

2012

## MULTIMODAL PERFORMANCE AND RELIABILITY ANALYSIS OF THE PIERRE AUGER OBSERVATORY NORTH SITE

Maira Maskevics  
*Michigan Technological University*

Follow this and additional works at: <https://digitalcommons.mtu.edu/etds>



Part of the [Computer Engineering Commons](#)

Copyright 2012 Maira Maskevics

---

### Recommended Citation

Maskevics, Maira, "MULTIMODAL PERFORMANCE AND RELIABILITY ANALYSIS OF THE PIERRE AUGER OBSERVATORY NORTH SITE", Master's Thesis, Michigan Technological University, 2012.  
<https://doi.org/10.37099/mtu.dc.etds/469>

Follow this and additional works at: <https://digitalcommons.mtu.edu/etds>



Part of the [Computer Engineering Commons](#)

MULTIMODAL PERFORMANCE AND RELIABILITY ANALYSIS OF THE  
PIERRE AUGER OBSERVATORY NORTHERN SITE  
COMMUNICATIONS SYSTEM,

By

Maira Maskevics

A THESIS

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE

In Computer Engineering

MICHIGAN TECHNOLOGICAL UNIVERSITY

2012

© 2012 Maira Maskevics

This thesis has been approved in partial fulfillment of the requirements for the Degree of  
MASTER OF SCIENCE in Computer Engineering.

Department of Electrical and Computer Engineering

Thesis Advisor: *Dr. Roger M. Kieckhafer*

Committee Member: *Dr. Daniel R. Fuhrmann*

Committee Member: *Dr. David F. Nitz*

Department Chair: *Dr. Daniel R. Fuhrmann*

# Table of Contents

Table of Contents .....	3
List of Figures .....	6
List of Tables .....	8
Acknowledgments.....	10
Abstract.....	11
1 Introduction .....	12
1.1 Auger South and North.....	13
1.1.1 Auger South .....	13
1.1.2 Auger North .....	13
1.2 Auger North Communication Challenge .....	14
2 Background .....	17
2.1 WAHREN Topology and Physical Layouts .....	17
2.1.1 Second-Order Power Chains and Rings.....	18
2.1.2 Higher-Order Chains and Rings.....	19
2.1.3 Non-Linear Physical Layouts.....	19
2.2 WAHREN Medium Access Control (MAC) Protocol .....	20
2.3 WAHREN Scheduling Protocol .....	22
2.4 Adaptation of WAHREN to Auger North .....	23
2.4.1 Auger North Topology and Physical Layouts.....	23
2.4.2 Auger North MAC Protocol.....	25
2.4.3 Auger North Scheduling Protocol.....	26
2.4.4 Common Mode Faults.....	26
3 Multimodal Approach and Methodology.....	28

3.1	Auger North Canonical Sector.....	28
3.2	Architecture of the Multimodal Model.....	30
4	GSPN Analysis of $f_{max}$ and $F(f)$ .....	33
4.1	Context of $f_{max}$ and $F(f)$ .....	33
4.2	Introduction to GSPNs.....	33
4.2.1	Petri Net Basics.....	33
4.2.2	Generalized Stochastic Petri Nets (GSPNs).....	38
4.2.3	State Space Explosion and Truncation.....	39
4.3	Stochastic Petri Net Package (SPNP).....	40
4.4	Modeling of Tank Availability.....	40
4.5	Numerical Results.....	42
5	Exhaustive Search for $C_{ij}(T)$ .....	43
5.1	Context of $C_{ij}(T)$ .....	43
5.1.1	Variation in $C_{ij}(T)$ at a Given Depth.....	43
5.2	Methodology.....	45
5.3	Numerical Results.....	45
6	Combinatorial Analysis of $C_i(T)$ .....	49
6.1	Context of $C_i(T)$ .....	49
6.2	Methodology.....	49
6.3	Numerical Results.....	49
7	Binomial Analysis of MER.....	51
7.1	Context of MER.....	51
7.2	Methodology.....	51
7.3	Numerical Results.....	52

8	DTMC Analysis of $q_i(d(T))$ .....	54
8.1	Context of $q_i(d(T))$ .....	54
8.2	Methodology.....	54
8.2.1	Case-0.....	54
8.2.2	Macro-Retransmission .....	59
8.2.3	Micro-Retransmission.....	61
8.2.4	Combined-Retransmission.....	65
8.3	Numerical Results.....	66
9	Final Top-Level Results.....	72
9.1	Calculations Explained.....	72
9.2	Sector Results .....	76
9.3	System Results.....	78
10	Conclusions.....	80
11	Future Work.....	81
	References.....	82
	Appendix A – GSPN.....	85
	A.a GSPN, Sector Fault Model – with Correlated Tank Faults .....	85
	A.b GSPN, Sector Repair & Truncation Subnet.....	86
	Appendix B – SPNP Code.....	87
	Appendix C – GSPN Results, $F(f)$ .....	94
	Appendix D – C-Program, Exhaustive Search, $C_{ij}(T)$ .....	95
	Appendix E – Exhaustive Search Results, $C_{ij}(T)$ .....	105
	E.a $C_{ij}(T)$ , $i=0$ .....	105
	E.b $C_{ij}(T)$ , $i=1$ .....	109

<i>E.c</i> $C_{ij}(T)$ , $i=2$ .....	113
<i>E.d</i> $C_{ij}(T)$ , $i=3$ .....	117
<i>E.e</i> $C_{ij}(T)$ , $i=4$ .....	121
Appendix F – Combinatorial Algebra Results, $C_i(T)$ .....	125
Appendix G – DTMC Spreadsheet .....	129
G.a Case-0 .....	129
G.b Micro .....	130
G.c Macro .....	131
G.d Combined .....	132
Appendix H – DTMC Results, $q_i(d(T))$ .....	133
H.a Case-0 DTMC Results, $q_i(d(T))$ .....	133
H.b Macro DTMC Results, $q_i(d(T))$ .....	137
H.c Micro DTMC Results, $q_i(d(T))$ .....	141
H.d Combo DTMC Results, $q_i(d(T))$ .....	145
Appendix I – Combinatorial Algebra Results, $Q(T)$ .....	149
Appendix J – Tank Reliability Results, $R(T)$ .....	153
Appendix K – Final Results .....	157
K.a Sector Results .....	157
K.b System Results .....	157

