thereby playing a crucial role in the evaluation process\(^5\). Today, a number of simulation models have been developed to optimize any aspect of mining related system activities. However, most of these models were developed either for specific applications or developed at various levels of design. If the model is designed to optimize a specific activity or process then the effect of the optimized solution cannot be monitored throughout the whole system. For example, in a Mining Operation, there are various unit operations such as drilling, mucking and hoisting. If mucking operation was specifically modeled to be optimized then the optimized solution may result in hoisting operation to become a bottleneck or may cause
a drilling operation to become a constraint as the capacity of drill area may not produce enough muck for the optimized operation to be utilized efficiently. This narrow focus optimization results in “Rolling Bottlenecks” with potential of effort, resources and money being spent on non bottleneck processes. This is waste because none or very little of the improvement makes it all the way to the final product.

Similarly, if a model is designed for the whole system but at different levels of designs then it becomes quite complex to link the various optimized solutions. For example, in a Milling & Smelting Operation, there are various unit operations such as Ore Recovery, Roasting, Casting etc. Now, if recovery model was designed chemically, roasting model was designed thermodynamically and casting model was designed mechanically then it will require tremendous amount of time and effort to link the optimized solution of each model.

Thus, to overcome these shortcomings of current practice of simulation models, this model is designed at the throughput level for the integrated mining operations. The important processes are identified in the entire plant or asset. Every process is treated according to SIPOC Model. Inputs into the processes and the capacity of processes are simulated stochastically using a distribution. Any factor which may affect the flow of the model can be entered into the model by linking it to the throughput.

1.5.3 Field Testing

The field testing started with mapping of the operation processes and collection of the data for the purpose of estimating historical process capability distributions. This field work was done at an integrated Nickel mining Vale Inco Limited operation. Field testing was conducted at an underground mine, a mill facility, a smelter facility, and a refinery facility.
Four separate models were built in such a way that they could either be utilized as standalone modules or as an integrated tool. Once the model was completed and validated for the flow of the processes, then further data was captured either through creation or utilization of an existing database to filter out the effect of misleading variation with the historical process data. This data was also collected through discussions with process experts if the collection of variation data was not plausible.

Next, the input distributions into the model were refined to reflect individual process variation. The model was re-validated against the historical data and significant production events. Once the model validation was adequately confirmed then the bottlenecks in every operation were individually identified. Next, various business initiatives were evaluated using this tool. The integrated Vale Inco mining operation has mandated the use of this tool for future throughput related business initiatives.

1.6 Style, Structure, and Scope of the thesis

1.6.1 Thesis Style and Format

Thesis style and format follows the Faculty of Graduate Students (FGS) thesis formatting guidelines. Language format used is U.S. English. A style appropriate to subject matter is followed throughout the thesis. The thesis document is printed on one sided 21.5 × 28cm (8.5" × 11"), portrait orientation. Left hand side margins are 38mm (1.5") wide. All other margins are at least 25mm (1") wide. Text for main body of the thesis is in a standard 12pt, Times New Roman font. The title of the thesis and the title for all entries in Table of Contents are cased. The order of items in entire thesis follows the FGS guidelines.
1.6.2 Structure and Organization

The general structure and organization of the thesis consists of seven chapters. Chapter 1 is the introductory chapter which outlines the background and motivation of the research and defines specifically the problem statement that led to the research. It highlights the aims, objectives and the methods used to achieve the objective. Scope, structure and organization of the thesis are also presented.

Chapter 2 gives is based on the literature review of principles underlying the cause of process bottlenecks. Focused topics are process variation and theory of constraints (TOC). The significance of these principles to the thesis is also presented.

Chapter 3 gives a review of methods and techniques essential for building a bottleneck model. It elaborates on the exchange of information gathered through discussions and consultations with process optimization industry six sigma and continuous improvement experts.

Chapter 4 gives the detail of the simulation software assessment that was conducted during this research. Software evaluation criteria, scoring and then selection of the software are covered in this chapter.

Chapter 5 deals with the design of the throughput bottleneck model. It details the theory, logic and systems used in building of the model. Model components, inputs, outputs, and general usage are detailed in this chapter.

Chapter 6 gives an account of testing of the model in the field. Advantages and extent of the use of the model are also covered through case studies in this chapter.
Chapter 7; the last chapter discusses the robustness of the model and of possible pitfalls one must avoid when utilizing this model. Recommendations for further research and conclusions are also presented in this chapter.

Appendices give details about the purpose, the usage and options of various ExtendSIM blocks used in development of the bottleneck model.

1.6.3 Scope of the thesis

The thesis was initially limited to the identification of the bottlenecks in an integrated mining operation. However, due to TOC and complexity of process variation, it became pertinent to broaden the scope to investigate the movement of bottlenecks and to estimate the severity of bottlenecks on the output of an integrated mining operation. Once the model was completed and then tested, it became apparent that the model could not only be used for original scope but also to investigate other aspects of process optimization; such as Causal Modeling and Lean Optimization.

1.7 Optimization Literature Research

1.7.1 Introduction to Optimization

Optimization in its basic roots a mathematical term also referred as mathematical programming. The idea is to either minimize (usually cost) or maximize (usually profit) a mathematical function by logically analyzing the solution for all possible scenarios within a tolerable set. It is generally difficult to develop an optimized model that tackles all characteristics of the quandary and its surroundings. It is like trying to best fit a known analyzable geometrical shape into an unknown irregular shape. Thus there has been
various different optimization techniques developed to best fit the real problem. However, it will be beneficial to first analyze a general optimization function then to get into various optimization techniques.

A general optimization model consists of three components: 1) the objective function, 2) the constraints, and 3) the variables. Objective function defines what is that needs to be optimized and whether is it to be maximized or minimized. Constraints define the limits for the optimization model. In other words, they set finite number of solutions that could be optimized. Variables allow us to define the aspects of a problem in mathematical terms. Thus, an optimization problem can be expressed in mathematical terms as follows.

Objective Function: \( f = \{x\}; f(x0) \leq f(x) \) (minimization) or \( f(x0) \geq f(x) \) (maximization)
Constraints: all values of a real set
Variable: \( x \)

So, problems that seek to maximize or minimize a mathematical function of a number of variables, subject to certain constraints, are known as optimization problems. Optimization problems may involve more than one objective function and are known as multi-objective optimization problems. Depending on the nature of the problem, the variables in the model may be real or a mixture. The optimization problem could be either constrained or unconstrained. In the constraint part of a mathematical model, the left-hand side of the constraint function is separated from the right-hand-side value by one of the following three eventualities: (1) equal to =, (2) less than or equal to \( \leq \), or (3) greater than or equal to \( \geq \).

### 1.7.2 History of Optimization

Clear evidence of optimization being employed can be observed as early as in 1900 when Gantt optimized scheduling jobs on machines using charts known as Gantt Charts. In
1915, Harris mathematically optimized inventory management by developing an ordering model from a vendor. Today, it is known as economic order quantity model. In 1917, Erland optimized the switchboard calling process. This optimization process is better known as queuing theory. During World War II, first British and then Americans optimized their limited resources to be used in battlefield. This is where “Operations Research” study was first classified.

After World War II, the operations research and more importantly optimization was introduced in day to day business operations. Optimization techniques have been available for more than a century. In 1947, Dantzig designed an optimization algorithm to solve complex linear programming problems. This algorithm is known as Simplex. Simplex allowed the complex problems to be solved using computers. As the computing technology improved so did the power of Simplex algorithm. In addition to many other conventional optimization techniques developed over the past half-a-century (as will be discussed later), the recent development of modern heuristic techniques such as simulated annealing, tabu search, genetic algorithms, neural computing, fuzzy logic, and ant colony optimization are providing practitioners with some sophisticated tools to address more complex situations.

1.7.3 Optimization Techniques

Use of mathematical optimization to solve real life problems can be divided into two major groups: (1) the classical optimization techniques and (2) the modern heuristic techniques. There are various mathematical programming techniques in use today:

Linear programming (LP) – problems involve the optimization of a linear objective function, subject to linear equality and inequality constraints.
Integer programming – similar to linear programming but the unknown variables are all required to be integers.

Quadratic programming – similar to linear programming solving techniques, however the solution is quadratic as the objective function is defined as a quadratic.

Nonlinear programming – process of solving a system of equalities and inequalities, collectively termed constraints, over a set of unknown real variables, along with an objective function to be maximized or minimized, where some of the constraints and/or the objective function are nonlinear.

Convex programming – the case when the objective function is convex and the constraints, if any, form a convex set. This can be viewed as a particular case of nonlinear programming or as generalization of linear or convex quadratic programming.

Stochastic programming – the case in which some of the constraints or parameters depend on random variables.

Robust programming – same as stochastic programming, however the uncertainty is introduced by deliberated inaccurate input data.

Combinatorial optimization – problems where the set of feasible solutions is discrete or can be reduced to a discrete one.

Infinite-dimensional optimization studies the case when the set of feasible solutions is a subset of an infinite-dimensional space, such as a space of functions.
Heuristic algorithms – an algorithm that ignores whether the solution to the problem can be proven to be correct, but which usually produces a good solution or solves a simpler problem that contains or intersects with the solution of the more complex problem.

Constraints satisfaction – the case in which the objective function $f$ is constant. This is mostly reserved for automatic reasoning and is the basis behind Artificial Intelligence.

Disjunctive programming – the case where at least one constraint must be satisfied but not all. This is mostly used in schedule optimization.

Trajectory optimization – as the name suggest, it is used to optimize trajectories for air and space vehicles.

Calculus of variations – a part of dynamic optimization; an objective defined over many points in time, by considering how the objective function changes if there is a small change in the choice path. Optimal control optimization is generalization of this programming.

Dynamic programming – a method of solving problems where one needs to find the best decisions one after another.

1.7.4 Applications of Stochastic Optimization in Complex Integrated Mining Operations

There has been a lot of scholarly work done in the field of stochastic optimization when applied to mining such as one of the latest paper published by Raj$^{16}$. However, almost all of this effort has been limited to optimizing single production mining operations. Stochastic optimization is also generally used in mining operations when conducting risk analysis or dealing with uncertainty$^{17}$. Due to complexity of the integrated mining operations, there seems to be no single optimization technique or tool available today.
CHAPTER 2: REVIEW OF PRINCIPLES UNDERLYING PROCESS BOTTLENECKS

2.1 Overview

This chapter gives an overview of natural and unnatural variation that exists in every process. Sources and types of variations are also discussed. Finally, the significance of process variation to the tool developed in this research is also detailed.

This chapter also gives an overview of the theory of constraints (TOC) that applies to every system and specifically to a mining operation system. The section on TOC also elaborates on terms used in business improvement literature. Finally, the significance of TOC to the tool developed in this research is also detailed.

2.2 Process Variation

Variation is a natural phenomenon; it is everywhere. All processes are subject to variation in performance and thus no two outputs will ever be exactly same. Variation is caused by sub-variations within the process (Figure 2.1) i.e. variation caused by:

**Materials** e.g. variation in ore grade

**Manpower** e.g. variation between different shifts for same process

**Measurement** e.g. variation in grade measurement of same concentrate by composite vs. series sampling

**Machine** e.g. variation in throughput or quality from two mills of similar specifications

**Methods** e.g. variation between two methods to achieve same outcome such as ore grind size by rod and a ball mill

**Environment** e.g. variation in ore treatment caused by oxidation due to weather
There are two types of variations\(^8\); common cause and special (assignable cause). Common cause variation is an inherent part of the process (or system) design and execution; hour after hour, day after day and effect everyone in the process (Figure 2.2).
The histogram or the distribution in (Figure 2.2) represents the common cause variation within a process. Assignable cause variation is not part of the process all of the time and do not affect it all of the time but arises out of specific circumstances e.g. breakdown of a machinery. The shift and the unpredictable spread of the histogram in (Figure 2.3) represent the special or assignable cause variation.

![Figure 2.3 – Effect of Special Cause Variation to a process outcome; within a day, and day to day](image)

### 2.2.1 Significance of Process Variation to the objective of this research

From the understanding of the variation; it is correctly believed that reduction in variation will result in higher throughput and thus more money in sales. However, this gives birth to the common myth that if the work is done to reduce the variation in any process then there will be sufficient gain in the throughput. This is not entirely correct as if the process with reduced variation is not the bottleneck or the constrained process then the throughput gain will be mostly lost in the bottleneck as the constrained process is already running at full capacity and thus can neither receive more input nor provide more output i.e. cannot process anymore. The model developed through this research highlights this
phenomenon as well as shows the improvement of reducing variation when conducted properly.

Process variation is a major contributor to the performance of any process and thus the model is designed with variation in mind. Inputs into the models are based on the distributions instead of averages to observe the affect of variation in the process. The use of averages instead of full variation significantly skews the full picture as a model built on averages at best predicts how the process will perform 50% of the time.

2.3 Theory of Constraints
Dr. Eliyahu M. Goldratt introduced the Theory of Constraints (TOC) in his 1984 book, The Goal1. Theory postulates that any system is restricted from achieving its goal by one or at most very few constraints at a given time. Thus, the philosophy is to identify the constraint (bottleneck), exploit it, redistribute all resources around it, elevate it, and then keep track of the constraint movement. Exploitation relates to the maximum utilization of the constraint, while elevations relates to the availability. Tracking of bottleneck movement is essential as if the bottleneck has moved then the process has to start over as the old constraint is not the bottleneck anymore.

**Goal** here is in basic terms is to “make money” now and in future. It is not high productivity, efficiency, utilization or even low cost, if in the end a company is not achieving the goal i.e. making money. Thus, the theory measures an organization with three parameters in terms of goal.

**Throughput** – higher the throughput, more money a company makes through sales.
**Operating Expenses** – It is the money spent to keep the company going.
**Inventory** – money invested by the company to sell its products. For example, the entire inventory created is to ensure that the product is produced as demanded by the customer.
**Constraint** here is not limited to just a mathematical term as in general optimization but is anything that is preventing the company from getting more throughput (money from sales) i.e. bottleneck.

### 2.2.1 Significance of TOC to the objective of this research

TOC applies directly to the main underlying problem behind this research. If a constraint is not identified the resources (time, money, people) are spent on working /improving something which in the end doesn’t work to achieve the Goal. Another important fact that TOC highlights is that even though there is usually one or at most few bottlenecks are present at any given time but these bottlenecks can shift places or move within the system overtime. Thus, it becomes very important to know where this bottleneck might move to & how profound the impact will be.
CHAPTER 3: REVIEW OF METHODS & TECHNIQUES
ESSENTIAL FOR BUILDING A BOTTLENECK MODEL

3.1 Overview
This chapter gives an overview of SIPOC modeling system. The origin of the system is also acknowledged. Finally, the significance of SIPOC application to the tool developed in this research is also detailed.

This chapter also gives an overview of the process mapping; a technique used by optimization industry experts. Utilization and types of process mapping are also discussed. Finally, the significance of mapping to the tool developed in this research is also detailed.

3.2 SIPOC
SIPOC; an acronym for Suppliers → Inputs → Process → Outputs → Customers, is a high level diagram of a process and a deduced version of a process map⁹. It helps in understanding the scope of a process (Figure 3.1).
SIPOC is a business management (such as six-sigma) mapping technique which represents an organized set of connected parts or activities that take inputs and transform and/or transfer them to produce a set of outputs. The definition of each SIPOC components is given below:

**Supplier** – provide inputs to the process. These could be the customer of the previous process in the sequence.

**Inputs** – these are the inputs usually the material, service and/or information that are used by the process to produce the outputs.

**Process** – sequence of activities, usually adds value to inputs to produce outputs for the customers.

**Outputs** – these are the outputs usually the products, services, and/or information that are valuable to the customers.
Customer – usually are the users of the outputs produced by the process. These in turn become the suppliers for the next process.

3.2.1 Significance of SIPOC to the objective of this research

The model built in the research employs the SIPOC principal for every process modeled. This can be best understood using an example elaborated in (Figure 3.2).

In this example, we have three processes. Every process is represented as a supplier and as a customer in the model. When crusher is the process then mine is the supplier which supplies ore to the crusher which processes it and outputs the crushed ore to the customer i.e. grinding mills. Now, Grinding mills are the process and the crusher is the supplier which supplied crusher rock as an input to the process which in turn processes it and outputted the ground ore to the customer i.e. Flotation. So Flotation received the input from the supplier (Mills) and processed it and outputted the concentrate to its customer which is Smelter. It is essential to build the model on this principal as it allows not only in identifying which process is the bottleneck but also recognizes the actual component of the system which is becoming the constraint. Just to clarify, even though above example depicts processes in series, same SIPOC terminology can be applied to processes in parallel as the processes in parallel will still have inputs by suppliers and outputs to customers.
3.3 Process Mapping

It is a 6σ technique utilized to understand the organization and the performance of a process. A mining operation is full of processes, not only technical processes such as mining, milling, smelting, refining, etc. but also administrative, marketing and managerial processes such as purchasing, warehousing, manpower, sales, handling orders.

A process map usually gives a 2 dimensional picture of a process; lateral and causal. Lateral or alignment view elaborates the relative position of a process to other processes on the same level. It is usually used to describe the relationship between the described outputs of a process and the parameters that impact those outputs (Figure 3.3). Causal or analytical view elaborates the details of sub processes or productive units within the focused process. It is usually used to build a hypothesis for improving the performance of a process.
Process mapping differs from a flowchart by creating a hypothesis describing the current best understanding of the relationship between the desired outputs of a process and parameter that impacts those outputs.

### 3.3.1 Significance of Process Mapping to the objective of this research

One of the steps in building the model researched in this thesis is to map the plant or the operation to identify the bottleneck(s). This is important as it gives a visual representation of the whole process and allows in achieving an appropriate balance with respect to the level of detail incorporated in the model. Too much detail unnecessarily consumes the analyst’s time. It may also hamper the tractability of attaining a solution to the model or realizing extended analytical objectives such as mathematical optimization. Conversely, too little detail may result in a model that is an abstraction of little relevance to the problem at hand.
CHAPTER 4: SIMULATION SOFTWARE ASSESSMENT

4.1 Overview

This chapter gives an overview of the simulation software assessment. The Evaluation criteria for software assessment are discussed. Finally, the scoring of the software assessed is also detailed.