## List of Figures

Figure 1: Location of Auger North and its size relative to Auger South .....	14
Figure 2: Graph topology of a 2 <sup>nd</sup> -order power chain .....	18
Figure 3: Physical layout of a wireless 2 <sup>nd</sup> -order power chain, as used in Auger North ..	19

Figure 4: Mobius Fold in a 2 <sup>nd</sup> -order power chain .....	20
Figure 5: Hybrid TDMA/CSMA window.....	22
Figure 6: Single-Source Systolic Broadcast .....	22
Figure 7: Concentrator Station service area with one sector partitioned .....	24
Figure 8: Embedding of 2 <sup>nd</sup> -order power chains into one “sector” of the Auger North SD Array .....	24
Figure 9: Local Dynamic Rerouting around two adjacent faulty nodes. ....	26
Figure 10: Canonical Sector Tank ID Layout.....	29
Figure 11: Canonical Sector Tank Depth Layout .....	29
Figure 12: Architecture of Multimodal Model and Sub-models .....	32
Figure 13: Petri Net Symbols.....	34
Figure 14: Basic Petri Net Transition Firing.....	36
Figure 15: AND and OR Equivalents in Petri Nets .....	37
Figure 16: Petri Net Conflict.....	37
Figure 17: Inhibitory Arcs in Petri Nets .....	38
Figure 18: Multiple Arcs in a Petri Net .....	38
Figure 19: Tank $C_j$ Illustration .....	45
Figure 20: $C_{ij}(T)$ Results, $i = 0$ .....	46
Figure 21: $C_{ij}(T)$ Results, $i = 1$ .....	47
Figure 22: $C_{ij}(T)$ Results, $i = 2$ .....	47
Figure 23: $C_{ij}(T)$ Results, $i = 3$ .....	48
Figure 24: $C_{ij}(T)$ Results, $i = 4$ .....	48
Figure 25: $C_i(T)$ Results .....	50
Figure 26: Case-0 Before-and-After State Diagram .....	55
Figure 27: Case-0 DTMC State Transition Diagram.....	57
Figure 28: Micro-Retransmission State Transition Diagram.....	62
Figure 29: Micro-Retransmission DTMC State Transition Diagram .....	63
Figure 30: Originating Message from Tank ID 129, CR $q_i(d(T))$ .....	71
Figure 31: Messages Lost per Year Sector Results .....	78
Figure 32: Messages Lost per Year System Results.....	79



## List of Tables

Table 1: Fault Probabilities per Sector.....	42
Table 2: $C_i(T)$ Comparison of Tanks at same depth .....	50
Table 3: MERs for Relevant Message Types .....	53
Table 4: Case-0 DTMC Equations.....	57
Table 5: Case 0 Arrival Probabilities at the Two Concentrator Radios (CRs) from Tank ID 129, with Zero Faults in the Path.....	58
Table 6: Case 0 Arrival Probabilities at the Two Concentrator Radios (CRs) from Tank ID 129, with One Fault in the Path .....	59
Table 7: Macro-Retransmission Arrival Probabilities between the CRs and Tank ID 129, Zero Faults in the Path .....	61
Table 8: Macro-Retransmission Arrival Probabilities between the CRs and Tank ID 129, with One Fault in the Path .....	61
Table 9: Micro-Retransmit DTMC Equations .....	63
Table 10: Micro-Retransmission Arrival Probabilities at the Two Concentrator Radios (CRs) from Tank ID 129, with Zero Faults in the Path .....	64
Table 11: Micro-Retransmission Arrival Probabilities at the Two Concentrator Radios (CRs) from Tank ID 129, with One Fault in the Path.....	64
Table 12: Macro-Retransmission Arrival Probabilities at the Two Concentrator Radios (CRs) from Tank ID 129, Zero Faults in the Path .....	66
Table 13: Macro-Retransmission Arrival Probabilities at the Two Concentrator Radios (CRs) from Tank ID 129, One Fault in the Path.....	66
Table 14: DTMC Spreadsheet: MER Calculation .....	67
Table 15: DTMC Spreadsheet, Initial State .....	68
Table 16: DTMC Spreadsheet, MER and MDR at Each Hop .....	69
Table 17: DTMC Spreadsheet, Resulting $Q(h)$ .....	70
Table 18: $C_{ij}(T)$ of Tank ID 7 .....	73

Table 19: $C_i(T)$ of Tank ID 7 .....	74
Table 20: $q_i(d(T))$ of Tank ID 7 .....	75
Table 21: $Q(T)$ of Tank ID 7 .....	76
Table 22: Sector Results .....	77
Table 23: System Results .....	79

## **Acknowledgments**

I am grateful to my advisor Dr. Roger M. Kieckhafer for his guidance and support throughout this research. I would like to thank my thesis committee members, Dr. David F. Nitz and Dr. Daniel R. Fuhrmann for their advice and time. I also acknowledge early preliminary work done by James Hollenbeck that helped to narrow down the range of model evaluation methods. Finally, I would like to thank Dr. Kieckhafer, Dr. Nitz and the Pierre Auger Collaboration for giving me the opportunity to do this work.

## **Abstract**

The Pierre Auger Cosmic Ray Observatory North site employs a large array of surface detector stations (tanks) to detect the secondary particle showers generated by ultra-high energy cosmic rays. Due to the rare nature of ultra-high energy cosmic rays, it is important to have a high reliability on tank communications, ensuring no valuable data is lost. The Auger North site employs a peer-to-peer paradigm, the *Wireless Architecture for Hard Real-Time Embedded Networks* (WAHREN), designed specifically for highly reliable message delivery over fixed networks, under hard real-time deadlines.

The WAHREN design included two retransmission protocols, Micro- and Macro-retransmission. To fully understand how each retransmission protocol increased the reliability of communications, this analysis evaluated the system without using either retransmission protocol (Case-0), both Micro- and Macro-retransmission individually (Micro and Macro), and Micro- and Macro-retransmission combined.

This thesis used a multimodal modeling methodology to prove that a performance and reliability analysis of WAHREN was possible, and provided the results of the analysis. A multimodal approach was necessary because these processes were driven by different mathematical models. The results from this analysis can be used as a framework for making design decisions for the Auger North communication system.

# 1 Introduction

The Pierre Auger Cosmic Ray Observatory is a multi-national project for research on ultra-high energy cosmic rays [1,5,6]. Of particular interest are extremely rare cosmic ray events above 10 EeV ( $10^{19}$ eV) which arrive only at a rate of approximately one particle per square kilometer per year [1]. It is not possible to directly observe the primary particles because they are annihilated by interactions in the upper atmosphere. However, these interactions produce an “air shower” of secondary particles that do reach the Earth’s surface (also referred to as “events”). The observatory employs an array of Surface Detector (SD) stations to detect the shower as it impacts the ground. Since the footprint of the shower can cover several square kilometers, SD stations are deployed over a wide geographical area. Due to their remoteness, the stations are solar powered and communicate with the observatory campus via a wireless network.

The SD array is essentially a fixed-infrastructure Wireless Sensor Network (WSN); therefore both node failures and transmission errors will be commonplace. Reliability of message delivery is critical due to the extreme rarity of the ultra-high energy cosmic rays and the real-time deadlines of message delivery.

Hard real-time deadlines are placed on time-critical messages to prevent buffer overrun, resulting in the loss of data at the SD stations. Critical messages that come from the SD stations are called Trigger-2 (T2) messages. Originating from a SD, T2 messages are transmitted to Concentrator Radios (CR), which forward the messages to a Concentrator Station (CS). The CS bundles messages and sends them to the observatory campus via an optical fiber network. T2 messages consist of a timestamp and promising event data. Though an actual event happens rarely, many triggers will be detected per second in the system. For example, Auger South receives approximately 400 triggers per day, but only 2% are real showers [2]. The observatory campus correlates T2 data and timestamps across SDs to determine if the triggers are from random noise in the atmosphere, or if it is an actual event. If an event is detected, the campus will request further information from the SDs via a Trigger-3 (T3) request message requesting data about the correlated timestamps. The SDs reply with T3 reply message with the requested data, which is set to

campus to be studied. The design imposes a six second hard real-time deadline on T2 messages going to campus, and a 60 second soft real-time deadline on T3 reply. If the T2 messages are not delivered within the allotted timeframe, the SD will suffer from buffer overrun and T2 messages will be lost. This thesis focuses on T2 message reliability as it is the most critical due to the hard real-time deadline imposed on the message.

## **1.1 Auger South and North**

Since its inception, it had always been envisioned that the Auger Observatory would have two sites, one each in the southern and northern hemispheres.

### **1.1.1 Auger South**

The southern site, *Auger South*, is located in Mendoza province, Argentina [1]. Completed in 2008, the Auger South SD comprises of 1660 water Cherenkov stations arrayed on a 1.5 km triangular grid, covering a surface area of 3,000 km<sup>2</sup>. In Auger South, each station communicates directly with one of four tower-mounted *Concentrator* stations located on the periphery of the array. From there, data is forwarded via commercial microwave links to a fifth tower at the observatory campus. The necessary timing precision is provided by a GPS receiver on each SD station.

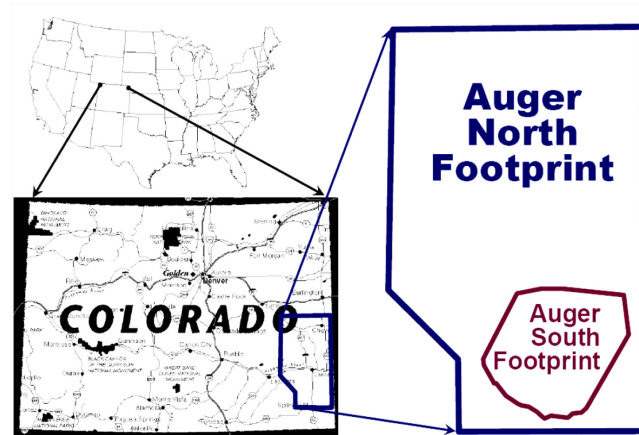
### **1.1.2 Auger North**

The design for *Auger North* builds upon the same detector concepts and technology that have succeeded in Auger South. To achieve the desired improvement in sensitivity above 60 EeV, Auger North needs nearly three times as many SD stations covering seven times the surface area as Auger South. Plans call for 4,000 stations to be arrayed on a regular square grid with  $\sqrt{2}$  mile (2.3 km) spacing, and another 400 stations to fill in a dense “infill” sub-array with one mile (1.6 km) spacing.

The original design calls for locating Auger North in the state of Colorado, USA, as shown in Figure 1. The footprint of the Auger South array is overlaid onto that of Auger North to illustrate the magnitude of the size difference between the two arrays. The differences in geographical area and number of SD stations, combined with differences in topography and available infrastructure have rendered the direct single-hop station-to-

tower routing of Auger South impractical in Auger North. Thus, a new wireless networking paradigm has been adapted to the requirements and constraints of Auger North.

This thesis evaluates the performance and reliability of the wireless networking paradigm, as it applies to Auger North.



**Figure 1: Location of Auger North and its size relative to Auger South**

## **1.2 Auger North Communication Challenge**

The Auger South wireless network was successfully designed and implemented based on single-hop station-to-tower routing. However, for several reasons, the Auger South tank-to-tower system was infeasible, so the change was made to use local *peer-to-peer* (P2P) communication [3,22]. In a P2P-WSN, the transfer of data between a station and the Concentrator is accomplished via multi-hop relaying of data between neighboring SD stations.