4.2 Simulation Software

In the optimization world, simulation is a common tool used to understand how a process or system performs and would perform when modifications/changes are made, without the need to conduct expensive and time consuming trials. Unfortunately, size of many, if not most practical problems often make the use of simulation programming infeasible from computational perspective\(^\text{11}\). Commercial simulation & optimization framework software have emerged as alternative to help build simulation models. However, these software are designed as “one size fit all” situation simulation software and thus are not ideal for all industries. For the scope of this research, three software were assessed; ExtendSIM, SIMUL8, and ARENA. The evaluation criterion as described in a paper\(^\text{12}\) from Purdue University was used for assessing the software but the scoring was done by keeping the scope of this research in mind (Table 4.1).
Table 4.1 – Scoring of assessed Simulation Software according to the evaluation criteria

<table>
<thead>
<tr>
<th>Evaluation Criteria</th>
<th>Extend SIM</th>
<th>SIMUL8</th>
<th>ARENA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Model Building Structure</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hierarchical model</td>
<td>GOOD</td>
<td>GOOD</td>
<td>GOOD</td>
</tr>
<tr>
<td>Accessibility</td>
<td>AVERAGE</td>
<td>GOOD</td>
<td>GOOD</td>
</tr>
<tr>
<td>Reusability</td>
<td>GOOD</td>
<td>GOOD</td>
<td>AVERAGE</td>
</tr>
<tr>
<td>User Defined Elements</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Extensibility</td>
<td>GOOD</td>
<td>GOOD</td>
<td>AVERAGE</td>
</tr>
<tr>
<td>Design Facility</td>
<td>GOOD</td>
<td>GOOD</td>
<td>AVERAGE</td>
</tr>
<tr>
<td>Interaction with Applications</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Internal Database</td>
<td>GOOD</td>
<td>GOOD</td>
<td>GOOD</td>
</tr>
<tr>
<td>External Databases</td>
<td>GOOD</td>
<td>GOOD</td>
<td>GOOD</td>
</tr>
<tr>
<td>Dynamic Model Updating</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Optimization</td>
<td>GOOD</td>
<td>GOOD</td>
<td>GOOD</td>
</tr>
<tr>
<td>Updating on the fly</td>
<td>GOOD</td>
<td>POOR</td>
<td>AVERAGE</td>
</tr>
<tr>
<td>Routing</td>
<td>GOOD</td>
<td>AVERAGE</td>
<td>GOOD</td>
</tr>
<tr>
<td>Miscellaneous</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Multiple Simulations</td>
<td>AVERAGE</td>
<td>AVERAGE</td>
<td>AVERAGE</td>
</tr>
<tr>
<td>Animation</td>
<td>GOOD</td>
<td>GOOD</td>
<td>AVERAGE</td>
</tr>
<tr>
<td>Built-in items</td>
<td>AVERAGE</td>
<td>AVERAGE</td>
<td>AVERAGE</td>
</tr>
<tr>
<td>Statistical Ability</td>
<td>GOOD</td>
<td>GOOD</td>
<td>GOOD</td>
</tr>
<tr>
<td>Model Protection</td>
<td>GOOD</td>
<td>POOR</td>
<td>GOOD</td>
</tr>
</tbody>
</table>
4.3 Evaluation Criteria

Following criteria was selected from Purdue University’s paper\textsuperscript{12} as these factors were essential to the effective development of the model.

\textit{4.3.1 Hierarchical Model}

This is the ability to capture various levels of details within a model. It is important to have this capability as it allows hiding unnecessary detail in a model when only conceptual design is required and conversely, allows showing the details when needed.

\textit{4.3.2 Accessibility}

This is the capability of a model to link various items in a model. Good modeling software must allow connecting different items in the background to keep the model less clustered. However, it must also have the ability to directly link and communicate between items when required. Here, items could be various processes, statistical or data calculation blocks etc.

\textit{4.3.3 Reusability}

It is the ability of using a model or item into another model.

\textit{4.3.4 Extensibility}

It is the capability of changing the model state rules depending on occurrence of an event. In other words, it is the ability to run a model with different rules at different times.
4.3.5 Design Facility

It is the ease of designing in the modeling environment. Least amount of coding required is classified as best.

4.3.6 Internal Database

This criterion partly relates to the capability of creating spreadsheets or database within the modeling software and partly to the ease of capturing the data to the database.

4.3.7 External Database

This criterion partly relates to the capability of capturing data to spreadsheets or database outside the modeling software such as in Excel and partly to the ease of capturing the data to the database.

4.3.8 Optimization Programming

This relates to the accessibility and extent to various Operation Research optimization techniques such as Linear Programming, Queuing policies, etc.

4.3.9 Updating on the Fly

This relates to the capability of changing or modifying a model within the simulation run.

4.3.10 Routing

It is the ability to change the path of the flow in a model. In other words, it is the ability to define how the material flows through a system.
4.3.11 Multiple Simulation

It is the ability to run multiple simulations at the same time.

4.3.12 Animation

It is the ability to visualize how the material is flowing within the system and how the processes are behaving. A good model should be able to show some extent of animation within the simulation run.

4.3.13 Built-in Items

This relates to the resourcefulness of the items (processes, statistical blocks, mathematical blocks, variables) provided as default with a simulation software.

4.3.14 Statistical Ability

It is the ability to define various statistical distributions and the ability to capture various statistical trends.

4.3.15 Model Protection

It relates to the user content protection provided by the software.

4.4 Simulation Software Selected

As outlined in the table 4.1, ExtendSIM scores most “GOOD” rating in evaluated criteria. Even though ExtendSIM was chosen to be used for building the model in this research, it must be noted SIMUL8 and ARENA can be used to build similar models. ExtendSIM
however allows the discrete rate simulation in addition to continuous and discrete event simulations. The discrete rate simulation allows better and much realistic simulation of mining processes.
CHAPTER 5: DESIGN OF THE THROUGHPUT BOTTLENECK MODEL

5.1 Overview

This chapter gives an overview of the theory behind the model developed for this research. Underlying system and mapping utilized in development of this tool are also discussed. Model logic, inputs and outputs are detailed. Construction and Simulation of the model is also detailed. Finally, the use and the scenario analysis are also covered in depth.

5.2 Theory

Simulation is one of the most widely applied techniques of management science. Simulation tools can evaluate the efficiency and possible drawbacks of certain options before they are actually implemented in practice, thereby playing a crucial role in the evaluation process\textsuperscript{13}. Today, a number of simulation models have been developed to optimize any aspect of mining related system activities. However, most of these models were developed either for specific applications or developed at various levels of design.

If the model is designed to optimize a specific activity or process then the effect of the optimized solution cannot be monitored throughout the whole system. For example, in a Mining Operation, there are various unit operations such as drilling, mucking and hoisting. If mucking operation was specifically modeled to be optimized then the optimized solution may result in hoisting operation to become a bottleneck or may cause a drilling operation to become a constraint as the capacity of drill area may not produce enough muck for the optimized operation to be utilized efficiently. This narrow focus optimization results in “Rolling Bottlenecks” with potential of effort, resources and
money being spent on non bottleneck processes. This is waste because none or very little of the improvement makes it all the way to the final product.

Similarly, if a model is designed for the whole system but at different levels of designs then it becomes quite complex to link the various optimized solutions. For example, in a Milling & Smelting Operation, there are various unit operations such as Ore Recovery, Roasting, Casting etc. Now, if recovery model was designed chemically, roasting model was designed thermodynamically and casting model was designed mechanically then it will require tremendous amount of time and effort to link the optimized solution of each model.

Thus, to overcome these shortcomings of current practice of simulation models, this model is designed at the throughput level for the whole Thompson Operations, from Mining to Refinery. The important processes are identified in the entire plant or asset. Every process is treated according to SIPOC Model. Inputs into the processes and the capacity of processes are simulated stochastically using a distribution. Any factor which may affect the flow of the model can be entered into the model by linking it to the throughput.

5.3 System

Every process in the model is based on the SIPOC system. Every parameter is either directly modeled or converted back to a throughput unit. If a model is designed for the whole system but at different levels of designs then it becomes quite complex to link the various optimized solutions. For example, in a Milling & Smelting Operation, there are various unit operations such as Ore Recovery, Roasting, Casting etc. Now, if recovery model was designed chemically, roasting model was designed thermodynamically and casting model was designed mechanically then it will require tremendous amount of time and effort to link the optimized solution of each model.
Thus, to overcome these shortcomings of current practice of simulation models, this model is designed at the throughput level for the integrated mining operation. The important processes are identified in the entire plant or asset. Every process is treated according to SIPOC Model. Inputs into the processes and the capacity of processes are simulated stochastically using a distribution. Any factor which may affect the flow of the model can be entered into the model by linking it to the throughput.

Advantage of keeping every parameter reporting in the same unit is that any type of process can now be linked together based on SIPOC system principle. For example Flotation or recovery is a chemical process but performance of it can be measured in increase/decrease of throughput. Other example could be of a roasting process which is a thermodynamic process but an optimization in the process can still be reported in throughput processed by the roaster. By deducing everything back to throughput, all processes can now easily be placed in the SIPOC system.

### 5.4 Mapping

Every plant is process-mapped before building the model. This helps in identifying the level of detail needed for the scope of the model. For example, if the aim is to find where the bottleneck sits in an integrated mine operation then first the whole operation would be mapped as mine → mill → smelter → refinery etc. Then, mine could be causally mapped into drill → blast → muck → crush → hoist etc. Next, the model can be run to identify the process which is the major bottleneck and then can be further causally mapped e.g. if the bottleneck is the mucking process then map could look like shovels (LHDs) → truck (trams) → conveyor → bins etc. This is the advantage of mapping the process laterally and causally before building the model as it allows getting results with minimum amount of effort and time.
5.5 Logic

Model is designed to simulate one day at a time over duration of one year (365 days). This was necessary as the objective was to capture day to day movement and quantity of the bottlenecks. There is a simulated stockpile with infinite capacity present behind every process to capture bottlenecks with exception of where a real stockpile (e.g. ore pass/bins etc.) may be present. This was designed so no constrained process may slow down a process behind it. It is important to note here that total flow through all processes will still be same if these simulated stockpiles were not present.

Simulated stockpiles are essential part of the model in identification of bottlenecks and in visualizing the impact of de-bottlenecking on the final product. Without these stockpiles, the user may be able to see that an initiative may not result in an improvement in the final product but would not be able to see what process(es) stopped that improvement. Hence, the simulated stockpiles play an important role in identifying which processes to optimize.

5.6 Inputs

Inputs are entered randomly but according to a distribution. In other words, the inputs are generated rationally not just randomly. Rational Random inputs as pure randomness will not represent the actual variation of the processes. Stochastic inputs add confidence in the optimized solutions and this can be achieved as probability distributions are either known or can be estimated. There are three main types of inputs entered in this model usually based on the distribution.

Amorphous
Defined
Empirical
5.6.1 Amorphous

Triangular distribution is generally used for this type of input as least amount of information is known about a process which occurs usually at the beginning of the model database building\textsuperscript{15}. Triangular distribution is based on PERT (Performance Evaluation Review Technique). Figure 5.1 depicts the only three values needed for these inputs which are Minimum (pessimistic), Maximum (optimistic), and Most likely (mean).

![Distribution Plotter window](image)

Figure 5.1 – Illustration of Amorphous inputs into the model

Distribution Plotter window can be accessed by pressing the “Plot Sample” button on Random Number window. Plotter illustrates the rational stochastic distribution.

5.6.2 Defined

This type of input is used when some data already exist about a process behavior. In some cases, these input distributions are used as the output from the previous process is observed to be behaving as well defined process distributions. Figure 5.2 depicts the only two values needed for these inputs for a normal distribution, which are Mean (average) and Std Dev (standard deviation). Refer to Section 2 for instructions on how to enter this input.
Empirical inputs are used when no known distribution will adequately fit the whole span of data and PERT distribution will significantly over or underestimate the probability density (Figure 5.3). An example will be when a process has a multi-model distribution. In this case, the data will be split into several bins of occurrences and then a frequency of occurrences will be calculated. From these bins and frequencies of occurrences an empirical table is inputted into model. Figure 5.3 depicts an empirical input capture from a model.
In the case depicted in Figure 5.3, the inputs will be entered using various distributions at a given time. In this case, 10% of the time input will be 0, 50% of the time input will be represented by normal distribution as represented in Window 1, 25% of the time input will be represented by triangular distribution as represented in Window 2, and 15% of the time input will be represented by uniform distribution as represented in Window 3. The advantage of using this type of input is that it can be used for any kind of distribution. The process depicted in Figure 5.4 has multiple distributions; one when the process is running optimally from 135 st to 270 st, and second when process is running at a slow rate due to slowdowns and shut downs as shown from 0 st to 105 st.
5.7 Outputs

There are several outputs which are generated from the simulating the model:
Production Output
Severity of Constraints
Capacity Constraints
Cumulative Bottleneck Plots

5.7.1 Production Output

This output illustrates a day to day production at the end of a model in a form of a histogram. The production output histogram allows a user to observe the distribution of produced result and aids in defining the input of sequential module. Histogram is also
helpful in comparing the results of the model to historical actual production results. Figure 5.5 depicts a histogram of ore feed skipped to mill captured by a mine model.

5.7.2 Severity of Constraints

This output allows a user to identify which processes became a constraint on a day to day basis and the severity of those constraints over a year. There are three types of plots that generated through simulating the model. They are:
5.7.2.1 BAR CHART

This plot is updated on day to day basis and is available while the simulation is running. This plot helps the user identify what process or combination of processes are the constraints for a given day. Also, it allows the user to observe the severity of the constraint in reference to the units of the model (e.g. sh. tons). Figure 5.6 depicts an example of this chart.

![Bar Chart Example](image1)

- a - Crusher, Conveyor & Flotation were the constraints this day.
- b - Crusher was the only constraint this day.
- c - every process except Flotation were the constraints this day.
- d - Crushers and Conveyors were the constraints this day.

• Figure 5.6 – Bar Chart Output: an illustration of day to day constraint of various processes.

5.7.2.2 SCATTER CHART

This plot is generated to identify the severity of each process for each day in a year. This plot is available at the end of simulation. Figure 5.7 depicts an example of this chart.
5.7.2.3 COLUMN CHART

This plot is available at the end of the simulation. Out of the three this plot gives the user the most information. There are two column charts generated; one at the end of each run and one for multiple runs (Monte Carlo). There are three columns for each process; first illustrates the number of days a process became a constraint during the simulation year, second illustrates the average sh. tons that were constrained by a process when the process became a constraint, and third illustrates the average sh. tons per day constrained by a process. Figure 5.8 depicts an example of this chart.
5.7.3 Capacity Constraint

This plot is generated at the end of the simulation. This output is helpful in observing the number of days a physical capacity (e.g. Ore Bins, Ore Passes, Convertor Shell) was either full or filling (backing up). Figure 5.9 depicts an example of this output.
5.7.4 Cumulative Bottleneck Plots

This output helps a user visualize which process is growing bottleneck and which one is not. This output adds value as it shows the increase and decrease in the flow constrained by a process. Figure 5.10 depicts an example of this output.
This output is beneficial in visual interpretation of the severity of the bottlenecks. Figure 5.10 illustrates three processes which become the bottleneck over the span of a year (365 days). Roaster occasionally becomes the bottleneck but usually have enough capacity in following days to process the constrained throughput. Casting though is not a constant accumulating bottleneck but when it does become a bottleneck it could take most of the year for it to process the constrained throughput. Convertor is the primary bottleneck as it never catches up to the constrained throughput.
5.8 Construction and Simulation of the Bottleneck Model

The plant/operation is first process-mapped using the SIPOC system. Second, the inputs are defined using rational stochastic distributions; these include the SIPOC inputs and process capabilities. Final setup step entails defining the simulated stockpiles to capture the constraint information (Figure 5.11). Next, the model is simulated and the outputs are captured. First output is the day to day bar severity chart which shows what combination of process are becoming bottlenecks and the severity of these day to day bottlenecks is captured. Second output is the severity column charts which help identify the magnitude of severity of bottlenecks over the simulation time e.g. over a year. From the simulation results, the cumulating bottleneck charts are generated which provide the overall picture of all process bottlenecks. Final output is a histogram of throughput produced. Once the base case is established through multiple runs then, the inputs can be changed according to established business cases and initiatives and the model is re-simulated. All the outputs are again captured. Improvement if any is clearly established in the throughput histogram output. If there is no improvement then the severity column charts clearly show where the improved/freed throughput was lost and in turn confirms if the initiative was actually improving the major bottleneck.

5.8.1 Structure of the Model

Processes can be defined into a model either by using built-in blocks within ExtendSIM or by creating custom blocks. Custom blocks can be created by using a Mod-L, ExtendSIM programming language. The models built for this thesis utilizes both methods. Multiple custom Blocks were built as needed to properly simulate some complex processes. A sample of customized code written in Mod-L programming language is included in Appendix J.
5.8.2 Data Capture and Filtering

This step during the creation of the model is most essential and time consuming. Due to process variation principal, any database of actual data from a production facility will include distorted and shifted data. If this data is used in the throughput model then the results obtained would also be distorted. Issue lies with the effect of variation on all processes and thus the capacity of each process being affected by extremities of each other process.

To resolve this issue, a database must be created to capture slowdown and shutdowns of each process included in the scope of a model. This in turn then allows filtering out the minimums from each process which are in the database due to process variation. MiniTab was used to accomplish data filtering for models built in this thesis. This filtering can be achieved in MS Excel too but MiniTab not only provides data manipulation functions for prompt data filtering but also allows running real-time statistical test to ensure statistical confidence in filtered data distributions.
Figure 5.11 – A flowchart of steps involved in building and simulating the model
5.9 Model Buildup and Operation

In ExtendSIM discrete rate flow model, every main process is treated as a flow valve which is represented by $\mathcal{RQ}$. Every input into the process is represented by either $\text{Rand}$ for triangular distribution, or $\text{RandMean}$ for normal distribution or combination of random variable and equation blocks $\text{RandRandMeanRandMinimum}$ for multi model distribution. The actual stockpiles such as bins, passes, etc. with physical capacity are represented by $\mathcal{CDO}$. The simulated stockpiles present with infinite capacity for identifying severity of bottlenecks are represented by $\mathcal{CO}$. There is one more type of tank blocks which are generally present at the beginning of a model; $\mathcal{CDO}$, they represent the infinite supply source of material. $\mathcal{CDO}$ is used to represent merging of two or more flow streams into one. $\mathcal{CTF}$ is used for diverging one flow stream into many. $\mathcal{CF}$ are used to throw and catch flow i.e. to direct flow without actually connecting the blocks, generally used to make model less clustered. $\mathcal{CDO}$ is used to increase or decrease the amount of flow by a factor. It can also be used to change the units of flow e.g. tons of convertor matte to anodes. Refer to Appendices for detailed description and usage of these ExtendSIM blocks.
5.9.1 Simulation of a Model

Flow was simulated as required by programming the governing mechanism in the executive block of each model. Every integrated mining process is represented by the valve block as it allows programming the constraints using the random input blocks. Each physical capacity in a mining operation such as ore passes, bins, and stockpiles are represented by tank block as they allows controlling the inventories. Merge and Diver Blocks are used to represent the splitting and joining of flow streams.

Each block will show the information about the activity if simulation is run with animation. Animation can be quite useful in getting live information about the processes as the simulation is run. It is also essential when trying to understand how the model works and if the model logic is accurate. Figure 5.12 is a snapshot of a mine model with animation on.

![Simulation Model in ExtendSIM with Animation On]

---

*Figure 5.12 - Simulation Model in ExtendSIM with Animation On*
There are several pieces of information that can be gained from this figure such as a process “84 deep” ore stope is being constrained. This is observed by looking at the information displayed on top of “84 deep” block; first number is the amount of flow that is passing through the block and second is the amount of flow that could have passed through the block if the process was not constrained by the process ahead. For “84 deep” ore stope, there is 300 st of ore being mined while the capacity was there to mine an approximate total of 378 st of ore that day. Following the same stream of flow, next process is “Birchtree Truck” which has only one number displayed on top of the block. This represents that this process is one of the bottlenecks for the day. In this case Birchtree truck can haul 300 st a day from the 84 deep stop and hence, causing the 84 deep stope to be constrained.

Other information that can be obtained from Figure 5.12 is the various level of physical capacity. One example can be noted in the “Birchtree 124 Pad” block where the ore pad is completely empty, the other is the “124 Crushed Stockpile” block which is being constrained by the Truck in front as the truck seems to be down (zero flow). This information can still be captured by exporting the data to excel at the end of the run but Animation allows us to capture same information while the model is simulating.

5.10 Use of the model & Scenario Analysis

The full extent of the model can be observed through a use of a fictional case study of a mine operation. Assume a mine which has the following unit operations as process-mapped in Figure 5.12 using ExtendSIM.