The greatest disadvantage to P2P routing is reliability. In a P2P system, the failure of one radio could disconnect all stations upstream of that point from the Concentrator. Thus, redundant routing and fault-tolerance must be incorporated into the WSN. The particular WSN paradigm chosen to satisfy the Auger North requirements is the *Wireless Architecture for Hard Real-Time Embedded Networks* (WAHREN). Although WAHREN was initially created for applications with linear topologies [4], the process of adapting it

to the two-dimensional Auger SD array has yielded several enhancements to its capabilities [3,5,6].

Due to the rare event of observing an ultra-high energy cosmic ray, and the real-time deadlines imposed on message delivery, it is important to ensure message delivery. The different physical processes that contribute to message loss imply different mathematical models. These processes and the sub-models used to evaluate them are described below.

- **The availability of a tank** (a station in SD) in one sector, for a given time period, is over a continuous time interval, therefore analyzed as a Continuous Time Markov Chain (CTMC).
- **The probability that a given fault would be in a particular tank's path** to the CR is dependent on its position in the sector, making it necessary to do an exhaustive search for each tank position, and each fault position in a sector.
- **Mapping Bit Error Rate (BER) to Message Error Rate (MER) is a binomial process.** Bit errors are random and independent; therefore MER could be mapped from BER using simple binomial distribution.
- **The probability a message is lost is dependent on the receiver's BER.** A message lost due to BER between tanks is a discrete event; therefore a Discrete-Time Markov Chain (DTMC) was used to determine the probability a message is lost.

These sub-modeled probabilities were combined to determine tank, sector, and system reliability. Reliabilities that are not being analyzed are systemic (global change) failure (two consecutive tanks in a chain), CS failure, CR failure, and source node failure. To ensure that a *Babbling Transmitter* failure mode cannot be caused (creating a single point of failure), the tank's radio incorporates a channel guardian which independently monitors the transmitter output and system timing. If the channel guardian determines that the radio is transmitting outside of its assigned time slot, it disables the transmitter. The Guardian thus biases the behavior of a faulty radio towards the *Fail Silent* mode.



This analysis is of *Point Failure*, not *Systemic Failure*. *Point Failure* of the communication system is defined as random independent failures of scattered single tanks in the power chain. *Systemic Failure* is defined as the interruption of the peer-to-peer relay chain resulting in the loss of data from several stations for an extended period of time. Systemic failure and formal verification of the correctness of the WAHREN protocol has been previously analyzed [7], therefore was not studied in this analysis.

## 2 Background

As its name implies, WAHREN provides the following highly desirable properties [3,22]:

WA = Wireless Architecture: Unlike an ad hoc WSN, WAHREN provides a base infrastructure of fixed nodes suitable for fixed installations such as the Auger North SD array. These nodes are referred to as Infrastructure Nodes (INs). However, WAHREN also supports a limited number of mobile, Non-Infrastructure Nodes (NINs), which can communicate with the INs, on a not-to-interfere basis.

HR = Hard Real-time: The vast majority of real-time WSN paradigms address only soft real-time criteria, under which an occasional missed deadline is not a serious matter. However, hard real-time systems, such as Auger North, require that delivery delays for all messages be time-bounded and determined a priori for all survivable fault scenarios.

EN = Embedded Networks: Embedded systems such as Auger North often require high levels of reliability. Thus, WAHREN employs fault-tolerant topologies and protocols to ensure continued communication in the presence of faults. Once any message leaves its originating node, WAHREN transparently provides redundant message routing.

The WAHREN paradigm will now be introduced in terms of its three primary characteristics: the IN network topology, the Medium Access Control (MAC) protocol, and the P2P message scheduling protocol.

### 2.1 WAHREN Topology and Physical Layouts

WAHREN is based on a well-understood family of graph topologies known as *Power Graphs of a Chain* [8]. In graph theory, a *Graph* (G) comprises a set of *Vertices* (V), connected by a set of *Edges* (E). This definition is formally expressed by the notation  $G = \{V, E\}$ . In graph theoretical terms, a *chain* is a series of vertices (nodes), each of which is connected to *only* its two nearest-neighbor vertices via two edges (links). This is illustrated by *solid* lines connecting adjacent nodes in Figure 2. Dotted lines indicate second-order links.



**Figure 2: Graph topology of a 2<sup>nd</sup>-order power chain**

Although a chain topology is a natural fit for a P2P WSN with limited radio range, it is clearly *not* fault-tolerant, since the failure of even a single node breaks the chain.

However, any topology suitable for WAHREN must provide some degree of built-in redundancy. To this end, we now formally define a Power Graph. Specifically:

*Given any graph,  $G^1 \equiv G = \{V, E\}$ : its  $p^{\text{th}}$ -order power graph  $G^p = \{V, E^p\}$  is obtained by adding an edge between every pair of vertices located at a skip-distance (number of hops) of  $d \leq p$  in the original graph  $G^1$ .*

For brevity, we will refer to the power graphs of chains and rings as *power chains* and *power rings*, respectively.

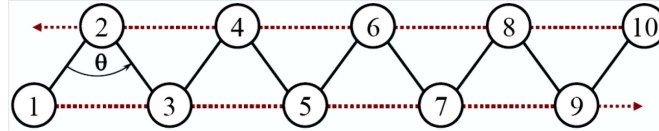
### 2.1.1 Second-Order Power Chains and Rings

Figure 2 shows the graph topology for a second-order power chain. The *dotted* lines are the second-order links between what were originally second-nearest neighbors. In this topology, communication is inherently redundant because each message is multicast to both its first-order and second-order neighbors, all of whom can then forward that message.

The difference between a chain and a ring is whether the end nodes wrap around to communicate with each other. For example, in Figure 2, if nodes 1 and 10 wrapped around to be nearest neighbors, then the graph would be a power ring instead of a power chain. Second-order power rings have appeared under a variety of names, most commonly as braided rings. Because of its inherent redundancy, a wired version of the *braided ring* has recently been proposed for safety-critical aviation applications [9].