In this case study, a mine has two sections; open pit and underground. The main processes involved in open pit operation are: fragmenting, scooping, and hauling. The main processes involved in underground operation are: drilling & blasting, mucking, and
hauling. Both ore supplies are then fed to a crusher via ore pass and a feeder bin. Crushed material is stored in a surge bin which is then skipped to mill.

Data entered and simulated results for this fictional case study can be found in Appendix K.
Figure 5.13 – Fictional Mine Model - an illustration of a mine operation designed in ExtendSIM.
The model was simulated and outputs were captured. Figure 5.13 and 5.14 show the severity of constraints and tons of material produced respectively captured as outputs for the base case scenario. From severity output, it can be observed that hauling is a major constraint in open pit flow stream and mucking is a major constraint is underground flow stream as these processes constrain the major flow per day on average.
From production output it was observed that that the ore feed skipped to mill was on average 7000 sh. tons per day.

There were three business cases considered. The **First initiative** was to work on debottlenecking the hauling process for open pit ore stream. According to the business case, 5 additional trucks will increase the hauling process capacity by 10% and thus should result in extra 1000 sh. tons ore skipped to mill. The input values of open pit hauling process were increased by 10% and then the model was re simulated. Figure 5.15 and 5.16 show tons of material produced and the severity of constraints respectively captured as outputs for this initiative.

From production output it was observed that that the ore feed skipped to mill was increased to 7274 sh. tons per day. That is an increase of 7274 – 7000 = 274 sh. tons per day. However, it was also observed from severity chart output that there was a total of approx. 888 sh. tons = (1326 – 438) freed from this initiative which were less then the projected output of the business case due to variation in hauling process.
Figure 5.16 – Fictional Case Study: severity of constraints (Initiative 1).

However, that is a return of $274 \div 888 = 31\%$. So, what happened to the rest of 69% of the freed flow from this initiative? Answer can be found by closely observing Figures 5.13 and 5.16. From Figure 5.16, it is obvious that 69% of the freed material actually was constrained by crushing and skipping processes. Thus, the true value of this initiative can be estimated using only 31% return.

The Second initiative was to work on debottlenecking the mucking process for underground ore stream. This business case states that if were to increase the number of LHDs we can get 10% improvement in the output of the mine. The input values of underground mucking process were increased by 10% and then the model was re simulated. Figure 5.17 and 5.18 show tons of material produced and the severity of constraints respectively captured as outputs for this initiative.
From production output it was observed that the ore feed skipped to mill stayed at 7000 sh. tons per day. But that would mean a 0% return. Let’s examine Figure 5.18.

**Severity of Constraints**

- Figure 5.17 – Fictional Case Study: Production histogram (Initiative 2).
- Figure 5.18– Fictional Case Study: severity of constraints (Initiative 2).
Here, there was actually $986 - 41 = 945$ sh. tons per day of material was freed but all of the freed material was constrained by hauling, crushing and skipping processes. Thus, the true value of this initiative can be estimated 0% return.

The **Third initiative** was to work on debottlenecking the crushing process by increasing the utilization of the crusher. Again, the input values of crushing process were increased by 10% and then the model was re simulated. Figure 5.19 and 5.20 show tons of material produced and the severity of constraints respectively captured as outputs for this initiative.

![Figure 5.19 – Fictional Case Study: Production histogram (Initiative 3).](image)

From production output it was observed that that the ore feed skipped to mill was increased to 7250 sh. tons per day. That is an increase of $7250 - 7000 = 250$ sh. tons per day. It was also observed from severity chart output that there was a total of approx. 264 sh. tons $= (274 - 10)$ freed from this initiative.
Thus, the true value of this initiative can be estimated at $250 \div 264 = 95\%$ return.

Hence, using this case study, first we were able to identify the major bottlenecks in the system. Second, we were able to identify the areas of interest where to improve processes. Thirdly, we were able to estimate a return on three purposed initiatives. Here, it is important to note that though the physical return (tons) of initiative 1 is little more than initiative 3. The effort and resources that might be needed to increase the process capacity in initiative 1 will probably outweigh the effort and resources required to achieve the improvement from initiative 3. And that is one of the reason we should compare the return % as well as the physical return when comparing initiatives.
CHAPTER 6: FIELD TESTING AND DISCUSSIONS

6.1 Overview

All the field testing was conducted at one of Vale Inco Limited integrated Nickel Mining Operation. There were four facilities where various business initiatives were tested using the model developed through this research. There are six distinct advantages of using this tool which were field tested and the results are detailed in six case studies.

6.2 Case Study 1 – Bottleneck Identification

This study was done at a mine operated by Vale Inco. The common belief is that mine is capable of producing much more then it currently does, however for one reason or the other it always seem to be constrained by a process within the mine. Drilling and blasting process always seem to keep up with the planned throughput and hence the constraint seems to be in front of the ore bins. The mine process was mapped using ExtendSIM as depicted in Figure 6.1.

The major processes modeled were mucking/hauling, feeder, crusher and skips to surface. Hauling process which consists of a Tram train pulling blasted ore from the bins at different levels dumps the ore in an ore pass. Ore pass opens into a feeder which dumps ore into a crusher bin which also receives ore from other ore passes from various levels. Then this ore is crushed and moved to a surge bin. Surge bins feed the skips which hoist the ore to surface to be fed to mill.
Figure 6.1 – Process Map of a Vale Inco Mine Facility
Model was completed and simulated; following results were captured. Figure 6.2 shows the distribution of ore shipped to mill i.e. throughput of the mine. Figure 6.3 illustrate number of days in a simulation year (365 days), the ore bins and passes were full. Figure 6.4 shows the severity of the bottlenecks thru main processes. Finally Figure 6.5 shows the visual representation of the bottlenecks.

![Histogram of Feed to Mill](Figure 6.2 – Histogram of Mine throughput to Mill (Base Case))
Figure 6.3 – Frequency of Ore Bins & Passes being Full

Figure 6.4 – Severity of Bottlenecks as identified by the model (Base Case)
It is evident from Figure 6.5; Tram process is the major accumulating bottleneck. Figure 6.4 highlights the severity of this bottleneck as there is approx. 800 sh. tons of ore being constrained by this process every day.
6.3 Case Study 2 – Perceived Bottleneck

This study was done at a smelter facility operated by Vale Inco. Over many years in past Vale Inco Limited has spent enormous amount of resources in improving the throughput of this smelter facility but all the initiatives have resulted in minimum to limited improvements. During conducting research on this facility, it was observed that many process experts had different opinions about perceived bottleneck in the smelter facility. However, the majority of experts believed it to be the roaster process as it seemed to be down most of the time. The process map of the smelter facility as modeled in ExtendSIM is illustrated in Figure 6.6.
Figure 6.6 – Process Map of a Vale Inco Smelter Facility
The outputs of the model simulations were captured as depicted in Figure 6.7 & Figure 6.8. Histogram in Figure 6.7 shows the anodes produced per day simulated from the Smelter model.

**Histogram of Anodes Produced**

- Figure 6.7 – Histogram of Anodes produced; throughput output of the model (Base Case)

- Figure 6.8 – Severity of bottlenecks within the smelter facility (Base Case)
From Figure 6.8, it can be observed that, 102 days out of a simulated year (365 days), roaster was a constraint. And when it became a constraint, there was approx. 366 sh. tons of material constrained by it and thus it is constraining about 100 tons of material every day.

So, to test the popular theory of roaster being the major bottleneck in the smelter facility, the capacity of the roaster process was increased by 10% in the model and then model was re-simulated. Now, if this initiative was actually implemented then the cost of this initiative will be around $2 Million. But the business case will project an improvement of 20 more anodes a day which will be \((20 \times 365) = 7300\) anodes a year. At approximately $2000 per anode that will generate a revenue of $14.6 Million. On paper, this business case looks very solid as the payback is within months. The results of this 10% increase in the roaster capacity as simulated by the model are depicted in Figure 6.9 & Figure 6.10.

**Figure 6.9 – Throughput of Smelter Facility after Roaster initiative (capacity increase by 10%)**

\[
\mu = 967
\]

**Sh. tons**
From analysis of Figure 6.8 & Figure 6.10, it can be observed that yes there was material freed from the roaster (102 – 88), but all of this material was lost in convertor and casting processes. Thus the total gain in anodes from this initiative is 0 as evident in Figure 6.9. In other words, $2 Million down the drain. Even though scope of this case study was to show the result of an initiative done without identifying the bottleneck first, the tool was used to actually identify the primary bottleneck, which is the convertor process. Methodology described in case study 1 was used to identify the bottleneck.
6.4 Case Study 3 – Variation Reduction

This study was done at a mill facility operated by Vale Inco. This case study was undertaken to better understand the effect of variation reduction on the performance of the facility. The mill facility was mapped using ExtendSIM as shown in Figure 6.11. To better illustrate the point of identifying the bottleneck first, this study was done in two parts. First the variation of the crusher process was reduced by 20% and the mean was shifted as to reflect the reduction of the minimums of process capability distribution. Crusher process was chosen as it has the widest variation when compared to other mill processes.
Figure 6.11 – Process Map of a Vale Inco Mill Facility
Figure 6.12 and Figure 6.13 show the throughput produced by the mill and the severity of process constraints for the base case before the variation reduction in crusher. Figure 6.12 shows the histogram of tons of concentrate produced from the Mill model.
Figure 6.13 – Severity of Bottlenecks in the Mill (Base Case)

Figure 6.14 and Figure 6.15 show the throughput produced by the mill and the severity of process constraints for the simulation case with the variation reduction in crusher.

Figure 6.14 – Throughput Histogram of the Mill (Crusher Initiative)
Thus, it can be observed from the Figure 6.15 that even though severity of crusher constraint lowered but the freed material was consumed by the following processes. Important thing to note here is there was no statistically significant gain in throughput by reducing the variation in crusher process distribution.

The second part of this case study was to first identify the bottleneck and then reduce the variation of the bottlenecked process to see the effect of reducing variation. Grinding Mills were identified as the major bottleneck of the mill facility from Figure 6.16
Next the variation in grinding mills was reduced by 20% in such a way that the minimum were lowered and mean was shifted for the grinding mills distribution. Model was re-run and the outputs were captured. Figure 6.17 and Figure 6.18 show the throughput produced by the mill and the severity of process constraints for the case with the variation reduction in mills.
Figure 6.17 – Throughput Histogram of the Mill (Grinding Mills Initiative)

Figure 6.18 – Severity of Bottlenecks within the mill as identified by the model (Grinding mills initiative)
The throughput was increased on average from 715 to 749 sh. tons per day i.e. approx. 35 sh. tons per day. The severity output in Figure 6.18 illustrates that variation reduction in grinding mills freed more material (1793-1131) which should have resulted in a net gain of 60 sh. tons of concentrate if all of it passed through the rest of the system but it can be observed in Figure 6.18 that some of the freed material was constrained by the flotation & Separation process which confirms the results in Figure 6.17 as flotation though is not the major bottleneck but is the 2nd biggest bottleneck in the mill facility. Thus, the total gain thru this variation reduction initiative is about 58%.

The aim of this study was to show that yes, reducing the variation helps in improving the performance of a process but the bottleneck of the system should first be identified or there is a risk of no to very low gain.
6.5 Case Study 4 – Lean Optimization

This study was done at a refinery facility operated by Vale Inco. This case study was undertaken to see if any waste can be identified using this tool. The idea behind lean optimization is to identify any waste and eliminate it. For the scope of this research, waste would be any capacity of a process which is not being used and then can be eliminated to keep the throughput the same. Here it is important to note that to utilize this model, the base case will be established as outline earlier and then the range will be calculated using power test in such a way as to test if two systems are same and not different. Again, as before the power test will only prove that the processes are the same and cannot prove that the processes are different. In other words, if two cases fail to satisfy power test, it does not mean that they are different but that there is not enough proof for them to be similar.

Process map of the refinery facility drawn in ExtendSIM is illustrated in Figure 6.19.
• Figure 6.19 – Process Map of a Vale Inco Refinery Facility
The output throughput histogram (lbs of Ni produced per day with a mean of 293515) from base case simulation are illustrated in Figure 6.20.

For this case study, tank house capacity was reduced by 5% as the common belief is that tank house process is never fully utilized. The output throughput histogram (lbs of Ni produced per day with a mean of 293485) from base case simulation are illustrated in Figure 6.21. The difference falls within the test range to classify two processes as the same with a 95% CI. This was shown to be true by utilization the T-Test.
Thus, it is concluded that the tool developed in this research can be used for lean optimization (identification of waste capacity).
6.6 Case Study 5 – Un-Localized Optimization

This case study was conducted to illustrate the fact that in an integrated mining operation, an initiative must be judged on the merit of producing an optimized outcome for the whole operation. This cases study was conducted at a Vale Inco integrated mining operation with a scope enveloping mine, mill and smelter facilities. The initiative was to increase the bottleneck capacity identified in Case Study 1 i.e. the capacity of Tram process. Thus, for this case study, the capacity of Tram was increased by 10% and the mine model was re-simulated. The outputs for mine throughput and the bottleneck severity were captured as displayed in Figure 6.22 and Figure 6.23 respectively.
Comparing throughputs from Figure 6.2 and Figure 6.22, it is concluded that this initiative for increasing the tram (mine bottleneck) will increase the throughput of mine by approx. 100 sh. tons per day which in monetary terms will equate to $32000 per day at a price of $8/lb and a grade of 2%. A very solid business case if the mine was a business by itself. But at this Vale Inco’s mining operation, the mine ore throughput is fed to the mill which in turn feeds concentrate to the smelter. So, the next step is to examine how the throughput of the mill is affected by this increase in mine throughput. The increased mine throughput distribution was modified into the mill model and then the mill throughput results were captured as shown in Figure 6.24.
Comparing this to the base case mill throughput as illustrated in Figure 6.12 under Case Study 3; it can be concluded that the increase in Tram capacity of 10% would increase the mill concentrate throughput by approx. 8 sh. tons with a monetary value of $19200 per day at a price of $8/lb and a concentrate grade of 15%. That is about a decrease of 40% in business case return. In other words, 40% of the mine throughput increased through the Tram initiative is lost in the mill bottlenecks. However, it is still a good business case if the mine & mill was the business by itself. Next step was to modify this concentrate throughput increase into the Smelter model. The smelter model was re-simulated with the modification of mill throughput increase and the smelter throughput output was captured as illustrated in Figure 6.25.
Comparing this result to the smelter base throughput result in Figure 6.7, it can be observed that the throughput remained unchanged. In other words, the increase in mill throughput caused by Tram initiative was completely constrained by the smelter bottlenecks. The lesson learned through this case study is that a business case should not only look at the local optimization but at the end of the system optimization. Hence, this model through the identification of bottlenecks and simulation could be utilized as integrated planning tool for evaluating business cases.
6.7 Case Study 6 – Further Casual Mapping/Modeling

This case study was conducted to achieve one of the objectives of this research; to identify bottleneck area and then further casual map the process and model the constraint to understand the impact of sub-processes/factors within the constrained process. This case study was completed at a Vale Inco Smelter facility as highlighted in Case Study 2. First the base case was simulated and the bottleneck output was captured as in Figure 6.26. From this output, convertor process was identified to be the major bottleneck in the facility.

![Figure 6.26 – Cumulative Bottlenecks within Smelter as identified by the model (Base Case)](image)

The convertor process distribution was further mapped into three categories; availability, utilization, and quality Figure 6.27. Availability was mapped as a product of number of
convertors to average availability of convertors. Utilization was mapped as the average utilization of the convertors and the quality was mapped as a product of number of charges per convertor to the sh. tons of material throughput per charge.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure6_27.png}
\caption{A snapshot of casual mapping of convertor process in ExtendSIM}
\end{figure}
By further causal mapping of the constraint process (convertor), various scenarios now can be evaluated within the convertor process to optimize the total throughput of the smelter facility. Average number of charge per convertor was identified as the most optimum factor and returned the most improvement out of all individual identified causes in Figure 6.27. Scenarios were evaluated using the methodology described in case study 2.
CHAPTER 7: CONCLUSIONS

7.1 Overview
In this chapter, summary of the tool, the robustness and the comparison of various business initiatives through the model are provided. Also, the importance of using the proper input data is detailed. Finally, the recommendation and conclusion are also covered.

7.2 Robustness of the Model
This model is simulated using rational stochastic discrete optimization which functions on the basis of Monte Carlo technique. Due to randomness in data through every input, it is essential to simulate a given scenario at least thirty times to lock in the base case model values as the population values. Once the base case is established then using the statistical t-test we can determine the range of values for a given Confidence interval (CI) by which an initiative can be compared and judged to be statistically different or inconclusive. It is important to note here that by definition a t-test can only prove that two given cases are different and not same. If two cases fail to satisfy t-test, it does not mean that they are similar but that there is not enough proof for them to be different.
Let’s consider a case with a base throughput value of 100 tons. Through t-test it was determined that for an initiative to be different with CI of 95% the value will have to be ±20 tons. Thus if an initiative shows an improvement of 10 tons or more then t-test will conclude that the initiative has improved the process Figure 7.1.
It is very probable for the example depicted in Figure 7.1 that scenario 1 yields a positive impact, as even the minimum value is greater than 10 units off the base case. Although, the generated value of scenario 2 (105) is greater than of the base (100), the overlap is significant and therefore statistically insignificant. In other words, the return in the improvement is unsure.

7.3 Data Analysis

It is important for this tool to function effectively that the data entered in this model for the process capacities is filtered properly and is not just actual historical data. Filtering of data for a given process is essential as the historical data is actually affected by the variation in processes before and after the given process as depicted in Figure 7.2. If data is entered into the model with built-in variation then the model will execute the flow through the each process using built-in variation. Each individual process would
behave with a spread out throughput distribution resulting in total flow being affected over and over as the flow is simulated through each process.

Filtering of data can be conducted by creating a database of all shutdown and production reductions in a given operation and then conducting a Pareto Analysis. If the database is not available and time is of the essence then same information can be obtained through informal discussion with the given process experts.

In Figure 7.2, the variation of captured historical production data of a convertor is showed. This convertor process is located in a sequence of processes inside a smelter facility. The historical production data for convertor will show the variation not only cause by convertor process but all processes before and after it i.e. sum of variation caused by mining, milling, dewatering, roasting, melting, casting, and refining processes. If this accumulated variation is entered in the model instead of actual process variation

---

**Sources of Variation**

<table>
<thead>
<tr>
<th>Mining</th>
<th>Milling</th>
</tr>
</thead>
<tbody>
<tr>
<td>Roaster</td>
<td>Furnace</td>
</tr>
<tr>
<td>Converter</td>
<td>Casting</td>
</tr>
<tr>
<td>TOTAL</td>
<td>Converter</td>
</tr>
</tbody>
</table>

*Figure 7.2 – Filtering of Data: an illustration of before & after variation affecting actual performance of a process.*
then the results of the model will be much skewed to the low side and will not show the actual picture of the bottlenecks.

Data Analysis stage in building of a throughput model takes the most time and effort. Data such as tonnage, flow, capacity and delays are observed through a live system such as PI. Then, the data is analyzed and tagged for actual minimums. Next, distributions are filter using tagged processes and then data is reanalyzed to create new throughput distributions.

7.4 Conclusions

- An optimization tool has been developed and tested
- The objective proposed to develop and test an integrated optimization tool for a complex integrated mining operation has been achieved
- Results were validated and field tested at multiple Vale Mine Operations
- Integration Optimization Tool developed in this research has been adopted and is being used by Vale Base Metals to optimize and evaluate large capital expenditures

7.5 Recommendations for Further Research

There were six unique cases for utilization of this tool to conduct proper optimizations which were tested at four integrated mining facilities; mine, mill, smelter and refinery. This research revealed the following areas in bottleneck discrete rate simulation modeling that needs to be studied further.

Through the field testing of the tool, it was observed that though the model is exceptional in identifying the bottlenecks and capturing the severity of these constraints, it does not magnify the severity of the lost capacities of the processes which are being constrained
by the bottleneck. The use of this advantage of capturing this information should be further researched.

Exporting of the data from the model causes a significant waste in an analyst’s time. Further investigation and researched is needed into advanced programming of a direct entry from a model into an external database.
APPENDIX A – EXTENDSIM VALVE BLOCK

Symbol:

Purpose:
Controls, monitors, and transfers flow. This block places an upper bound on the rate at which flow is allowed to pass through. The block's maximum rate can be controlled by the Maximum rate field on the Valve tab, the R input, or the Goal and Hysteresis options found on the Flow Control tab.

Connecters:
Value input connectors: (Listed in the order they appear on the connector.)

R: Maximum rate. This overrides the Maximum rate field in the dialog.

G: Goal. This connector is used to specify the size of a new goal (either in terms of flow volume or units of time). In some configurations it can also be used to start a new goal.

start: This connector is used to start hysteresis and under some configuration goals as well.

stop: This connector is used to stop hysteresis.

Value output connectors: (Listed in the order they appear on the connector.)
Q: The cumulative quantity of flow that has passed through this block.

S: Valve status. The status of the Valve can cycle between any of the following four states: limiting (0), starved (1), blocked (2), starved & blocked (4). If the checkbox "Valve animates and reports blocking and starving information" found in the Executive on the Discrete Rate tab has been checked, this field reports all four states. However, if the checkbox is unchecked, only the limiting (0) and non-limiting (1) states are reported.

GQ/GD: This connector reports the amount of progress that has been made towards a goal in terms of flow quantity or time depending on whether a quantity goaled or duration goal has been selected.

G#: This result reports the number of goals the block has pursued. For example, if the Valve has completed 2 goals and is currently in the middle of a third, the number reported is 3.

GS: Goal status reports: none (0), start (1), in progress (2), end (3), interrupt (4).

TL: Time limiting reports the amount of simulation time this block spent limiting the effective rate of flow.

TU: Time unlimiting reports the amount of simulation time this block spent not limiting the effective rate of flow.

TLO: Time limiting zero reports the amount of simulation time this Valve blocked flow while being shutdown.
TUO: Time unlimiting zero reports the amount of simulation time this Valve did not block flow while being shutdown.

Tabs:

Valve Tab

Maximum rate: The current maximum rate is displayed here as the maximum number of flow units per time which can flow in this section of the model.
Maximum rate at R: If the R connector is connected, the maximum rate field will have this title.

Add Shutdown: Clicking this button causes a Shutdown block to be added to your model and automatically connected to the R connector.

Initial maximum rate: The initial maximum rate the Valve will have at the start of the simulation. This feature is useful if the R connector is connected.

Poll constraint only each: When this option is turned on, the Valve will query the block connected to its R connector for a new value at fixed intervals. If the Maximum rate changes between two intervals the new maximum rate will be taken into account only at the end of the interval. This option is useful when the R input is connected to a passive block like the Random number block.

Flow Control Tab
Disable control: This popup option turns off flow control.

Goal: There are two types of goals: quantity and duration. Irrespective of which type is chosen, a Valve’s maximum rate is observed while the goal is On. While the goal is off, you have the option to choose whether the maximum rate is observed or ignored (i.e., the maximum rate is set to infinity). A duration goal switches states by monitoring the passage of time. A quantity goal switches states by monitoring the quantity of flow that has passed through. When a legitimate value is received at the start connector, the goal switches to On and remains that way until the goal has been achieved or interrupted.
Goal quantity: With this option chosen, specify a fixed goal quantity that does not change during the simulation run.

Goal quantity at G: If the G connector is connected, the Goal quantity field will have this title. The value at the G connector is used to specify the goal quantity and may change during the simulation run.

Goal duration: With this option chosen, specify a fixed goal duration that does not change during the simulation run.

Goal duration at G: If the G connector is connected, the Goal duration field will have this title. The value at the G connector is used to specify the goal duration and may change during the simulation run.

Goal impact on flow when off: There are three options that control how the Valve behaves when the goal is off. Stop flow shuts the valve down. Ignore maximum rate sets the maximum rate to infinity, thereby removing any influence the Valve can have on the movement of flow. Observe maximum rate uses the rate specified on the Valve tab to limit the flow.

Start Run with goal: Three options are available for controlling how the goal is initialized at the beginning of each run. Off specifies that the goal is off at the start of the simulation. On specifies that the goal is on at the beginning of the simulation. If the Start new goal when start connector value and the Start run with goal defined by "start" connector value options have both been selected, then the start connector value at the beginning of the run determines Valve's initial on/off goal status. On the other hand, if the Start new goal when receive new goal at G connector and the Start run with goal defined by G connector value options have been both selected, then the G connector
value at the beginning of the run determines Valve's initial on/off goal status (the goal is on if G connector value is greater than 0 and not BLANK).

Initial goal quantity or Initial goal duration appears only if the option Start Run with goal on or Start Run with goal defined by "start" connector value are selected. If this checkbox is checked, a field appears allowing you to specify an initial goal at the start of the simulation, this number has to be greater than 0.

Start new goal when: Three options are available for controlling how a new goal is initiated. Start connector value allows you to define which values at the start connector will be used to start a new goal. Note in order to start a new goal with this option, not only does the start connector have to have a legitimate start value, but a message must also be received at the start connector. Receive new goal at G connector will start a new goal when a message is received. Previous goal finishes initiates a new goal as soon as the previous one has finished.

Interrupt goal when "stop" connector value: With this optional checkbox, the current goal may be interrupted if a legitimate stop value and associated message are received at the stop connector.

If a new goal arrives before the previous one is finished: There are four options available for handling the arrival of a new goal before the current goal has completed. Ignore new goal ignores any new goal that may arrive while the block is currently pursuing an existing goal. Start over using new goal will start a new goal by interrupting the current one. Continue progress using new goal use the progress that has been made towards the current goal and apply it to the new goal that has arrived prematurely. For example, if progress of 50 flow units has been made towards the current goal and a new goal of 75
arrives, then only 25 more flow units will need to be passed through before this new goal has been achieved. Use new goal after the previous goal has finished will finish up the current goal and remember the new goal was submitted prematurely.

Hysteresis: Hysteresis is a property of systems that causes them to not react instantly to a change. The purpose of adding hysteresis in a model is to introduce a delay in the time it takes some part of the system to switch from one state to another. Hysteresis allows you to insert a lag or delay in a Valve’s response to system requirements. It is used to avoid oscillations or "system nervousness" in order to achieve better control over flow movement. This is accomplished by using model conditions to explicitly control both when a Valve’s maximum rate is observed and when it is ignored. Unlike the quantity and duration goals where the conditions for applying the Valve’s maximum rate are entered in the dialog, hysteresis must always get its control information from outside the block through the start and stop connectors. The hysteresis option always relies on the Valve’s start input to control when the Valve’s maximum rate will be observed and its stop input to control when the maximum rate will be ignored. When the maximum rate is ignored, the Valve’s dialog provides a popup menu for choosing if the flow stops or if the Valve does not constrain the flow.

Observe Maximum rate when "start" connector: This field allows you to specify the conditions under which the start connector will cause the Maximum rate to be observed.

Ignore Maximum rate when "stop" connector: This field allows you to specify the conditions under which the stop connector will cause something other than the Maximum rate to be observed.
When Ignoring: There are two actions that can be taken when hysteresis is in the ignoring state; the flow may be stopped, or the maximum rate may be ignored (i.e., the maximum rate is unbounded).

Start run with hysteresis: Three options are available for controlling if the Hysteresis is ignored or observed at the beginning of the run. Ignored the hysteresis is ignored when the simulation starts. Observed the hysteresis is observed when the simulation starts. Defined by "start" connector value at the start of the simulation, the hysteresis ignored or observed depends on the value at the start connector.

Options Tab

![Options Tab Image]
Select units: In the "Select units" frame, the first popup allows the user to choose the flow units for all the blocks residing in this block's unit group by either selecting an existing flow unit category or by creating a new one. A unit group is a collection of blocks connected together through flow connections and sharing the same flow unit. Notice that once you've made your selection, not only will the "flow units / time unit" for the Maximum rate field on the Valve tab reflect that change, but the flow units for all blocks in this unit group will have changed as well. When the square button to the right of this popup menu is pressed, all blocks included within this block's unit group are selected. The second popup allows the user to define the time units for this particular Valve block.

Use Shift: The popup menu in this frame allows you to select from a list of existing shift blocks to control the Maximum rate to be 0 when the shift is off. If the selected shift is an On/Off shift, the Valve will shut down when off shift and come back online with the original value in the Maximum rate field when the shift is On. Blocks from the Rate library are not compatible with Numeric shifts.

Add Shift: This button allows you to automatically add Shift blocks to the model.

Executive: This button will open the Executive block to the Discrete Rate tab where you can define how often Rate blocks in the model update their status. For more information on this feature, please consult the Executive's help text.

Update animation and results at each event: This checkbox allows you to update the block's icon animation at every event during the model run. In order for this checkbox to be enabled, the Blocks update flow status: each block defines how often setting must be selected on the Discrete Rate table in the Executive.
Results Tab

Effective rate: The current effective rate at which flow is moving through the block.

Total flow: The total amount of flow having passed through the block.

Valve status: The status of the Valve can cycle between any of the following four states: limiting (0), starved (1), blocked (2), starved & blocked (4). If the checkbox "Valve animates and reports blocking and starving information" found in the Executive on the
Discrete Rate tab has been checked, this field reports all four states. However, if the checkbox is unchecked, only the limiting (0) and non-limiting (1) states are reported.

Cumulative time when effective rate is > 0 and Valve is limiting: The total amount of time during which flow passed through and the Valve was limiting the effective rate.

Cumulative time when effective rate is > 0 and Valve is not limiting: The total amount of time during which flow passed through and the Valve was not limiting the effective rate.

Cumulative time when effective rate is = 0 and Valve is limiting: The total amount of time during which flow did not pass through because the Valve's Maximum rate was set to zero.

Cumulative time when effective rate is = 0 and Valve is not limiting: The total amount of time during which flow did not pass through even though the Valve's Maximum rate was not set to zero.

Goal progress: This result reports the amount of progress that has been made towards a goal in terms of amount of flow for a quantity goal and amount of time for a duration goal.

Goal #: This result reports the number of goals the block has pursued. For example, if the Valve has completed 2 goals and is currently in the middle of a third, the number reported is 3.
APPENDIX B – EXTENDSIM TANK BLOCK

Symbol:

Purpose:
Acts as source, intermediate storage, or sink. As a residence type block the Tank has the capacity to hold defined amounts of flow as time advances. Contents may be 1) received from an upstream source of flow or 2) an initial quantity in the tank. If a Tank has no outflow connection, by definition it is being used as a sink. Conversely, if a tank has no inflow connection, by definition it is being used as a source.

Connecters:
Value input connectors:
C: Capacity. This overrides the Capacity field on the Tank tab.

Value output connectors: (Listed in the order they appear on the connector.)
CO: Contents; Reports the current flow level.
I: Indicator; Reports the current indicator level as defined by the table on the Indicator tab.
S: Status. Reports whether the level of flow in the Tank is currently going up (1), stable (0), or down (-1).
Tabs:

Tank Tab

Initial contents: This field allows you to initialize the starting contents of the tank to the desired flow level. There is the option to set this field to infinity. Note: if Initial contents are set to infinity, Capacity is automatically set to infinity also.

Capacity: This field allows you to set the tank's capacity for holding flow. A Tank’s capacity can be infinite, a finite but non-zero number, or zero. If the tank's contents
exceed capacity (can be the case when capacity is changed dynamically) the tank will not receive any new product until the flow level falls below capacity.

Maximum inflow rate: This option allows you to set an upper limit on the maximum inflow rate. Its behavior is similar to placing a Valve block upstream of the inflow connector. Using this feature is optional.

Maximum outflow rate: This option allows you to set an upper limit on the maximum outflow rate. Its behavior is similar to placing a Valve block downstream of the outflow connector. Using this feature is optional.

Options Tab
Use identical flow units for the group and the block: With this option chosen, there is no difference between the flow and block units.

Define a flow unit for the group and a block unit for the block: With this option chosen, flow units are different from block units as defined by the Unit factor field. Block units are used to calculate the level of flow, capacity, indicators, etc. For example, if flow units are defined as boxes and block units as tons, the inflow and outflow will be expressed in boxes and the capacity and contents for the tank will express in tons.

Flow unit: The first popup menu in this field allows the user to choose the flow units for all the blocks residing in this block's unit group by either selecting an existing flow unit category or by creating a new one. A unit group is a collection of blocks connected
together through flow connections and sharing the same flow unit. Notice that once you've made your selection here, the flow units for all blocks in this unit group will have changed. When the square button to the right of this popup menu is pressed, all blocks included within this block's unit group are selected. The second popup menu in this field allows the user to define the time units for this particular Tank block.

Block unit: This popup menu allows you to change the block units for this Tank. Block units are an internal unit of volume specific to the Tank. If you desire a block unit that differs from the flow units that come into and out of this block, you must enter a unit factor. The unit factor represents the ratio of the block unit to the flow unit. This internal representation of volume is specific to this Tank only.

Unit factor: This is the field where the conversion factor between flow and block units is defined.

Use Shift: The popup menu in this frame allows you to select from a list of existing shift blocks to control the Maximum inflow rate and Maximum outflow rate to be 0 when the shift is off. If the selected shift is an On/Off shift, flow will not be allowed to either enter or exit the block while the shift is off. Blocks from the Rate library are not compatible with Numeric shifts.

Add Shift: This button allows you to automatically add Shift blocks to the model.

Executive: This button will open the Executive block to the Discrete Rate tab where you can define how often Rate blocks in the model update their status. For more information on this feature, please consult the Executive's help text.

Update animation and results at each event: This checkbox allows you to update the block's icon animation at every event during the model run. In order for this checkbox to
be enabled, the Blocks update flow status; each block defines how often setting must be selected on the Discrete Rate table in the Executive.

Indicators Tab

As the simulation runs, the level of flow in the Tank will vary over time. You might want an indication when the flow level is within a certain range of values. This is common when monitoring the tank to determine if its contents are approaching or have reached one or more important benchmarks. For instance, some emergency procedures might need to take place if a Tank’s level reaches the “high” range; they can be
discontinued when the contents return to a “normal” range. Indicators are a method of reporting what category or range the current level of flow falls into. With this feature, each range is assigned a name, a lower limit, and an upper limit. When the level of flow reaches a value that falls within a different range, the block reports the change on its I (indicator) value output connector and alerts any connected blocks to the change in status.

Limits are absolute numbers: The numbers you type in the "Low Limit" column (the second column) will be interpreted as absolute numbers.

Limits are percentages: The numbers you type in the "Low Limit" column (the second column) will be interpreted as percentages.

Show Example: This button will fill out the Indicators table with an example of a typical set of indicators Note, the levels defined in the table must be placed in descending order, that is, the first level in row 0 must be the highest level in the Tank while the last level in the bottom row of the table must be the lowest level in the Tank.

Table column 1 (Indicator Name): This is the column where you give each level a name or label. Again, your levels must be defined in descending order.

Table column 2 (Low limit): This is the column where you type the lower limit for a particular level. Again, your levels must be defined in descending order.

Table column 3 (High Limit): This column is filled for you automatically.

Table column 4 (Value to Output): The numbers in this column are used to report the current level out the I (indicator) connectors.
Results Tab

Contents (units): Reports the current flow level.

Contents (% capacity): Reports the percentage of capacity occupied by the current level of flow.

Indicator: If Indicators are being used, this field reports which indicator the level of flow currently occupies.

Effective Rate (Inflow): Reports the current inflow effective rate.
Effective Rate (Outflow): Reports the current outflow effective rate.

Total flow (Inflow): Reports the total amount of flow that has entered this Tank.

Total flow (Outflow): Reports the total amount of flow that has exited this Tank.
APPENDIX C – EXTENDSIM DIVERGE BLOCK

Symbol:

Purpose:
Distributes the input flow to two or more outputs. The systems modeled using discrete rate technology frequently have one flow stream that needs to be split (or diverged) into multiple streams (referred to as branches). The Diverge block has been designed specifically to model this type of routing behavior. It has seven different rule-based options that define how the inflow will be distributed across the outputs.

Connecters:
ID: This connector is only used when the Diverge block is in Select an outflow mode to specify which branch should currently be open.

Go: Any dynamic changes to the merge parameters found in the Flow connections table will be ignored until a TRUE value has been received at the GO connector.

Tabs:
Diverge Tab
Diverge mode: This popup menu allows you to select from one of 7 diverge modes.

priority of outflows: The Priority mode allows you to attach priorities to the outflow branches of the Diverge block. Given the maximum rate of flow which can go through the block, the Flow will be directed in preference to the higher priority outflow branches. Note, in Priority mode, if you enter the same Priority number for more than one outflow branch, those specific outflows will behave the same as if they were in Distributional mode with identical proportions. Also note, if the priority for a particular branch has been set to blank, the effective rate for that branch will be zero and flow will stop for that branch.
select an outflow: When the Diverge block is in Select mode, only one selected branch at a time is open. The Flow connections table allows you to assign a unique ID number to each outflow branch (the ID number cannot be blank). The ID connector on the block’s icon is then used to select which branch to open. The ID value field displays the current value at the ID connector. The Invalid value at ID popup provides options for handling values at the ID input that don't match any of the branch IDs listed in the table: 1.) Choose top connection, 2.) Choose bottom connection, 3.) Stop flow, 4.) Generate error. A blank value received at the ID connector always stops the flow until the connector receives a valid input.

proportional: With the Proportional mode, you define in the Flow connections table what the proportion of flow through each branch will be. The proportion for each branch is defined in the table relative to each of the other branches. For instance, assume you have a Diverge block with two outflow branches. A value of 2 for the top outflow branch and 4 for the bottom outflow branch would indicate that the bottom branch should send twice the amount of flow as the top branch. If a particular branch's proportion has been defined to be blank or <= 0, the effective rate for that branch is set to 0 and the flow is stopped for that branch. This mode uses a fixed flow rule where the effective rate at each branch is required to meet the proportion defined by the table. Consequently, if the flow through one or more of the branches is blocked or starved, the effective rates for all branches will be set to zero and all flow through the block is halted. Note the Sum of proportions displays the total sum of the proportions that have been entered in the table.

distributional: Similar to Proportional mode, the Distributional mode allows you to define a desired set of proportions for each branch. However, unlike the Proportional mode (but similar to Priority mode), these proportions serve as the decision rule for assigning effective rates to the branches only when discrepancies arise between the upstream flow supply and the downstream flow demand. When the upstream supply is greater than or equal to the downstream demand, the block passes as much flow through each branch as
the downstream demand will allow and the proportions are ignored. However, when downstream demand exceeds upstream supply, the proportions assigned to each branch are used as preferences to determine how the limited supply should be distributed across the outflow branches. Note the Sum of proportions displays the total sum of the proportions that have been entered in the table.

unbatch: In this mode, the effective rate for the inflow connector is required to be equal to the effective rates for each outflow connector. Consequently, every unit of inflow is unbatched into one unit of flow for each outflow branch. This means the total amount of inflow is different from the total amount of outflow. The Diverge block's behavior in this mode, then, is similar to that of the Unbatch block (Item library). Note, the Unbatch mode is different from all the other diverge modes because the amount of total inflow is never equal to the amount of total outflow.

neutral: Unlike any of the modes discussed previously, the Neutral mode does not allow you to control the effective rates for the branches. This is a passive mode where no branch has a throughput advantage; the branch that gets chosen cannot be predicted. It is used when the system does not need to control how the flow is routed. When the upstream supply is greater than or equal to the downstream demand, the block passes as much flow through each branch as downstream demand will allow. However, when downstream demand exceeds upstream supply, the distribution of flow across each branch cannot be predicted.

demand sensing: Similar to the Proportional mode, the Demand Sensing mode uses proportions to calculate the effective rates for the branches. However, unlike the Proportional mode where you directly enter or control the proportions for each branch, the proportions for the Sensing modes are derived dynamically from the model as it runs. Proportions for the outflow branches are calculated as a function of the potential downstream demand. For instance, the downstream demand placed on a particular
outflow branch becomes the proportion for that branch. In the Sensing mode, you must define the maximum possible rate of flow through each branch in the Flow connections table. This upper bound is used as a way to limit throughput so that proportions can be determined when the upstream supply and downstream demand are infinite.