It is important to note that a given *graph* topology can be realized in a variety of physical layouts. For example, Figure 3 shows one *physical* realization of a second-order power chain. As angle  $\theta$  becomes smaller (more acute), second-order neighbors move closer

together. Once they are within radio range of each other, the second-order links (dotted lines) are established. Thus, the physical layout of Figure 3 is functionally identical to the graph topology shown in Figure 2.



**Figure 3: Physical layout of a wireless 2<sup>nd</sup>-order power chain, as used in Auger North**

### 2.1.2 Higher-Order Chains and Rings

The Auger North WSN uses the layout of Figure 3 to implement a second-order power chain topology. However, it is worth noting that this is just a special case of a generic  $p^{\text{th}}$ -order power chain. Higher-order power chains and rings afford greater redundancy as the order  $p$  increases. As with the second-order graphs, the *physical* locations of the nodes in higher-order graphs need not correspond precisely to their *topological* locations.

Considerable leeway can be afforded in positioning the nodes, as long as lines-of-sight and maximum ranges to neighboring nodes are not violated. In addition, the axis of the graph may curve, as long as the radius of the curvature is not severe enough to disrupt the connectivity of the outer nodes. Thus, power chains of any order can be embedded into any physical infrastructure that even *approximates* a linear, tubular, sinuating, or toroidal shape.

### 2.1.3 Non-Linear Physical Layouts

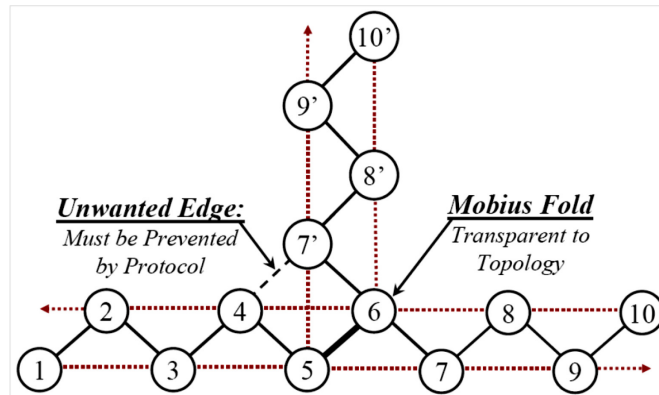
While the preceding physical layouts allow gradual changes in direction, they do not allow for sudden changes, such as a 90 degree turn. However, this kind of turn is exactly what one would expect in a building corridor, roadway, and other applications.

Fortunately, power chains can turn a sharp corner in a manner that is *transparent to the graph topology*. For example, Figure 4 shows a second-order power chain that splits off into two alternative paths (nodes 7-10 and nodes 7'-10').

The vertical chain (with primed node numbers) forms a sharp corner with respect to the original chain. At the point where this chain splits off from the main chain, the node numbers reverse their left-right orientation, but otherwise the sequence remains intact. We call this structure a *Mobius Fold* after its similarity to the half-fold in a Mobius strip.

The existence of a Mobius Fold is transparent to the graph topology. For example, node 5 connects to nodes 6 and 7 in the main chain, and to 6' and 7' in the side chain. Similarly, node 6 connects to nodes 7 and 8 in the main chain and to nodes 7' and 8' in the side chain. The difference is indistinguishable. Higher-order analogues of the second-order Mobius Fold exist for higher-order power chains.

One final observation from Figure 4 is that the fold reduces the distances between some nodes. For example, nodes 4 and 7' are close enough together to establish a radio link. It is up to the message scheduling protocol to ensure that this link is not actually utilized.



**Figure 4: Mobius Fold in a 2<sup>nd</sup>-order power chain**

## 2.2 WAHREN Medium Access Control (MAC) Protocol

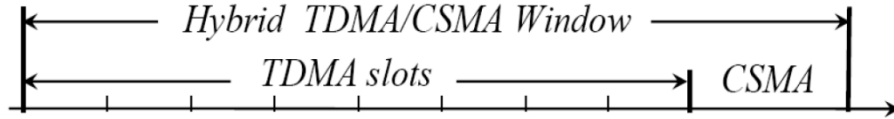
Many wireless standards use contention-based MAC protocols such as Carrier Sense Multiple Access (CSMA). In CSMA, any node wishing to transmit first listens to the medium; if no other node is transmitting, then that node may begin transmission.

However, a collision occurs if two nodes start transmitting at the same time. In a wireless system, the signal strength of a node's own transmitter will overpower any incoming signals. Thus, a collision will not be recognized until the recipient fails to acknowledge

the transmission. Then, one or both of the transmitters must wait for a randomly determined period of time and retry its transmission. This process is referred to as *backoff and retry*.

While CSMA is popular in general purpose protocols, its collisions, backoffs, and retries are inconsistent with predictable, time-bounded delays. However, a Time Division Multiple Access (TDMA) MAC protocol resolves these issues because each node is assigned a time slot in which it is the *only* node allowed to transmit. Hence, delays are both bounded and predictable. For greater bandwidth utilization, transmitting nodes outside of mutual interference range from each other can share a given time slot; this is referred to as *Spatial Reuse TDMA* (SR-TDMA). A SR-TDMA protocol groups a fixed number of time slots ( $S$ ) into a TDMA *window*. Nodes are then assigned to transmission slots in a round-robin fashion. For example, assigning node  $k$  to transmit only in slot  $(k \bmod S)$  assures that any two nodes transmitting in the same slot are separated by at least  $(S-1)$  intervening nodes.

WAHREN requires that all INs follow a strict SR-TDMA MAC Protocol. However, it does allow potentially mobile NINs to communicate with the network, as long as they do not interfere with IN communication. This is accomplished by appending an additional CSMA slot to the window, as shown in Figure 5. During the CSMA slot, NINs may contend with each other for the right to transmit without interfering with IN transmissions in the TDMA slots. This approach is similar to hybrid modes in FlexRay [10], TT-CAN [11], and the Guaranteed Time Slot (GTS) mode of IEEE 802.15.4 [12]. Unlike WAHREN, FlexRay and TT-CAN employ a wired, global broadcast bus instead of short range peer-to-peer links. Thus, they cannot employ spatial reuse of time slots. By contrast, the IEEE 802.15.4 wireless protocol can employ spatial reuse. However, the GTS mode requires dedicated nodes to serve as “coordinators” for local groups of nodes. These coordinators constitute single points of failure for the group. However, in WAHREN, no such coordinators are needed; all nodes are true peers, and thus equally expendable.