Flow connections table grow button: Use the green +/- grow button from this table to define the desired number of outflow connectors.

Flow connections table column 1 (To Block): Displays the block label and global block number of the block connected to this particular outflow branch.

Flow connections table column 2 (Parameter): The different meanings of this column are described above as its interpretation is dependant upon which diverge mode has been selected.

Flow connections table column 3 (Initialize): When the Use value in initialize column to initialize parameters checkbox is checked, this column will appear for those diverge modes whose branch requirements can be changed dynamically through input value connections. The modes where this column has meaning include: priority of outflows, proportional, distributional, and demand sensing. This option to initialize is useful when branch requirements change during the run, and you want the starting conditions to always be the same.

Throw connections table grow button: Use the green +/- grow button from this table to define the desired number of throw connections. Note, this can also be done on the Throw tab.

Throw connections table column 1 (Parameter): The Throw connections table and first column appear for most of diverge modes when the Enable throw connections checkbox
has been checked. It is merely an extension of Flow connections table column 2 for the throw branches.

Throw connections table column 2 (Initialize): This column is just an extension of Flow connections table column 3 for the throw branches.

Use value in initialize column to initialize parameters: When this checkbox is checked, a third column for initializing will appear in the Flow connections table for those diverge modes whose branch requirements can be changed dynamically through input value connections. The modes where this column has meaning include: priority of outflows, proportional, distributional, and demand sensing. This option to initialize is useful when branch requirements change during the run, but you want the starting conditions to always be the same. (Note, if throwing has been enabled a second column for initializing will appear in the Throw connections table as well.)

Enable throw connections: When this checkbox is checked, the Diverge block's throwing capabilities are enabled. For the modes that require a Flow connections table (including priority of outflows, select an outflow, proportional, distributional, and demand sensing), an additional and analogous throw connections table will also appear. This new table requires the same type of diverging information needed by the Flow connections table so that flow may be distributed correctly across all the branches.

Options Tab
Parameter defined using value connectors: The parameters that get defined in the Flow connections table can also be defined dynamically through a separate set of input value connections when this checkbox is checked.

Update only when a True value is received at the GO connector: With this checkbox checked, any dynamic changes to the diverge parameters found in the Flow connections table will be ignored until a TRUE value has been received at the GO connector.

Poll new parameters only each: If this checkbox is checked, then requests for new parametric values for the Flow connections table will be made by this block at dedicated time intervals. This may make sense if the Flow connections table is linked to a database.
table or if the Parameter defined using value connectors checkbox is checked. In other words you may want to consider using this option if the decision rule for distributing the flow can change dynamically.

Select units: In the "Select units" frame, the first popup allows the user to choose the flow units for all the blocks residing in this block's unit group by either selecting an existing flow unit category or by creating a new one. A unit group is a collection of blocks connected together through flow connections and sharing the same flow unit. Notice that once you've made your selection here, the flow units for all blocks in this unit group will have changed. When the square button to the right of this popup menu is pressed, all blocks included within this block's unit group are selected. The second popup allows the user to define the time units for this particular Bias block.

Executive: This button will open the Executive block to the Discrete Rate tab where you can define how often Rate blocks in the model update their status. For more information on this feature, please consult the Executive's help text.

Update animation and results at each event: This checkbox allows you to update the block's icon animation at every event during the model run. In order for this checkbox to be enabled, the Blocks update flow status: each block defines how often setting must be selected on the Discrete Rate table in the Executive.
Enable throw connections: This checkbox enables the Diverge block's throwing capabilities.

Table column 1 (Block#): Enter the catch type block you want the associated "throw branch" to throw to (you also can select block# in the list of available blocks into the next column).

Table column 1': Using the popup for this column, select the catch type block you want the associated "throw branch" to throw to.

Table column 2 (Position): Enter the "catch branch" you want to throw to. Note, this option only makes sense if you are throwing to a Merge block.
Table column 2' : Using the popup for this column, select the "catch branch" you want to throw to. Note, this option only makes sense if you are throwing to a Merge block.

Table column 3 (Filter group): This column allows you to control which group of catch type blocks you would like to see when you click on a popup from column 1'.

Table column 4 (Type filter): This column allows you to control what type of catch type blocks (Catch, Merger or both) you would like to see when you click on a popup from column 1'.

Table column 5 (Open): This column allows you to open the Catch (or Merge) block associated with a particular outflow branch from this block.

Filter only non-connected blocks: This checkbox allows you to show only the "free" catch type blocks (those not already assigned to a throw) when you click on a popup menu from column 1'.

My Group: This popup allows you to choose which "throw/catch" group you want this block to be associated with.

Show My Group on icon: When this checkbox is checked, the "throw/catch" group this block belongs to is displayed on the icon.

Model Settings Tab
Executive: Clicking on this button will open the Executive block's dialog to the Discrete Rate tab. Once there you can choose from two bias order control options: defined by simulation order or each block chooses its own.

Bias order: If each block chooses its own bias order (as defined by the Bias order popup in the Executive), then the bias order table will be enabled. Clicking the << or >> buttons will let you change the bias value on the selected row in the table.

Bias order table column 1 (Bias): Lists the bias value for the block associated with this row of the table.
Bias order table column 2 (Block Label[#]): Shows the block label plus the global block number for the block associated with this Bias value.

Bias order table column 3 (Block Name): Shows the Block Name associated with this Bias value.

Bias order table column 4 (Mode): Shows the selected mode of the block associated with this row of the table.

Bias order table column 5 (Sim Order): Shows the simulation order of the block associated with this row of the table.

Bias order table column 6 (Open): Allows you to open the dialogs of any blocks shown in this list.

Executive: Clicking on this button will open the Executive block's dialog to the Model Setting tab. From here you can choose from one of three options: show bias order on icon, don't show bias order on block icons, each block decides whether to show bias order on icon. If show bias order on icon is chosen, the bias order will be displayed under bracket <#> on the icon of the block.

Show bias order on icon: If the option "each block decides whether to show bias order on icon" has been chosen in the Discrete Rate tab of the Executive, then the Show bias order on icon checkbox will appear. If it is checked, the bias order will be displayed under bracket <#> on the icon of the block.
Effective rate: This field displays the current effective inflow rate.

Total flow: This field displays the total amount of flow that has entered the block.

Flow connections table column 1 (Outflow Rate): The effective outflow rate for each out flowing connection branch.

Flow connections table column 2 (Total Flow Out): The total amount of flow that has exited each out flowing connection branch.
Throw connections table column 1 (Outflow Rate): The effective outflow rate for each out flowing throw branch.

Throw connections table column 2 (Total Flow Out): The total amount of flow that has exited each out flowing throw branch.
APPENDIX D – EXTENDSIM MERGE BLOCK

Symbol: 

Purpose:
Merges flows from multiple inputs into one output. The systems modeled using discrete rate technology frequently have multiple flow streams (referred to as branches) that need to be merged into one stream. The Merge block has been designed specifically to model this type of routing behavior. It has seven different rule-based options that define how the inflow will be merged from all inputs.

Connecters:
ID: This connector is only used when the Merge block is in Select an inflow mode to specify which branch should currently be open.

Go: Any dynamic changes to the merge parameters found in the Flow connections table will be ignored until a TRUE value has been received at the GO connector.

Tabs:
Merge Tab
Merge mode: This popup menu allows you to select from one of 7 diverge modes.

Priority of inflows: The Priority mode allows you to attach priorities to the inflow branches of the Merge block. Given the maximum rate of flow which can go through the block, the Flow will be taken in preference from the higher priority inflow branches. Note, in Priority mode, if you enter the same Priority number for more than one inflow branch, those specific inflows will behave the same as if they were in Distributional mode with identical proportions. Also note, if the priority for a particular branch has been set to blank, the effective rate for that branch will be zero and flow will stop for that branch.
select an inflow:  When the Merge block is in Select mode, only one selected branch at a
time is open. The Flow connections table allows you to assign a unique ID number to
each inflow branch  (the ID number cannot be blank). The ID connector on the block’s
icon is then used to select which branch to open.  The ID value field displays the current
value at the ID connector.  The Invalid value at ID popup provides options for handling
values at the ID input that don't match any of the branch IDs listed in the table:  1.)
Choose top connection, 2.) Choose bottom connection, 3.) Stop flow, 4.) Generate error.
A blank value received at the ID connector always stops the flow until the connector
receives a valid input.

proportional:  With the Proportional mode, you define in the Flow connections table what
the proportion of flow through each branch will be. The proportion for each branch is
defined in the table relative to each of the other branches. For instance, assume you have
a Merge block with two inflow branches.  A value of 2 for the top inflow branch and 4
for the bottom inflow branch would indicate that the bottom branch should receive twice
the amount of flow as the top branch. If a particular branch's proportion has been defined
to be blank or <= 0, the effective rate for that branch is set to 0 and the flow is stopped
for that branch.  This mode uses a fixed flow rule where the effective rate at each branch
is required to meet the proportion defined by the table. Consequently, if the flow through
one or more of the branches is blocked or starved, the effective rates for all branches will
be set to zero and all flow through the block is halted.  Note the Sum of proportions
displays the total sum of the proportions that have been entered in the table.

distributional:  Similar to Proportional mode, the Distributional mode allows you to
define a desired set of proportions for each branch. However, unlike the Proportional
mode (but similar to Priority mode), these proportions serve as the decision rule for
assigning effective rates to the branches only when discrepancies arise between the
upstream flow supply and the downstream flow demand.  When the downstream demand
is greater than or equal to the upstream supply, the block passes as much flow through each branch as the downstream demand will allow and the proportions are ignored. However, when upstream supply exceeds downstream demand, the proportions assigned to each branch are used to determine how the excess supply should be distributed across the outflow branches. Note the Sum of proportions displays the total sum of the proportions that have been entered in the table.

batch: In this mode, the effective rate for each inflow connector is required to be equal to the effective rate for the outflow connector. Consequently, one unit of flow from each inflow branch is batched into one unit of flow for the outflow branch. This means the total amount of inflow is different from the total amount of outflow. The Merge block's behavior in this mode, then, is similar to that of the Batch block (Item library). Note, the Batch mode is different from all the other merge modes because the amount of total inflow is never equal to the amount of total outflow.

neutral: Unlike any of the modes discussed previously, the Neutral mode does not allow you to control the effective rates for the branches. This is a passive mode where no branch has a throughput advantage; the branch that gets chosen cannot be predicted. It is used when the system does not need to control how the flow is routed. When the downstream demand is greater than or equal to the upstream supply, the block passes as much flow through each branch as upstream supply can provide. However, when upstream supply exceeds downstream demand, the distribution of flow across each branch cannot be predicted.

supply sensing: Similar to the Proportional mode, the Supply Sensing mode uses proportions to calculate the effective rates for the branches. However, unlike the Proportional mode where you directly enter or control the proportions for each branch, the proportions for the Sensing modes are derived dynamically from the model as it runs. Proportions for the inflow branches are calculated as a function of the potential upstream
supply. For instance, the upstream supply placed on a particular inflow branch becomes the proportion for that branch. In the Sensing mode, you must define the maximum possible rate of flow through each branch in the Flow connections table. This upper bound is used as a way to limit throughput so that proportions can be determined when the upstream supply and downstream demand are infinite.

Flow connections table grow button: Use the green +/- grow button from this table to define the desired number of inflow connectors.

Flow connections table column 1 (From Block): Displays the block label and global block number of the block connected to this particular inflow branch.

Flow connections table column 2 (Parameter): The different meanings of this column are described above as its interpretation is dependant upon which merge mode has been selected.

Flow connections table column 3 (Initialize): When the Use value in initialize column to initialize parameters checkbox is checked, this column will appear for those merge modes whose branch requirements can be changed dynamically through input value connections. The modes where this column has meaning include: priority of inflows, proportional, distributional, and supply sensing. This option to initialize is useful when branch requirements change during the run, and you want the starting conditions to always be the same.

Catch connections table grow button: Use the green +/- grow button from this table to define the desired number of catch connections. Note, this can also be done on the Catch tab.
Catch connections table column 1 (Parameter): The Catch connections table and first column appear for most of the merge modes when the Enable throw connections checkbox has been checked. It is merely an extension of Flow connections table column 2 for the catch branches.

Catch connections table column 2 (Initialize): This column is just an extension of Flow connections table column 3 for the catch branches.

Use value in initialize column to initialize parameters: When this checkbox is checked, a third column for initializing will appear in the Flow connections table for those merge modes whose branch requirements can be changed dynamically through input value connections. The modes where this column has meaning include: priority of outflows, proportional, distributional, and supply sensing. This option to initialize is useful when branch requirements change during the run, but you want the starting conditions to always be the same. (Note, if throwing has been enabled a second column for initializing will appear in the Thow connections table as well.)

Enable catch connections: When this checkbox is checked, the Merge block's catching capabilities are enabled. For the modes that require a Flow connections table (including priority of outflows, select an outflow, proportional, distributional, and supply sensing), an additional and analogous Catch connections table will also appear. This new table requires the same type of merging information needed by the Flow connections table so that flow may be combined correctly across all the branches.
Parameter defined using value connectors: The parameters that get defined in the Flow connections table can also be defined dynamically through a separate set of input value connections when this checkbox is checked.

Update only when a True value is received at the GO connector: With this checkbox checked, any dynamic changes to the merge parameters found in the Flow connections table will be ignored until a TRUE value has been received at the GO connector.
Poll new parameters only each:  If this checkbox is checked, then requests for new parametric values for the Flow connections table will be made by this block at dedicated time intervals. This may make sense if the Flow connections table is linked to a database table or if the Parameter defined using value connectors checkbox is checked. In other words you may want to consider using this option if the decision rule for distributing the flow can change dynamically.

Select units:  In the "Select units" frame, the first popup allows the user to choose the flow units for all the blocks residing in this block's unit group by either selecting an existing flow unit category or by creating a new one. A unit group is a collection of blocks connected together through flow connections and sharing the same flow unit. Notice that once you've made your selection here, the flow units for all blocks in this unit group will have changed. When the square button to the right of this popup menu is pressed, all blocks included within this block's unit group are selected. The second popup allows the user to define the time units for this particular Bias block.

Executive:  This button will open the Executive block to the Discrete Rate tab where you can define how often Rate blocks in the model update their status. For more information on this feature, please consult the Executive's help text.

Update animation and results at each event:  This checkbox allows you to update the block's icon animation at every event during the model run. In order for this checkbox to be enabled, the Blocks update flow status: each block defines how often setting must be selected on the Discrete Rate tab in the Executive.
Enable catch connections: This checkbox enables the Merge block's catching capabilities.

Table column 1 (Block#): Enter the Throw type block you want the associated "catch branch" to catch from (you also can select block# in the list of available blocks into the next column).

Table column 1': Using the popup for this column, select the throw type block you want the associated "catch branch" to catch from.
Table column 2 (Position): Enter the "throw branch" you want to catch from. Note, this option only makes sense if you are catching from a Diverge block.

Table column 2': Using the popup for this column, select the "throw branch" you want to catch from. Note, this option only makes sense if you are catching from a Diverge block.

Table column 3 (Filter group): This column allows you to control which group of throw type blocks you would like to see when you click on a popup from column 1'.

Table column 4 (Type filter): This column allows you to control what type of throw type blocks (Catch, Merger or both) you would like to see when you click on a popup from column 1'.

Table column 5 (Open): This column allows you to open the Throw (or Diverge) block associated with a particular inflow branch from this block.

Filter only non-connected blocks: This checkbox allows you to show only the "free" throw type blocks (those not already assigned to a catch) when you click on a popup menu from column 1'.

My Group: This popup allows you to choose which "throw/catch" group you want this block to be associated with.

Show My Group on icon: When this checkbox is checked, the "throw/catch" group this block belongs to is displayed on the icon.

Model Settings Tab
Executive: Clicking on this button will open the Executive block's dialog to the Discrete Rate tab. Once there you can choose from two bias order control options: defined by simulation order or each block chooses its own.

Bias order: If each block chooses its own bias order (as defined by the the Bias order popup in the Executive), then the bias order table will be enabled. Clicking the << or >> buttons will let you change the bias value on the selected row in the table.
Bias order table column 1 (Bias): Lists the bias value for the block associated with this row of the table.

Bias order table column 2 (Block Label[#]): Shows the block label plus the global block number for the block associated with this Bias value.

Bias order table column 3 (Block Name): Shows the Block Name associated with this Bias value.

Bias order table column 4 (Mode): Shows the selected mode of the block associated with this row of the table.

Bias order table column 5 (Sim Order): Shows the simulation order of the block associated with this row of the table.

Bias order table column 6 (Open): Allows you to open the dialogs of any blocks shown in this list.

Executive: Clicking on this button will open the Executive block's dialog to the Model Setting tab. From here you can choose from one of three options: show bias order on icon, don't show bias order on block icons, each block decides whether to show bias order on icon. If show bias order on icon is chosen, the bias order will be displayed under bracket <#> on the icon of the block.

Show bias order on icon: If the option "each block decides whether to show bias order on icon" has been chosen in the Discrete Rate tab of the Executive, then the Show bias order on icon checkbox will appear. If it is checked, the bias order will be displayed under bracket <#> on the icon of the block.
Executive: Clicking on this button will open the Executive block's dialog to the Discrete Rate tab. Once there you can choose from three options when a Merge block is in Proportional mode and loops are empty: branches need simultaneous inflows to push flow, blocks push flow even in empty loops, and each block defines how it will push flow.

Results Tab

![Merge Rate Dialog Box](image)

**Outflow results**

- Effective rate: __________ units/time
- Total flow: __________ units

**Inflow results**

Flow connections

<table>
<thead>
<tr>
<th>Inflow Rate</th>
<th>Total Flow In</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
</tr>
</tbody>
</table>

Effective rate: This field displays the current effective outflow rate.
Total flow: This field displays the total amount of flow that has exited the block.

Flow connections table column 1 (Outflow Rate): The effective inflow rate for each inflowing connection branch.

Flow connections table column 2 (Total Flow Out): The total amount of flow that has entered each inflowing connection branch.

Catch connections table column 1 (Outflow Rate): The effective inflow rate for each inflowing catch branch.

Catch connections table column 2 (Total Flow Out): The total amount of flow that has entered each inflowing catch branch.
APPENDIX E – EXTENDSIM UNITS BLOCK

Symbol: 

Purpose:
Changes the flow unit of measurement. Flow units describe what is flowing from one Rate library block to another. Blocks that are connected together through flow connections and share the same flow unit are called a unit group. The Change Units block uses the conversion factor to create a new unit group. This causes the blocks downstream of the Change Units block to be in a unit group different from its upstream blocks.