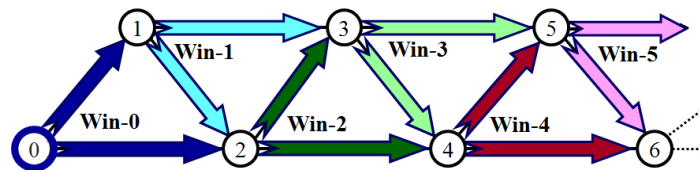


**Figure 5: Hybrid TDMA/CSMA window**

In order to define the boundaries between slots, TDMA requires a common time base across nodes. Each SD station will be *externally* synchronized to GPS time pulses provided at one-second intervals. However, WAHREN also uses a fault-tolerant *internal* clock synchronization protocol, in which nodes timestamp messages received from their neighbors and adjust their own clocks accordingly. This is done for two reasons: 1) to maintain tighter synchronization between GPS pulses, 2) to allow a station radio to stay synchronized and operational in the event that its GPS receiver fails. The WAHREN synchronization protocol employs a variant of the *Fault-Tolerant Midpoint* voting algorithm, which provides the maximum possible level of fault-tolerance [13].

### 2.3 WAHREN Scheduling Protocol

The MAC protocol defines which *slot* (time slot) each node is allowed to transmit in any given window. By contrast, a message scheduling protocol defines which *window* a node should use to transmit a given message. WAHREN employs a *Systolic Broadcast* scheduling protocol<sup>1</sup>, reminiscent of Slotted Ring and Buffered Insertion Ring protocols, e.g. [14]. Figure 6 illustrates this protocol for a single originating node.



**Figure 6: Single-Source Systolic Broadcast**

In this figure, node 0 transmits its own message in window 0. Then, in each subsequent window,  $k$ , node  $k$  forwards the message received during the previous window to its two

<sup>1</sup> In computer science, the term *systolic* refers to an array of cells (or nodes) in which data is synchronously passed from cell to cell at regular intervals, analogous to the systolic pulses in a cardiovascular system.

neighbors ( $k+1$  and  $k+2$ ). Except for the initial transmission from node 0 to node 1, all nodes receive two copies of the message from two different sources.

For example, by the end of window 2, node 3 should have received one copy from node 1 (in window 1) and one copy from node 2 (in window 2). If one or both copies were indeed received and validated, then node 3 selects one copy to forward to nodes 4 and 5 (in window 3). Thus, in a second-order power chain, this protocol automatically provides 2-way node redundancy, 2-way path redundancy, and 2-way time redundancy for each message. Thus, if any node fails to receive or forward a message, an alternate route already exists.

Examination of the window utilization in Figure 6 reveals that, for a single data source, only one node actually transmits in each window. Specifically, node  $k$  transmits *only* in window  $k$ . This observation makes it possible for every node in the system to simultaneously transmit its own original message (each within its assigned slot of window 0). Then, in each subsequent window, all messages are forwarded at the rate of one node per window.

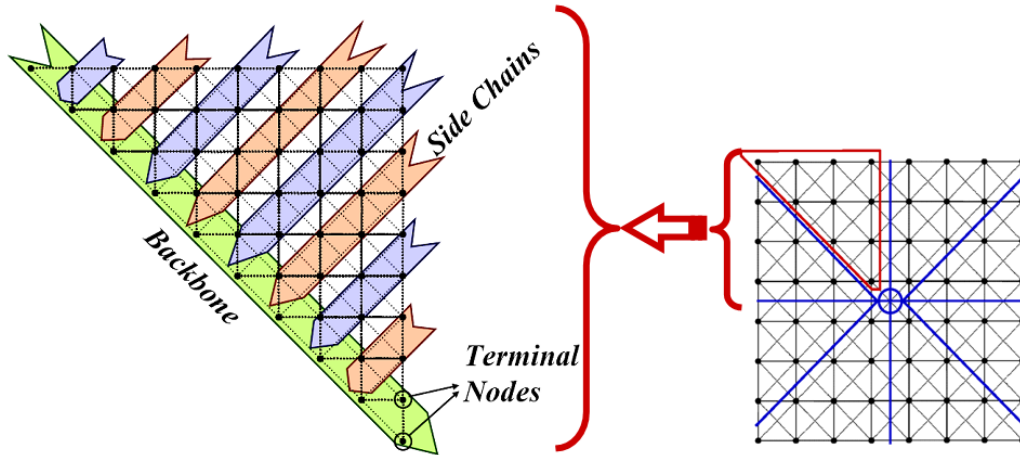
## **2.4 Adaptation of WAHREN to Auger North**

### **2.4.1 Auger North Topology and Physical Layouts**

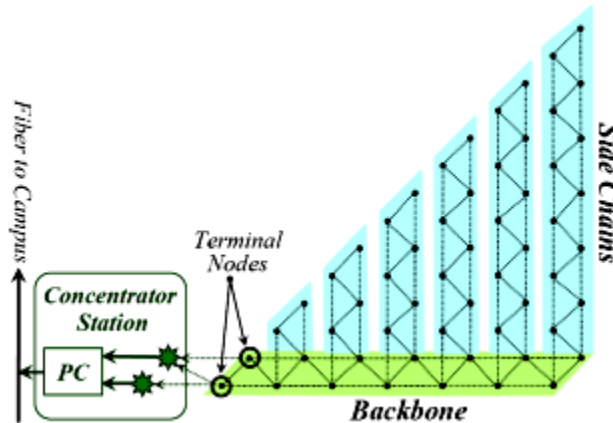
The goal of WAHREN in Auger North is to route data from each of 4,400 SD stations to a few Concentrators. However, a power chain is a linear topology, while the SD is a square array. It is thus necessary to map 1-Dimensional power chains onto the 2-Dimensional array. This mapping is based on two observations: (1) the  $\sqrt{2}$  mile array places stations at alternating corners of the 1 mile square road grid that dominates this region of Colorado. This yields a square array with main axes that are rotated 45 degrees from the four cardinal compass directions; (2) any two adjacent rows or columns running parallel to the cardinal compass directions naturally form the power chain layout of Figure 3, with angle  $\theta = 90$  degrees. Figure 7 illustrates a square service area served by one CS (represented by the circle in the center of the square) with one triangular sector



partitioned. Figure 8 illustrates the embedded second-order power chain in one triangular sector of the array.