Connecters:
factor: This input overrides The factor is field on the Units tab.

Tabs:

Units Tab
Change units from: These are the flow units coming into the block. The popup allows you to choose the flow units for all the blocks residing in this block's upstream unit group by either selecting an existing flow unit category or by creating a new one. A unit group is a collection of blocks connected together through flow connections and sharing the same flow unit. Notice that once you've made your selection here, the flow units for all the upstream blocks in this unit group will have changed. When the square button to the right of this popup menu is pressed, all blocks included within this block's upstream unit group are selected.

to: These are the flow units exiting from the block. The popup allows you to choose the flow units for all the blocks residing in this block's downstream unit group by either selecting an existing flow unit category or by creating a new one. A unit group is a collection of blocks connected together through flow connections and sharing the same flow unit. Notice that once you've made your selection here, the flow units for all the downstream blocks in this unit group will have changed. When the square button to the
right of this popup menu is pressed, all blocks included within this block's downstream unit group are selected.

The factor is: This field is used to define a constant conversion factor. The popup to the left of this field allows you to define the conversion factor in terms of inflow units per outflow units or vice versa. Negative, zero or blank conversion factors are invalid.

Show flow unit change on icon: With this checkbox checked, the "from / to" flow units will be displayed on the Change Units block icon.

Results Tab
Inflow rate: This field reports the effective rate of flow entering the block.

Outflow rate: This field reports the effective rate of flow leaving the block.

In: This field reports the total amount of flow having entered the block.

Out: This field reports the total amount of flow having left the block.

History Tab

Record conversion factor history:

Table column 1 (Date): This column reports what time the associated factor was first introduced.
Table column 2 (Duration):  This column reports the amount of time the associated conversion factor was used.

Table column 3 (Factor):  This column reports the conversion factor.

Table column 4 (Quantity In):  This column reports how much flow entered the block while the associated conversion factor was in use.

Table column 5 (Quantity Out):  This column reports how much flow exited the block while the associated conversion factor was in use.
APPENDIX F – EXTENDSIM RANDOM NUMBER BLOCK

Symbol:

Purpose:
Generates random integers or real numbers based on the selected distribution. You can use the dialog or the three inputs, 1, 2, and 3 to specify arguments for the distributions. You can select the type of distribution: Uniform (integer or real), Beta, Binomial, Cauchy, Chi Squared, Constant, Erlang, Exponential, Extreme Value type 1a, Extreme Value type 1b, Gamma, Geometric, HyperExponential, HyperGeometric, Inverse Gaussian, Inverse Weibull, Johnson SB, Johnson SU, Laplace, Logarithmic, Logistic, Log-Logistic, LogNormal, Negative Binomial, Normal, Pareto, Pearson type V, Pearson type VI, Poisson, Power Function, Rayleigh, Triangular, Weibull, or Empirical Table.

The Empirical distribution uses a table to generate a discrete, stepped, or interpolated empirical distribution. To change the number of rows in the table, click on the resize button on the table (the green +- icon in the lower right hand corner of the table). This button will allow you to resize the table to whatever size you wish. You will also be asked to resize the table when your first select the Empirical distribution.

Where it occurs, the location parameter shifts the entire distribution by the location value. This has the same effect as adding a constant to the output of the block.

Connecters:
1: Value of argument 1. If connected, this overrides the (1) dialog parameter.

2: Value of argument 2. If connected, this overrides the (2) dialog parameter.
3: Value of argument 3. If connected, this overrides the (3) dialog parameter.

The output is the random value.

Tabs:

Distributions Tab
(1): The first argument for the selected distribution. This variable changes depending on the type of distribution.

(2): The second argument for the selected distribution. This variable changes depending on the type of distribution; it is sometimes unused.

(3): The third argument for the selected distribution. This variable changes depending on the type of distribution; it is sometimes unused.

Beta: A continuous distribution with finite upper and lower bounds. The two shape parameters can be used to generate a wide variety of density patterns within the two bounds. The lower bound is the location parameter; the upper bound is the maximum added to the location parameter. This distribution is often used as a rough estimate in the absence of data, a distribution of a random proportion, or the time to complete a task. The uniform distribution is a special case of the beta distribution (both shape parameters are equal to one).

Binomial: Outputs a value which is the number of successes (argument (1) - Probability) in a fixed number of independent trials (argument (2) - N). Probability is a real number and N is an integer. For example, this distribution is used to show the number of defective items in a batch of size N, the probability of error in the transmission of a message consisting of a specific number of bits, or the probability that a specified number of people will recover from a rare blood disease.

Cauchy: The Cauchy distribution is an unbounded continuous distribution that has a sharp central peak but significantly broad tails. The tails are much heavier than the tails of the Normal distribution. Lambda is a scaling parameter.
Chi Squared: The Chi Squared distribution is a continuous distribution bounded on the left side. Note that the Chi Squared distribution is a subset of the Gamma distribution with beta = 2 and alpha = nu/2. Nu is a shape parameter.

Constant: Outputs a constant value.

Discrete: For the Empirical distribution only. The data table will be used as discrete probabilities of the values given in the "Value" column. This means that the values listed in the value column are the exact numbers that the block will output.

Empirical table: Enter values in the first column and enter the probability of that value occurring in 100 cases in the second column. The value column contains the various values that will be output; Probability describes the chance that value will occur. The probabilities need only have the proper values relative to each other, since ExtendSim scales them automatically. You may type the values in directly or import the values to the data table through the Clipboard using the commands in the Edit menu.

Erlang: Outputs a value varying around the given (1) mean, with a wide range of outcomes depending on the value of the second argument, "k". This distribution is used in telephone traffic and queuing theory when an activity or service time is considered to occur in phases with each phase being exponentially distributed. It is common to use the Erlang distribution as a service time when you want to simplify a model by combining several similar steps into one representative step. The value of "k" should be an integer. Like the Weibull, the curve approximates other distributions depending on the value of its Mean and especially the value of "k". A "k" of 1 resembles the exponential distribution while larger values tend to a normal distribution.

Exponential: A distribution shaped like a decaying exponential. This choice outputs a value varying around the (1) Mean, where the Mean is a non-negative real number.
However, the distribution is positively skewed (longer tail on the right), so it is more likely that the values will be between 0 and the Mean than between the Mean and two times the Mean. This distribution is the one most often used in science, business processes, and queuing theory. Use it to represent the length of a telephone conversation, the expected lives of electronic components, the time between failures for equipment, or any other situation where the events are completely independent of each other. It is generally not appropriate for modeling delay or processing times.

Extreme Value Type 1A: The Extreme Value IA distribution is an unbounded continuous distribution. It is also called the Gumbel distribution. The Extreme Value IA distribution describes the limiting distribution of the greatest values of many types of samples. Beta is a scale parameter.

Extreme Value Type 1B: The Extreme Value IB distribution is an unbounded continuous distribution. The Extreme Value IB distribution describes the limiting distribution of the least values of many types of samples. Beta is a scale parameter.

Gamma: A continuous distribution bounded by zero at the left and unbounded on the right. The exponential and Erlang are special cases of the gamma distribution. Because of its flexibility, the gamma distribution can be used for a wide variety of purposes including: interarrival times, time to complete a task, or lifetimes.

Geometric: A discrete distribution bounded by zero on the left and unbounded on the right. It can be defined as the number of failures before the first success in a series of trials. In shape, it is similar to the exponential distribution. Traditional uses include inventory demand and the number of items inspected before the first defective item is found.
HyperExponential: A distribution used in telephone traffic and queuing theory given its Mean. It perturbs the Exponential distribution in an opposite way to the Erlang. The second argument, "s", ranges from 0 to 0.5 with 0.5 giving an Exponential distribution.

Hypergeometric: The Hypergeometric distribution is a discrete distribution bounded by [0,s]. It describes the number of defects, x, in a sample of size s from a population of size N which has m total defects.

Interpolated: For the Empirical distribution only. The probability distribution will be interpolated between the data points. The value that is output will be the values in the table and the values between those values. The probability of any value being output is also adjusted.

Inverse Gaussian: The Inverse Gaussian distribution is a continuous distribution with a bound on the lower side. It is uniquely zero at the minimum x and always positively skewed. The Inverse Gaussian distribution is also known as the Wald distribution. Alpha is a shape parameter. Beta is a mixture of shape and scale.

Inverse Weibull: The Inverse Weibull distribution is a continuous distribution with a bound on the lower side. It is uniquely zero at the minimum x, and always positively skewed. In general, the Inverse Weibull distribution fits bounded, but very peaked, data with a long positive tail. Alpha is a shape parameter. Beta is a mixture of shape and scale.

Johnson SB: the Johnson SB distribution is a continuous distribution that has both upper and lower finite bounds, similar to the Beta distribution. The Johnson SB distribution, together with the Lognormal and the Johnson SU distributions are transformations of the Normal distribution and can be used to describe most naturally occurring unimodal sets.
of data. Lambda is the range of X above the minimum. Gamma is a skewness parameter. Delta is a shape parameter.

Johnson SU: The Johnson SU distribution is an unbounded continuous distribution. The Johnson SU distribution, together with the Lognormal and the Johnson SB distributions are transformations of the Normal distribution and can be used to describe most naturally occurring unimodal sets of data. Lambda is the range of X above the minimum. Gamma is a skewness parameter. Delta is a shape parameter.

Laplace: The Laplace distribution, sometimes called the double exponential distribution, is an unbounded continuous distribution that has a very sharp central peak, located at theta. The distribution scales with phi.

Location: Some distributions have a fixed origin of 0. For these distributions, the location parameter will appear. This parameter can be used as an offset to the distribution so that it can have any fixed origin value. A value is generated from the distribution and then the location parameter is added to the result. This is identical to adding a value to the output of the Input Random Number block.

Logarithmic: The Logarithmic distribution is a discrete distribution bounded by \([1, \ldots]\). Typically, if the data is bounded by \([0, \ldots]\), then translating the data before fitting is required. Theta is related to the sample size and the mean.

Logistic: The Logistic distribution is an unbounded continuous distribution which is symmetrical about its mean. The shape of the Logistic distribution is very much like the Normal distribution, except that the Logistic distribution has broader tails. Alpha is a shift parameter. Beta is a scale parameter.
Log-Logistic: For Shape = 1, it resembles the Exponential distribution. For Shape < 1, it tends to infinity at Location, and decreases with increasing X. For Shape > 1, it is zero at Location, and then peaks and decreases.

Lognormal: Natural log of the variable that follows the Gaussian or bell curve with the given (1) Mean and (2) Std Dev (standard deviation). This distribution outputs a value > 0, skewed so that most of the values occur near the minimum value (positive skew). Lognormal is often appropriate for multiplying processes, while the Normal is best for additive processes. This distribution is widely used in business for security or property valuation, such as the rate of return on stock or real estate returns.

Negative Binomial: Number of failures before Sth success. P specifies the probability of success.

Normal: Gaussian or bell curve with the given (1) Mean and (2) Std Dev (standard deviation). This choice outputs a value approximately equal to the Mean every time unit, where the value is as likely to be from 0 to the Mean as to be from the Mean to two times the mean. The Mean is specified as a real number and the standard deviation is specified as a non-negative real number. The larger the standard deviation is, the wider the spread of values around the mean. For example, given a mean of 6 and the expectation that 68% of the numbers will occur at ±4 (that is, that 68% of the values fall between 2 and 10), you would enter a STD Dev of 4. This is calculated as 4/1, where 1 represents 1 standard deviation width of values (68%). However, if you expect that 96% of the numbers, or 2 standard deviations, will fall within that same range, enter a STD Dev of 2. This is calculated as 4/2.

Pareto: The Pareto distribution is a continuous distribution bounded on the lower side. It has a finite value at the minimum x and decreases monotonically for increasing x. A Pareto random variable is the exponential of an Exponential random variable, and
possesses many of the same characteristics. Minimum is the minimum value for X. Alpha is a scale parameter.

Pearson type V: A distribution typically used to represent the time required to complete some task. A continuous distribution bounded by zero on the left and unbounded on the right. The density takes on shapes similar to lognormal, but can have a larger "spike" close to \( x = 0 \).

Pearson type VI: A distribution typically used to represent the time required to complete some task. A continuous distribution bounded by zero on the left and unbounded on the right.

Plot Sample: Plots a sample from the selected distribution in a histogram. Note: this plot is for the distributions on the left only; the Empirical distribution has its own "Plot Table" button.

Plot Table: Plots the outline of the Empirical distribution specified in the data table.

Poisson: Describes the number of events that occur in a given interval based on the given rate or (1) Mean. The variance equals the mean, so the larger the mean the wider the spread from the mean. This distribution is used to represent the number of telephone calls per minute, the number of errors per page, or the number of arrivals to a system.

Power function: A continuous distribution with both upper and lower finite bounds. It is a special case of the Beta distribution with \( q = 1 \). The Uniform distribution is a special case of the Power Function distribution with \( p = 1 \). Minimum is the minimum value, Maximum is the maximum value. Alpha is a shape parameter.
Rayleigh: The Rayleigh distribution is a continuous distribution bounded on the lower side. This distribution is frequently used to represent lifetimes because its hazard rate increases linearly with time, e.g. the lifetime of vacuum tubes. This distribution also finds application in noise problems in communications. Sigma is a scale parameter.

Stepped: For the Empirical distribution only. The data table will be used as probabilities of ranges of data. The lowest value in the Value column defines the low end of the bin while the next range value defines the upper end. This choice requires that the last set of points define the upper end of the distribution. For this reason the probability level of the last and next to last points must be equal. If this is not in the data table, an additional point will be added onto the data.

Triangular: Outputs a value N, where N is a real (decimal) number greater than or equal to the real number selected for argument 1 (the minimum) and less than or equal to the real number selected for argument 2 (the maximum) with the added provision that N tends towards its most likely, or modal value. You would use this distribution to specify a distribution in which you knew the lowest possible value, the highest possible value and a central tendency. The actual performance of this distribution will be similar to the normal distribution with the exception that it can be skewed (if the most likely value is specified to the left or right of the mean) and that there is no possibility of outlying values. Note that the most likely value is the mode and not the mean (or average). To determine the mean of the triangular distribution, sum the minimum, maximum, and most likely values together and then divide by 3.

Uniform integer: Outputs an integer (whole) number greater than or equal to the integer selected for argument 1 and less than or equal to the integer selected for argument 2. In this distribution, all integer values between the minimum and maximum are equally likely to occur. For instance, you would use this distribution to indicate that the expected
pricing for a new product is from 200 to 400 or to show values that represent "best case/worst case" scenarios.

Uniform Real: This is the default selection. Outputs a real (decimal) number greater than or equal to the value selected for argument 1 and less than or equal to the value selected for argument 2. In this distribution, all the values between the minimum and maximum are equally likely to occur. For instance, you would use this distribution to indicate "best case/worst case" scenarios, or that the least a piece of equipment would cost would be $347.50 and the most it would cost would be $452.95.

Weibull: This distribution can assume the properties of other distributions (such as the Exponential or Rayleigh) depending on its (1) Scale and (2) Shape arguments, both of which are non-negative real numbers. It is commonly used to describe failure rates, lifetime expectancies, or the time to complete a task. The curve of the distribution changes considerably depending on the value of Scale and especially the value of Shape. The Shape variable should be greater than 0. For example, given a Scale of 1 and a Shape of 1, the Weibull is essentially an exponential distribution. However, given a Scale of 1 and a Shape of 2 the curve resembles a skewed normal distribution. Note that there are other terms used for the scale and shape of the Weibull distribution; check the documentation for the source of the distribution carefully. In Stat::Fit, alpha is the shape and beta is the scale.

Options Tab
Generate one random number at simulation start: If checked, the Random Number block will generate one value at the start of the simulation run. This value will stay the same throughout the entire simulation.

Use block seed: Sets the offset value for the random number seed used by this block. If this is not checked, the offset is the block number (a unique identifier for each block) + 1. In most cases, each block which generates random numbers should have its own, unique seed value. The Statistics library (included with the Manufacturing or BPR libraries) contains the Random Seed Control block which can check for duplicate seed values.

Distribution Fitting Tab
Distribution fitting software: Distribution fitting software will help you to select the appropriate ExtendSim statistical distribution based on historical data. Use this pop-up menu to select your distribution fitting software. Currently, three Windows distribution fitting software packages are supported. These include:
- BestFit(R) from Palisade Corporation - (800) 432 7475 - http://www.palisade.com
- Stat::Fit(TM) from Geer Mountain Software - (860) 927 4328 - http://www.geerms.com

In addition, the statistical analysis software package, JMP(R) from the SAS Institute - (919) 677 8000 - http://www.jmp.com - is supported for both Macintosh and Windows.

Import distribution from clipboard (Stat::Fit): Use this to paste a distribution copied (exported) onto the clipboard from Stat::Fit. A preview of pasted distribution appears below this button.
Open: Opens the currently selected distribution fitting software.

Path: Selects the path for the distribution fitting software. In most cases you can use the default path entered by ExtendSim. You will need to change this, however, if you have installed your distribution fitting software in a non-default location or if you have selected "Other" as your distribution fitting software.
APPENDIX G – EXTENDSIM EQUATION BLOCK

Symbol:

Purpose:
Outputs the results of an equation entered in the dialog. You can use ExtendSim's built-in operators, functions, and some or all of the input values as part of the equation. The equation can have any number of inputs and any number of outputs. As you expand the Input Variables table, and the Output Variables table, the number of input connectors and output connectors will be expanded automatically.

ExtendSim's operators are:  +, -, *, /, ^ (exponentiation), MOD or % (modulus), AND or &&, OR or ||, NOT or !, == (equals), != or <> (not equal), <, <=, >, >=

To define multiple inputs or outputs, click on the resize button on the appropriate table. This button (the green + icon in the lower right hand corner of the table) will allow you to resize the table to whatever size you wish.

ModL functions from ExtendSim's development environment can be used in equations. Using these functions can greatly expand the range of things that can be done in equations. See the ExtendSim Developers Reference, or the ExtendSim online help (available under the help menu,) for more information about ModL functions.

Options on the options tab allow you to control when the equation block will execute. You can set the block to execute at the simulation initialization, at the end of the simulation, when the 3D window initializes, and several other choices. By default the equation is set to execute when necessary during the simulation.
Each input must be named in the Input Variables table in order to use it in the equation. Each output must be named in the Output Variables table. You can use the default input and output variable names or specify new names. ExtendSim will warn you if an input is used in the equation but it is not connected. This requirement is not the case for outputs. You can define an output variable that is not used in the equation, if desired.

The Equation block is discussed fully in the ExtendSim manual; refer to the manual for more information.

Warning: When doing arithmetic using integer variables, any resulting remainders will be lost.

Input Equation Variables: In addition to the properties on the item, the input variable types that can be defined are _DB Read Value, _DB Read Index, _DB Address, _DB Index, _Static first run init, _Static multi run init, _Connector:

_DB Read Value: This variable gets assigned a value from a fixed location in the ExtendSim database. If the this variable is pointing to a child field, the value assigned to it is from the parent's record value.

_DB Read PRI (Parent Record Index): Similar to the _DB Read Value variable, this variable points to a fixed location in the database. However in this case, the location it points to must be a child. That's because this variable takes on the parent's record index not the parent's record value. Consequently, this variable type can only be used if it is pointing at a child field.

_DB Address: This variable is assigned one or more elements of DB address, e.g., this variable could take on the value of a database index or a database index plus table index
plus field index plus a record index. This type of variable can be useful since some ModL database API functions require a DB Address as a parameter.

_DB Index: This variable contains a single DB address reference point, e.g. a reference to a record index only or maybe a field index only.

_Static First Run Init and _Static Multi Run Init: In general, a static variable is a local variable that retains its value between different calculations of the Equation block so they can be used to keep track of values over multiple calculations. The first run init will only initialize to the starting value on the first run of a multi run simulation, while the multi run init will initialize at the beginning of every run.

_Connector: The value of this variable is equal to the value of its associated input connector.

Output Equation Variables: In addition to the properties on the item, the output variable types (equation results) that can be defined are _DB Write Value, _DB Write Index, and _Connector:

_DB Write Value: This variable writes its value to a fixed location in the database. If this variable is pointing to a child field, the value being written must be a value that currently resides in the parent. Otherwise an error message will appear.

_DB Write PRI (Parent Record Index): Similar to the _DB Write Value variable, this variable points to a fixed location in the database. However in this case, the location it points to must always be a child. That's because in this case the value being written in interpreted as a parent's record index. Consequently, if the value being written to the
child is less than 1 or greater than the number of records in the parent, an error message will be posted.

Connector: The value of this variable defines the value for the associated output connector.

Connecters:
The input connectors correspond to the input variables in the Equation, as defined in the Input Variables table.

The output connectors correspond to the output variables in the Equation, as defined in the Output Variables table.

Tabs:

Equation Tab
Enter the equation...: Type an equation into the text entry box. The variables used in the Input Variables table and the Output Variables table should be used in the equation. Output variables should be assigned values. All input variables defined must be used in the equation.

Input Variables: Each row in this table defines one input variable that can be used in the equation. See the descriptions of the columns of the table to see how to define the variables.
Input Variables table Column 1 (Variable Type): This column of the table allows you to define the type of the input variable defined by this row. See above for the list of types that can be defined for inputs. Clicking on a cell will pop up a popup menu listing the possible types.

Input Variables table Column 2 (Variable Name): This column of the table allows you to define the name of the input variable defined by this row. This is the variable name that can be used in the text of the equation. Clicking on the cell will allow you to edit the variable name.

Input Variables table Column 3 (Variable Value): This column of the table displays different things for different Variable types. For Connector variables, it will display the last value that was present at the connector. For DB Read, DB Attribute, and DB Index types it will display a database reference, and can also be used to edit that reference by clicking on the cell. For Static Variable types, it displays both the initialization value for the static variable, and the current value of the static variable. The number to the left of the colon will be the initialization value, and the number to the right of the colon is the current value. Clicking on the cell will allow you to edit the initialization value.

Output Variables: Each row in this table defines one output variable that can be used in the equation. See the descriptions of the columns of the table to see how to define the variables.

Output Variables table Column 1 (Variable Type): This column of the table allows you to define the type of the output variable defined by this row. See above for the list of types that can be defined for outputs. Clicking on a cell will pop up a popup menu listing the possible types.
Output Variables table Column 2 (Variable Name): This column of the table allows you to define the name of the output variable defined by this row. This is the variable name that can be used in the text of the equation. Clicking on the cell will allow you to edit the variable name.

Output Variables table Column 3 (Variable Value): This column of the table displays different things for different Variable types. For Connector variables, it will display the last value that was output at the connector. For DB Write types it displays a database reference, and can also be used to edit that reference by clicking on the cell.

Test Equation: Clicking on this button will execute the equation.

Open Developer Reference: Clicking on this button will open the Developer Reference.

Options Tab
As necessary during the simulation (default): If this checkbox is checked, the equation will be executed during the simulation, as necessary. This means when a message comes into the block in a DE model, and on each step in a continuous model.

At the end of the simulation: If this checkbox is checked, the equation will be executed when the simulation ends.
Every __ time units: This option allows you to specify that the equation should be executed every Nth time unit during the model run. This option is only allowed in Discrete Event models.

Respond to link alerts: This option will cause the equation to be executed if one of the variable inputs is associated with a DB Read and if the data in the DB cell has changed.

Send GlobalProofStr to Proof: Assign a Proof command to GlobalProofStr to be sent to Proof. For example the equation:

```
outCon0 = inCon0;
GlobalProofStr = "WRITE ARRIVALS " + outCon0;
```

would send a write command to Proof to change the Arrivals message to the value of inCon0. This can be used to send virtually any Proof command. This option is available if Proof is installed.

Show input connector labels: Shows labels on the input connectors.

Show input connector values: Shows values on the input connectors.

Show output connector labels: Shows labels on the output connectors.

Show output connector values: Shows values on the output connectors.

Top equation result is time to next event: This option is associated with the every __ time unit's option. If this checkbox is checked, the top equation result will be used as the time to the next event. So each time the Equation is executed, the next time the equation is to be executed will be set by the top equation result.
Use block seed: This option allows you to set a block seed for the equation. This is only relevant if the block accesses random number functions.

When the 3D window is initialized: If this check box is checked, the equation will be executed when the E3D window is initialized.

When the simulation is initialized: If this check box is checked, the equation will be executed when the simulation is initialized.
APPENDIX H – EXTENDSIM THROW BLOCK

Symbol: Not connected

Purpose:
This block sends flow received by Catch Blocks or Merge blocks even though the blocks aren't connected by connection lines. The connection between the blocks is made in the dialog of the blocks by specifying in its dialog the label and block number of the block to connect.

The Throw Flow and Catch Flow blocks (and the Merge and Diverge blocks in certain modes as well) can be used to move flow without the use of connection lines. By creating a throw/catch connection via block dialogs, flow can be routed from a throwing type block to any catching type block in the model.

Connecters:
The input connector corresponds to the input flow which needs to be thrown.

Tabs:

Throw Tab
Block: This field is used to select the Catch or Merge block you want to throw to. The block number can be typed directly into the dialog or can be selected in the popup list on the right of the dialog item.

Position: If throwing to a Merge block, a position (i.e., a Merge branch) must be chosen because Merge blocks can have more than one catch branch. The position can be typed directly into the dialog or can be selected in the popup list on the right of the dialog item.

Show connection on icon: If this checkbox is checked, the catch type block you've selected this block to throw to will be displayed above this block's icon.

Filter available blocks: This checkbox enables the filtering capabilities. Since there can be many throw and catch type blocks in a model, it can be useful to organize them into groups. This can be used to create filtered lists or subsets of catch type blocks your Throw block can see.
Group: This is where you can choose a group to filter the list of catch type blocks that your Throw block can select.

Block type: This popup allows you to use block type to limit the list of potential catch type blocks your Throw block can select.

Only unconnected blocks: This checkbox further limits the list of potential catch type blocks to those blocks that have not already been assigned a "Throw/Catch" connection.

Count: This field displays the number of potential catch type blocks available to this Throw block.

My Group: This is where you assign your Throw block to a group. This is useful if you want the catch type blocks in your model to see a list of potential Throw blocks filtered by Group.

Show My Group on icon: If this checkbox is checked, the My Group selection will be displayed above this block's icon.

Options Tab
Select units: In the "Select units" frame, the first popup allows the user to choose the flow units for all the blocks residing in this block's unit group by either selecting an existing flow unit category or by creating a new one. A unit group is a collection of blocks connected together through flow connections and sharing the same flow unit. Notice that once you've made your selection here, the flow units for all blocks in this unit group will have changed. When the square button to the right of this popup menu is pressed, all blocks included within this block's unit group are selected. The second popup allows the user to define the time units for this particular Bias block.

Results Tab
Effective rate: This field displays the current effective rate of flow through this particular Throw block.

Total flow: This field displays the total volume of flow that has passed through this particular Throw block.

**APPENDIX I – EXTENDSIM CATCH BLOCK**

Symbol:

Purpose:
This block catches flow sent by Throw Blocks or Diverge blocks even though the blocks aren't connected by connection lines. The connection between the blocks is made in the dialog of the blocks by specifying in its dialog the label and block number of the block to connect.
The Throw Flow and Catch Flow blocks (and the Merge and Diverge blocks in certain modes as well) can be used to move flow without the use of connection lines. By creating a throw/catch connection via block dialogs, flow can be routed from a throwing type block to any catching type block in the model.

Connecters:
The output connector corresponds to the output flow which needs to be caught.

Tabs:

Catch Tab

Block: This field is used to select the Throw or Diverge block you want to catch from. The block number can be typed directly into the dialog or can be selected in the popuplist on the right of the dialog item.
Position: If catching from a Diverge block, a position (i.e., a Diverge branch) must be chosen because Diverge blocks can have more than one throw branch. The position can be typed directly into the dialog or can be selected in the popuplist on the right of the dialog item.

Show connection on icon: If this checkbox is checked, the throw type block you've selected for this block to catch from will be displayed above this block's icon.

Filter available blocks: This checkbox enables the filtering capabilities. Since there can be many throw and catch type blocks in a model, it can be useful to organize them into groups. This can be used to create filtered lists or subsets of throw type blocks your Catch block can select.

Group: This is where you can choose a group to filter the list of throw type blocks that your Catch block can select.

Block type: This popup allows you to use block type to limit the list of potential throw type blocks your Catch block can select.

Only unconnected blocks: This checkbox further limits the list of potential throw type blocks to those blocks that have not already been assigned a "Throw/Catch" connection.

Count: This field displays the number of potential throw type blocks available to this Catch block.

My Group: This is where you assign your Catch block to a group. This is useful if you want the throw type blocks in your model to see a list of potential Throw blocks filtered by Group.
Show My Group on icon: If this checkbox is checked, the My Group selection will be displayed above this block's icon.

Options Tab

Select units: In the "Select units" frame, the first popup allows the user to choose the flow units for all the blocks residing in this block's unit group by either selecting an existing flow unit category or by creating a new one. A unit group is a collection of blocks connected together through flow connections and sharing the same flow unit. Notice that once you've made your selection here, the flow units for all blocks in this unit group will have changed. When the square button to the right of this popup menu is pressed, all blocks included within this block's unit group are selected. The second popup allows the user to define the time units for this particular Bias block.
Results Tab

Effective rate: This field displays the current effective rate of flow through this particular Catch block.

Total flow: This field displays the total volume of flow that has passed through this particular Catch block.
APPENDIX J – SAMPLE OF CUSTOM BLOCK CODE IN MOD-L

This block receives values from a spreadsheet. One selects the file to read values from by typing its path name, or one can leave the name blank so ExtendSIM prompts for the name and fills in the path name. By typing in the sheet name, one may read the values from a specific sheet in the workbook. If the "Sheet name" is left blank, the values will be read from the top sheet. This block reads in the value at the cell defined by the values at the row and column connectors.

SAMPLE MOD-L CODE:
procedure GetNewEmbeddedWorkbookID()
{
    integer i,j;
    integer numbEmbeddedWorkbooks;
    integer workbookFound;
    string nextWorkbookName;
    integer lengthNextWorkbookName;
    string svWorkbookID;
    integer numberFound;
    string svTestNumber;
    real rvWorkbookID;

    if(badChosenEmbeddedWorkbookID == OLD_EMBEDDED_ID_DISABLED || useEmbeddedWorkbook_CB == FALSE)
        return;
XL_BuildEmbeddedWorkbookLists();  //build up embeddedWorkbookBlockNumsList, embeddedWorkbookNamesList

numbEmbeddedWorkbooks = GetDimension(embeddedWorkbookBlockNumsList);

//find which embedded workbook corresponds to badChosenEmbeddedWorkbookID
i = 0;
workbookFound = FALSE;
while(i < numbEmbeddedWorkbooks && workbookFound == FALSE)
{
    nextWorkbookName = embeddedWorkbookNamesList[i];
    lengthNextWorkbookName = StrLen(nextWorkbookName);
    svWorkbookID = "";
    numberFound = FALSE;

    //start at end of workbookName and work back until run into a "non-number" char
    for(j=1; j<=lengthNextWorkbookName; j++)
    {
        svTestNumber = StrPart(nextWorkbookName, lengthNextWorkbookName - j, j);
        if(StrToReal(svTestNumber))
        {
            numberFound = TRUE;
            svWorkbookID = svTestNumber;
        }
        else
            break;
    }
}
if(numberFound == TRUE)
    rvWorkbookID = StrToReal(svWorkbookID);
else
    rvWorkbookID = 0;
if(rvWorkbookID == badChosenEmbeddedWorkbookID)
    workbookFound = TRUE;
    i++;
} //while(i < numbEmbeddedWorkbooks || workbookFound == FALSE)
if(workbookFound == TRUE)
    chosenEmbeddedWorkbookID = embeddedWorkbookBlockNumsList[i-1];
else
    chosenEmbeddedWorkbookID = NO_EMBEDDED_WORKBOOK_CHOSEN;
    badChosenEmbeddedWorkbookID = OLD_EMBEDDED_ID_DISABLED;
} //procedure GetNewEmbeddedWorkbookID()

procedure ReceiveData()
{
    TranslateRowColumnIndices(rowIndex, columnIndex);

    if(!useEmbeddedWorkbook_CB && !FileIsOpen(externalWorkbookName_ET))
        OpenFile();

    if(connectionType_POP == DDE_OR_APPLE_EVENTSCONNECTION)
        //02/14/03
        

if (sheetName_ET == "")
    conversation = IPCConnect(server,externalWorkbookName_ET);
else
{
    setSheetStrng();
    conversation = IPCConnect(server,sheetStrng);
}
if (!conversation)
{
    userError("Extend failed to connect with "+server+". "+server+
may have been busy opening the spreadsheet "+
"when connection was attempted. Try running simulation again.
Call Technical Support if this occurs repeatedly.");
    abort;
}

DataOut = strToreal(IPCRequest(conversation,svRowColumn));
datarec = DataOut;
IPCDisconnect(conversation);
Conversation = 0;
}

else if(connectionType_POP == OLE_CONNECTION)
{
    worksheetRangeObjectHandle = XL_GetRangeObjectHandle();
    DataOut = DoPropertyGetReal(worksheetRangeObjectHandle,
_EXCEL_RANGE_VALUE);
    datarec = DataOut;
//procedure ReceiveData()

// This message occurs for each step in the simulation.

on simulate
{
    if (!advise && !discrete && (!triggerCon || (triggerCon && triggerIn > 0.5)))
        ReadVal(TRUE);
    if (advise)
        dataOut = dataRec;
}

on dataOut
{
    if ((readWhen == Out OR readWhen == Any) && !StartOnly)
    {
        if (triggerCon)
        {
            sendMsgToOutputs(triggerIn);
            if (triggerIn < 0.5)
                return;
        }
        if(RowCon)
        {
            sendMsgToOutputs(rowIn);
            rowIndex_P= RowIn;
        }
        if(ColCon)
        {
        }
    }
}
on RowIn
{
    if (readWhen == RowCol OR readWhen == Any)
    {
        if (triggerCon)
        {
            sendMsgToOutputs(triggerIn);
            if (triggerIn < 0.5)
                return;
        }
        
        if(ColCon)
            sendMsgToOutputs(colIn);

        ReadVal(TRUE);
    }
    
    sendMsgToInputs(DataOut);
}
if (readWhen == RowCol OR readWhen == Any)
{
    if (triggerCon)
    {
        sendMsgToOutputs(triggerIn);
        if (triggerIn < 0.5)
            return;
    }
    if (RowCon)
    {
        sendMsgToOutputs(rowIn);
        ReadVal(TRUE);
        sendMsgToInputs(DataOut);
    }
}

on triggerIn
{
    if (triggerIn < 0.5)
        return;
    if (RowCon)
    {
        sendMsgToOutputs(rowIn);
        rowIndex_P = RowIn;
    }
    if (ColCon)
    {
        185
sendMsgToOutputs(colIn);
columnIndex_P = ColIn;

} ReadVal(TRUE);
sendMsgToInputs(DataOut);

} on BlockReceive0
{
    if(FirstTime)
    {
        if(RowCon)
        {
            sendMsgToOutputs(rowIn);
            RowIndex_P = RowIn;
        }
        if(ColCon)
        {
            sendMsgToOutputs(colIn);
            ColumnIndex_P = ColIn;
        }
        ReadVal(FALSE);
    }
    SetDialogVariable(cloneInjectedBlockNumber_P, cloneInjectedDialogVariable_E T, dataOut, 0, 0);
}
// If the dialog data is inconsistent for simulation, abort.
on checkdata
{
    // 12/19/02 6.0 DJK added deletion of global arrays storing conversation
    // information in checkdata
    if(GaGetIndex("IPCCon") >= 0)
        GADispose("IPCCon");
    if(GaGetIndex("IPCFile") >= 0)
        GADispose("IPCFile");
    advise = FALSE;
    rowCon = rowIn;
    colCon = colIn;
    triggerCon = triggerIn;
    if (!rowCon)
        {
            if (noValue(rowIndex_P))
            {
                usererror("Data Receive Block number "+myNumber+" is missing
                cell location parameter. \n
                Please use the row and column connectors or enter values for the
                begining row and column in the block dialog.");
                abort;
            }
        }  
    if (!colCon)
        {
            if (noValue(columnIndex_P))
            
                
                abort;
            
        }
        
}
{ 
    usererror("Data Receive Block number "+myNumber+" is missing
cell location parameter. \n    Please use the row and column connectors or enter values for the
begining row and column in the block dialog.");
    abort;
}
}

if(!useEmbeddedWorkbook_CB)
{
    if(externalWorkbookName_ET == "")
    {
        UserError("Please choose a workbook for Data Receive block
number " +
                MyBlockNumber() + ").");
        abort;
    }

    else if(!FileIsOpen(externalWorkbookName_ET))
        OpenFile();
}

exec = sysGlobalint1; // the id number of the exec block**
APPENDIX K – SAMPLE DATA FOR FICTIONAL EXAMPLE AND USE OF THE MODEL