**Figure 7: Concentrator Station service area with one sector partitioned**



**Figure 8: Embedding of 2<sup>nd</sup>-order power chains into one “sector” of the Auger North SD Array**

In Figure 8, a *Backbone* power chain runs east from the Concentrator. Power chains running north from the backbone form *Side Chains*, which intersect the backbone via Mobius Folds. Each side chain is activated in-turn to relay data to the backbone, which then forwards all messages to the two Terminal Nodes in the southwest corner of the

sector. The *Terminal Nodes* communicate with two Concentrator radios, which mimic the power chain topology of the SD stations.

The “ideal” sector in Figure 8 is a triangle. Eight such sectors can be served by a single Concentrator. Those regions of the site with rougher terrain may not always afford line-of-sight in the directions required by an ideal sector. Fortunately, a power chain can transparently accommodate both gentle curves and sharp corners. Thus, WAHREN supports much more complex routing paths, including multiple backbones, meandering backbones and side chains, and additional layers of chains. Furthermore, chains can branch off from a parent chain in both the positive and negative directions. All this flexibility makes it possible to embed a logical sector of the array into areas with amorphous shapes, and to route data around holes in the array and obstacles to radio propagation.

One of the advantages of TDMA is that system throughput can be calculated *a priori*. The current design can satisfy up to 128 stations per sector. Thus, the entire 4400 station array comprises up to 35 sectors. Since each Concentrator can serve up to 8 sectors, as few as 5 Concentrators are required, as long as they are more or less centrally located.

#### **2.4.2 Auger North MAC Protocol**

The Auger North window comprises 8 TDMA slots and 1 CSMA slot. Each TDMA slot is further partitioned into two “Minislots”, one for *upload* data being sent from the stations to the Concentrator, and the other for *download* data being sent from the Concentrator to one or more stations.

Interference range calculations show that 8 slots are not quite sufficient to avoid the possibility of “mutual” interference between peer nodes, especially near a Mobius Fold. However, by including a pre-scheduled rotation over 2 radio channels, the distance between potentially interfering nodes was doubled to 16. This proved to be sufficient to avoid mutual interference *within* a sector. In Auger North, each node will actually rotate through 4 channels in order to also avoid interference *between* sectors, especially in the immediate vicinity of the Concentrator.

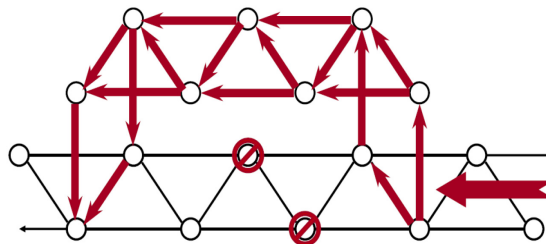
### 2.4.3 Auger North Scheduling Protocol

Data uploads from the stations to the Concentrator require data to be drained from all side chains, into the backbone, and thence to the Concentrator. The upload schedule activates only one side chain at a time. When all messages originating in one side chain have reached the backbone, then the next side chain is activated. This is done for two reasons: (1) to avoid mutual interference between nearby side chains, and (2) to prevent congestion in the backbone caused by parallel message receipts from multiple side chains. This scheduling protocol delivers one message per window to the Concentrator with no pauses or “hiccups” in the flow. Thus, the number of windows needed to transfer all original messages out of the sector equals the number of stations in the sector.

Data downloads are more sporadic than uploads. Thus, a streaming approach is employed. Concentrator radios initiate download transmissions in 2 adjacent windows, then sit idle for 6 windows. This 2-of-8 schedule provides sufficient message separation to prevent mutual interference between adjacent side chains as the messages migrate into the sector.

### 2.4.4 Common Mode Faults

In addition to independent station faults, further analysis was necessary to identify potential common mode failures caused by external events such as tornadoes and blizzards. In the absence of actual on-site data, the correlated double-fault rate was estimated at 1 event/year. The response to common mode failures was to introduce *Dynamic Rerouting*, as shown in Figure 9.



**Figure 9: Local Dynamic Rerouting around two adjacent faulty nodes.**

In Figure 9, the failure of two adjacent nodes causes a blockage in the lower power chain. Using only local diagnostic data, the neighboring nodes then reconfigure via four Mobius Folds to reroute messages around the blockage.