<table>
<thead>
<tr>
<th>Day</th>
<th>Feed</th>
<th>Scooping (OP)</th>
<th>Hauling</th>
<th>Mucking</th>
<th>Hauling (UG)</th>
<th>Crushing</th>
<th>Skipping</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>In</td>
<td>Out</td>
<td>In</td>
<td>Out</td>
<td>In</td>
<td>Out</td>
</tr>
<tr>
<td>1</td>
<td>7067</td>
<td>4075</td>
<td>7462</td>
<td>4780</td>
<td>4780</td>
<td>4813</td>
<td>5883</td>
</tr>
<tr>
<td>2</td>
<td>6806</td>
<td>3090</td>
<td>4473</td>
<td>3448</td>
<td>6467</td>
<td>5305</td>
<td>6557</td>
</tr>
<tr>
<td>3</td>
<td>7294</td>
<td>4825</td>
<td>5778</td>
<td>5454</td>
<td>6709</td>
<td>4377</td>
<td>6103</td>
</tr>
<tr>
<td>4</td>
<td>5997</td>
<td>3116</td>
<td>4967</td>
<td>4238</td>
<td>6605</td>
<td>5144</td>
<td>6545</td>
</tr>
<tr>
<td>5</td>
<td>7175</td>
<td>3456</td>
<td>6124</td>
<td>4429</td>
<td>5554</td>
<td>6206</td>
<td>6539</td>
</tr>
<tr>
<td>6</td>
<td>6545</td>
<td>3704</td>
<td>5696</td>
<td>4878</td>
<td>7717</td>
<td>4942</td>
<td>4942</td>
</tr>
<tr>
<td>7</td>
<td>6060</td>
<td>3000</td>
<td>6626</td>
<td>4028</td>
<td>8587</td>
<td>5526</td>
<td>6862</td>
</tr>
<tr>
<td>8</td>
<td>7933</td>
<td>4161</td>
<td>6782</td>
<td>4961</td>
<td>4961</td>
<td>4756</td>
<td>6320</td>
</tr>
<tr>
<td>9</td>
<td>6778</td>
<td>5201</td>
<td>5988</td>
<td>4707</td>
<td>5661</td>
<td>5161</td>
<td>5843</td>
</tr>
<tr>
<td>10</td>
<td>7047</td>
<td>3024</td>
<td>7401</td>
<td>4269</td>
<td>7186</td>
<td>5698</td>
<td>7315</td>
</tr>
<tr>
<td>11</td>
<td>7622</td>
<td>2297</td>
<td>5770</td>
<td>3807</td>
<td>6275</td>
<td>4617</td>
<td>4401</td>
</tr>
<tr>
<td>12</td>
<td>6730</td>
<td>2889</td>
<td>5637</td>
<td>4144</td>
<td>7026</td>
<td>5596</td>
<td>5117</td>
</tr>
<tr>
<td>13</td>
<td>5869</td>
<td>4486</td>
<td>5716</td>
<td>5821</td>
<td>7753</td>
<td>2822</td>
<td>6514</td>
</tr>
<tr>
<td>14</td>
<td>7403</td>
<td>3452</td>
<td>5462</td>
<td>4761</td>
<td>7057</td>
<td>5327</td>
<td>6519</td>
</tr>
<tr>
<td>15</td>
<td>6039</td>
<td>3671</td>
<td>6236</td>
<td>4848</td>
<td>6281</td>
<td>5357</td>
<td>5711</td>
</tr>
<tr>
<td>16</td>
<td>7186</td>
<td>4401</td>
<td>5135</td>
<td>5942</td>
<td>6912</td>
<td>4548</td>
<td>6441</td>
</tr>
<tr>
<td>17</td>
<td>6700</td>
<td>2191</td>
<td>6333</td>
<td>3495</td>
<td>7094</td>
<td>5324</td>
<td>5605</td>
</tr>
<tr>
<td>18</td>
<td>7101</td>
<td>3842</td>
<td>8643</td>
<td>4062</td>
<td>4062</td>
<td>4168</td>
<td>7179</td>
</tr>
<tr>
<td>19</td>
<td>7668</td>
<td>5203</td>
<td>4717</td>
<td>5167</td>
<td>5167</td>
<td>4132</td>
<td>5604</td>
</tr>
<tr>
<td>20</td>
<td>7313</td>
<td>2296</td>
<td>5799</td>
<td>3496</td>
<td>7132</td>
<td>4528</td>
<td>5122</td>
</tr>
<tr>
<td>21</td>
<td>7388</td>
<td>4128</td>
<td>5660</td>
<td>5131</td>
<td>7087</td>
<td>5378</td>
<td>5378</td>
</tr>
<tr>
<td>22</td>
<td>5939</td>
<td>2222</td>
<td>6046</td>
<td>3340</td>
<td>5876</td>
<td>6062</td>
<td>5542</td>
</tr>
<tr>
<td></td>
<td>50</td>
<td>6259</td>
<td>4077</td>
<td>5679</td>
<td>5523</td>
<td>6396</td>
<td>5026</td>
</tr>
<tr>
<td>---</td>
<td>-----</td>
<td>------</td>
<td>------</td>
<td>------</td>
<td>------</td>
<td>------</td>
<td>------</td>
</tr>
<tr>
<td>51</td>
<td>7025</td>
<td>4051</td>
<td>5957</td>
<td>5281</td>
<td>7174</td>
<td>3591</td>
<td>5020</td>
</tr>
<tr>
<td>52</td>
<td>6969</td>
<td>5042</td>
<td>5516</td>
<td>4662</td>
<td>4662</td>
<td>4201</td>
<td>5639</td>
</tr>
<tr>
<td>53</td>
<td>7090</td>
<td>2915</td>
<td>5168</td>
<td>3999</td>
<td>7870</td>
<td>5472</td>
<td>5472</td>
</tr>
<tr>
<td>54</td>
<td>7394</td>
<td>3641</td>
<td>5048</td>
<td>4916</td>
<td>7433</td>
<td>5897</td>
<td>6108</td>
</tr>
<tr>
<td>55</td>
<td>7705</td>
<td>4725</td>
<td>4905</td>
<td>5860</td>
<td>5753</td>
<td>5880</td>
<td>6106</td>
</tr>
<tr>
<td>56</td>
<td>6360</td>
<td>3059</td>
<td>5644</td>
<td>4119</td>
<td>6222</td>
<td>4247</td>
<td>6763</td>
</tr>
<tr>
<td>57</td>
<td>8320</td>
<td>3035</td>
<td>6253</td>
<td>4093</td>
<td>6234</td>
<td>4823</td>
<td>4823</td>
</tr>
<tr>
<td>58</td>
<td>7497</td>
<td>3896</td>
<td>7582</td>
<td>5350</td>
<td>6266</td>
<td>4722</td>
<td>5959</td>
</tr>
<tr>
<td>59</td>
<td>7231</td>
<td>3213</td>
<td>6754</td>
<td>4635</td>
<td>4985</td>
<td>6685</td>
<td>7168</td>
</tr>
<tr>
<td>60</td>
<td>6673</td>
<td>2828</td>
<td>6180</td>
<td>4232</td>
<td>7654</td>
<td>4185</td>
<td>5346</td>
</tr>
<tr>
<td>61</td>
<td>6786</td>
<td>2871</td>
<td>7708</td>
<td>4286</td>
<td>7479</td>
<td>5374</td>
<td>6276</td>
</tr>
<tr>
<td>62</td>
<td>7350</td>
<td>4742</td>
<td>7762</td>
<td>5937</td>
<td>7014</td>
<td>4850</td>
<td>7590</td>
</tr>
<tr>
<td>63</td>
<td>7582</td>
<td>2545</td>
<td>6993</td>
<td>4021</td>
<td>5949</td>
<td>6094</td>
<td>7043</td>
</tr>
<tr>
<td>64</td>
<td>7212</td>
<td>4190</td>
<td>6982</td>
<td>4846</td>
<td>4846</td>
<td>4892</td>
<td>5851</td>
</tr>
<tr>
<td>65</td>
<td>6429</td>
<td>2749</td>
<td>5238</td>
<td>3902</td>
<td>8047</td>
<td>4526</td>
<td>8022</td>
</tr>
<tr>
<td>66</td>
<td>7501</td>
<td>3293</td>
<td>8266</td>
<td>4585</td>
<td>7121</td>
<td>4610</td>
<td>5434</td>
</tr>
<tr>
<td>67</td>
<td>7070</td>
<td>2759</td>
<td>5912</td>
<td>4055</td>
<td>5837</td>
<td>5966</td>
<td>8317</td>
</tr>
<tr>
<td>68</td>
<td>6913</td>
<td>1987</td>
<td>5532</td>
<td>3275</td>
<td>7296</td>
<td>5476</td>
<td>5933</td>
</tr>
<tr>
<td>69</td>
<td>6125</td>
<td>2720</td>
<td>7121</td>
<td>4032</td>
<td>4590</td>
<td>3570</td>
<td>6658</td>
</tr>
<tr>
<td>70</td>
<td>7041</td>
<td>2614</td>
<td>7134</td>
<td>3783</td>
<td>7106</td>
<td>4480</td>
<td>6365</td>
</tr>
<tr>
<td>71</td>
<td>7670</td>
<td>3223</td>
<td>7962</td>
<td>4271</td>
<td>4790</td>
<td>4341</td>
<td>7814</td>
</tr>
<tr>
<td>72</td>
<td>7355</td>
<td>3821</td>
<td>5399</td>
<td>4664</td>
<td>4664</td>
<td>4871</td>
<td>5213</td>
</tr>
<tr>
<td>73</td>
<td>7737</td>
<td>4849</td>
<td>5319</td>
<td>6096</td>
<td>6162</td>
<td>5740</td>
<td>7235</td>
</tr>
<tr>
<td>74</td>
<td>6344</td>
<td>4691</td>
<td>7987</td>
<td>5910</td>
<td>5944</td>
<td>6976</td>
<td>6383</td>
</tr>
<tr>
<td>75</td>
<td>7024</td>
<td>3644</td>
<td>5806</td>
<td>4716</td>
<td>5012</td>
<td>4221</td>
<td>5606</td>
</tr>
<tr>
<td>76</td>
<td>6218</td>
<td>2582</td>
<td>6525</td>
<td>3666</td>
<td>6684</td>
<td>5079</td>
<td>7400</td>
</tr>
<tr>
<td>158</td>
<td>7349</td>
<td>3930</td>
<td>6220</td>
<td>5285</td>
<td>6045</td>
<td>4306</td>
<td>5332</td>
</tr>
<tr>
<td>------</td>
<td>------</td>
<td>------</td>
<td>------</td>
<td>------</td>
<td>------</td>
<td>------</td>
<td>------</td>
</tr>
<tr>
<td>159</td>
<td>7722</td>
<td>3999</td>
<td>7629</td>
<td>5206</td>
<td>7469</td>
<td>3318</td>
<td>8075</td>
</tr>
<tr>
<td>160</td>
<td>6431</td>
<td>2962</td>
<td>7140</td>
<td>4091</td>
<td>4179</td>
<td>5401</td>
<td>6854</td>
</tr>
<tr>
<td>161</td>
<td>7169</td>
<td>3704</td>
<td>6220</td>
<td>4992</td>
<td>5907</td>
<td>5520</td>
<td>5653</td>
</tr>
<tr>
<td>162</td>
<td>6513</td>
<td>2745</td>
<td>4480</td>
<td>4275</td>
<td>7224</td>
<td>5594</td>
<td>5126</td>
</tr>
<tr>
<td>163</td>
<td>6695</td>
<td>1842</td>
<td>6113</td>
<td>3184</td>
<td>7203</td>
<td>5484</td>
<td>6496</td>
</tr>
<tr>
<td>164</td>
<td>5807</td>
<td>3781</td>
<td>5959</td>
<td>5266</td>
<td>6973</td>
<td>4607</td>
<td>4667</td>
</tr>
<tr>
<td>165</td>
<td>6663</td>
<td>4631</td>
<td>5913</td>
<td>5992</td>
<td>6360</td>
<td>5993</td>
<td>5657</td>
</tr>
<tr>
<td>166</td>
<td>7391</td>
<td>2901</td>
<td>5794</td>
<td>4110</td>
<td>4479</td>
<td>5373</td>
<td>6266</td>
</tr>
<tr>
<td>167</td>
<td>6827</td>
<td>3487</td>
<td>5260</td>
<td>4573</td>
<td>6300</td>
<td>4951</td>
<td>5969</td>
</tr>
<tr>
<td>168</td>
<td>7566</td>
<td>2196</td>
<td>4270</td>
<td>3311</td>
<td>5006</td>
<td>5258</td>
<td>7155</td>
</tr>
<tr>
<td>169</td>
<td>7919</td>
<td>2862</td>
<td>6494</td>
<td>4314</td>
<td>6554</td>
<td>4793</td>
<td>6723</td>
</tr>
<tr>
<td>170</td>
<td>7052</td>
<td>4973</td>
<td>5463</td>
<td>6356</td>
<td>6586</td>
<td>3410</td>
<td>4688</td>
</tr>
<tr>
<td>171</td>
<td>7198</td>
<td>2744</td>
<td>6893</td>
<td>3959</td>
<td>7301</td>
<td>4479</td>
<td>4479</td>
</tr>
<tr>
<td>172</td>
<td>6364</td>
<td>3380</td>
<td>7767</td>
<td>4467</td>
<td>6931</td>
<td>4267</td>
<td>6090</td>
</tr>
<tr>
<td>173</td>
<td>7125</td>
<td>2441</td>
<td>6684</td>
<td>4099</td>
<td>5765</td>
<td>5426</td>
<td>6943</td>
</tr>
<tr>
<td>174</td>
<td>6854</td>
<td>2722</td>
<td>5521</td>
<td>3845</td>
<td>6365</td>
<td>4876</td>
<td>6678</td>
</tr>
<tr>
<td>175</td>
<td>6924</td>
<td>3934</td>
<td>5560</td>
<td>5477</td>
<td>5710</td>
<td>3526</td>
<td>6733</td>
</tr>
<tr>
<td>176</td>
<td>6360</td>
<td>2613</td>
<td>7365</td>
<td>4015</td>
<td>6393</td>
<td>5803</td>
<td>5803</td>
</tr>
<tr>
<td>177</td>
<td>7188</td>
<td>3335</td>
<td>5975</td>
<td>4416</td>
<td>7592</td>
<td>4527</td>
<td>5548</td>
</tr>
<tr>
<td>178</td>
<td>7068</td>
<td>4112</td>
<td>8085</td>
<td>5816</td>
<td>7646</td>
<td>5373</td>
<td>5289</td>
</tr>
<tr>
<td>179</td>
<td>6896</td>
<td>5076</td>
<td>5718</td>
<td>5680</td>
<td>5680</td>
<td>4135</td>
<td>4135</td>
</tr>
<tr>
<td>180</td>
<td>7676</td>
<td>3322</td>
<td>5444</td>
<td>4457</td>
<td>6276</td>
<td>5643</td>
<td>6583</td>
</tr>
<tr>
<td>181</td>
<td>7703</td>
<td>4095</td>
<td>6819</td>
<td>5435</td>
<td>7037</td>
<td>4981</td>
<td>4981</td>
</tr>
<tr>
<td>182</td>
<td>6975</td>
<td>3847</td>
<td>5579</td>
<td>5056</td>
<td>5244</td>
<td>5524</td>
<td>5887</td>
</tr>
<tr>
<td>183</td>
<td>6637</td>
<td>3604</td>
<td>6199</td>
<td>4765</td>
<td>7685</td>
<td>4602</td>
<td>4393</td>
</tr>
<tr>
<td>184</td>
<td>6109</td>
<td>4606</td>
<td>8381</td>
<td>4230</td>
<td>4230</td>
<td>4229</td>
<td>5567</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>293</td>
<td>7060</td>
<td>2792</td>
<td>7965</td>
<td>4210</td>
<td>7285</td>
<td>5487</td>
<td>6049</td>
</tr>
<tr>
<td>294</td>
<td>6707</td>
<td>3083</td>
<td>6793</td>
<td>4240</td>
<td>4536</td>
<td>5178</td>
<td>5343</td>
</tr>
<tr>
<td>295</td>
<td>6521</td>
<td>3836</td>
<td>4479</td>
<td>5041</td>
<td>4897</td>
<td>4710</td>
<td>8187</td>
</tr>
<tr>
<td>296</td>
<td>6627</td>
<td>3577</td>
<td>6687</td>
<td>4820</td>
<td>6896</td>
<td>4388</td>
<td>6385</td>
</tr>
<tr>
<td>297</td>
<td>5939</td>
<td>3369</td>
<td>6222</td>
<td>4717</td>
<td>6981</td>
<td>5618</td>
<td>5734</td>
</tr>
<tr>
<td>298</td>
<td>6358</td>
<td>3656</td>
<td>7383</td>
<td>5035</td>
<td>6744</td>
<td>4906</td>
<td>4661</td>
</tr>
<tr>
<td>299</td>
<td>5835</td>
<td>4804</td>
<td>5320</td>
<td>5978</td>
<td>6596</td>
<td>4884</td>
<td>6372</td>
</tr>
<tr>
<td>300</td>
<td>7283</td>
<td>4499</td>
<td>6768</td>
<td>5705</td>
<td>7477</td>
<td>5099</td>
<td>6355</td>
</tr>
<tr>
<td>301</td>
<td>6679</td>
<td>3353</td>
<td>6456</td>
<td>4720</td>
<td>5079</td>
<td>4080</td>
<td>4080</td>
</tr>
<tr>
<td>302</td>
<td>7196</td>
<td>5586</td>
<td>6820</td>
<td>4467</td>
<td>7146</td>
<td>5030</td>
<td>5896</td>
</tr>
<tr>
<td>303</td>
<td>7374</td>
<td>3063</td>
<td>5841</td>
<td>4261</td>
<td>7840</td>
<td>4577</td>
<td>6280</td>
</tr>
<tr>
<td>304</td>
<td>7245</td>
<td>3703</td>
<td>4978</td>
<td>4893</td>
<td>6900</td>
<td>4955</td>
<td>6406</td>
</tr>
<tr>
<td>305</td>
<td>6311</td>
<td>4002</td>
<td>5901</td>
<td>5227</td>
<td>7465</td>
<td>5152</td>
<td>6788</td>
</tr>
<tr>
<td>306</td>
<td>8320</td>
<td>3739</td>
<td>8634</td>
<td>4914</td>
<td>5241</td>
<td>4039</td>
<td>6017</td>
</tr>
<tr>
<td>307</td>
<td>6623</td>
<td>4930</td>
<td>5651</td>
<td>5305</td>
<td>6316</td>
<td>4990</td>
<td>6443</td>
</tr>
<tr>
<td>308</td>
<td>6613</td>
<td>3676</td>
<td>8564</td>
<td>4749</td>
<td>7321</td>
<td>6058</td>
<td>6488</td>
</tr>
<tr>
<td>309</td>
<td>7746</td>
<td>2427</td>
<td>5201</td>
<td>3682</td>
<td>5960</td>
<td>4800</td>
<td>7056</td>
</tr>
<tr>
<td>310</td>
<td>6939</td>
<td>3562</td>
<td>4535</td>
<td>4850</td>
<td>7468</td>
<td>5367</td>
<td>5367</td>
</tr>
<tr>
<td>311</td>
<td>6193</td>
<td>3121</td>
<td>6358</td>
<td>4319</td>
<td>7163</td>
<td>4742</td>
<td>6286</td>
</tr>
<tr>
<td>312</td>
<td>7645</td>
<td>3246</td>
<td>6387</td>
<td>4339</td>
<td>6965</td>
<td>4299</td>
<td>5926</td>
</tr>
<tr>
<td>313</td>
<td>7530</td>
<td>4325</td>
<td>7665</td>
<td>5362</td>
<td>6552</td>
<td>4717</td>
<td>6342</td>
</tr>
<tr>
<td>314</td>
<td>6522</td>
<td>3838</td>
<td>7938</td>
<td>5114</td>
<td>8062</td>
<td>5363</td>
<td>6623</td>
</tr>
<tr>
<td>315</td>
<td>7530</td>
<td>2768</td>
<td>5167</td>
<td>4042</td>
<td>4914</td>
<td>4842</td>
<td>7882</td>
</tr>
<tr>
<td>316</td>
<td>6292</td>
<td>2518</td>
<td>7885</td>
<td>3739</td>
<td>6852</td>
<td>4213</td>
<td>6278</td>
</tr>
<tr>
<td>317</td>
<td>6946</td>
<td>4030</td>
<td>3840</td>
<td>4929</td>
<td>7224</td>
<td>4355</td>
<td>6095</td>
</tr>
<tr>
<td>318</td>
<td>6136</td>
<td>2621</td>
<td>5942</td>
<td>4010</td>
<td>7381</td>
<td>5802</td>
<td>6264</td>
</tr>
<tr>
<td>319</td>
<td>6239</td>
<td>4075</td>
<td>5451</td>
<td>5169</td>
<td>7273</td>
<td>3508</td>
<td>6660</td>
</tr>
</tbody>
</table>
REFERENCES


6. Jane Eleanor Doe. “Formatting a PhD Level Thesis: An example for PhD level students to follow”. Faculty of Graduate Studies, Dalhousie University, 2009