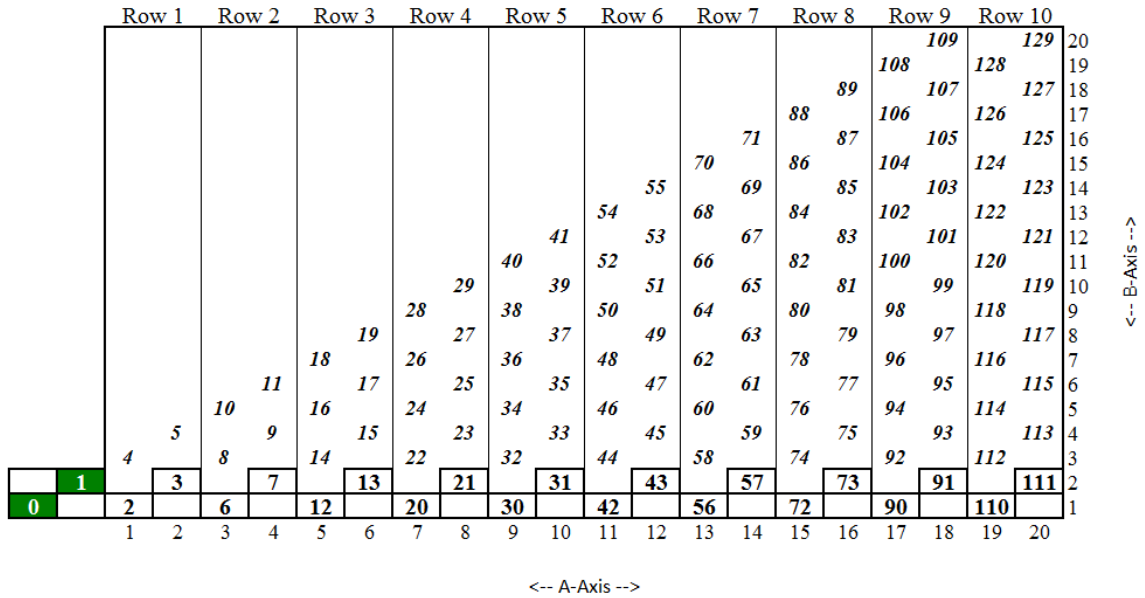
### 3 Multimodal Approach and Methodology

The objective of this thesis is to develop a multimodal modeling methodology to determine the probability of non-delivery of a message, or equivalently, fraction of messages not delivered over the lifetime of the system. Several individual processes contribute to the probability that a message is lost in a P2P-WSN. A multimodal approach is necessary because these processes are driven by different mathematical models.

#### 3.1 Auger North Canonical Sector

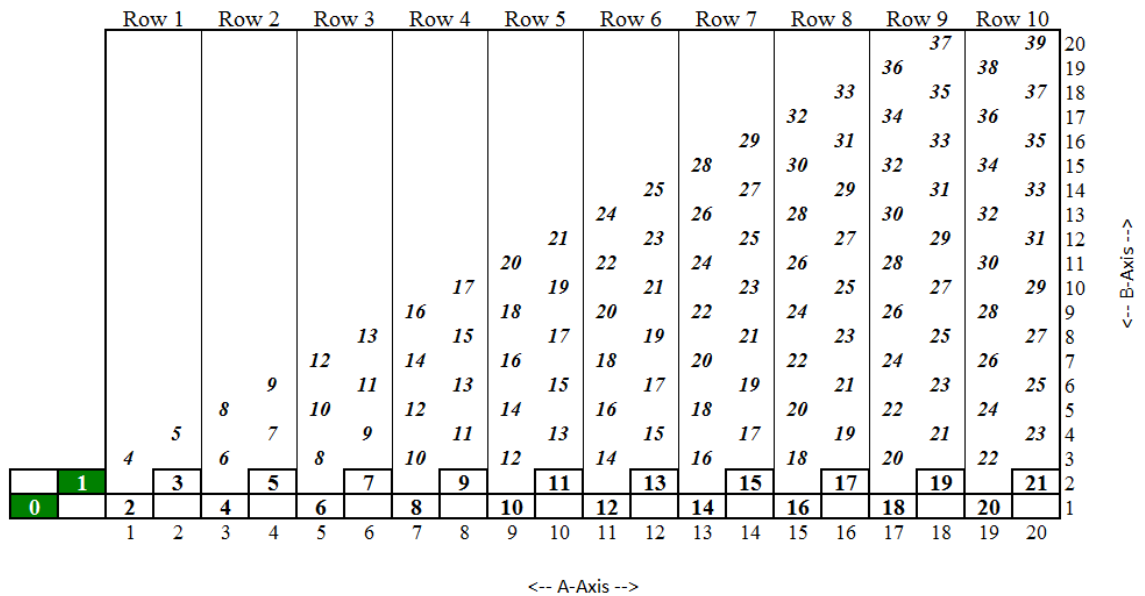
Auger North plans call for 4,400 local stations (also referred to as *Surface Detectors (SD)* or *tanks*), split into 35 *sectors*. Within each sector, all tanks route their sensor data to a single Concentrator Station (CS), which contains two redundant concentrator radios (CRs). A *Canonical Sector* is defined as a 128 station sector, consisting of 20 backbone stations, and 108 stations arranged in Sidechains (i.e. rows). In the canonical layout, the number of stations per row is incremented by two stations per row, moving away from the concentrator station, as illustrated in Figure 10. In this figure, the radios labeled “0” and “1” in the bottom left corner represent the two CRs. Each remaining number represents a local station, with the number being that station’s identifier (ID) number. The bold, boxed stations at the bottom of the sector are located on the “A-Axis”, or backbone. For Sidechain stations, the “B-Axis” aids in determining the number of stations in a row.

In the actual layout, a sector may not have exactly 128 tanks, and/or the rows may not be perfectly incremented by two stations per row, due to topographical constraints (e.g., large changes in elevation may affect P2P communication). For simplicity, a representative *canonical* sector was defined and used for this analysis.



**Figure 10: Canonical Sector Tank ID Layout**

In a sector, a tank's depth is defined as one plus the number of hops (or transmissions) a message must make between tanks to reach a CR. The number at each station position in Figure 11 represents that station's depth in a canonical sector layout.



**Figure 11: Canonical Sector Tank Depth Layout**

### 3.2 Architecture of the Multimodal Model

The following list describes the cause-and-effect of how these processes are integrated into sub-models, and how the sub-models correlate and contribute to the end result of a message being lost.

1. Message Error Rate (MER) is defined as the probability that a message is corrupted or lost over one transmission “hop”. MER is derived from the Bit Error Rate (BER) listed for the receiver. Bit errors are generally assumed to be independent and identically distributed (IID) [15]. Given these assumptions, BER maps to MER via a simple binomial distribution.
2. The MER maps to the losses along a chain via a Discrete Time Markov Chain (DTMC). This mapping is appropriate because the probability of a message being lost during a given transmission (or hop) is independent and identically distributed from that of all other hops. The exception to this rule is in the vicinity of a faulty tank. Because a faulty tank reduces redundancy within the chain and alters the values of the entries in the DTMC state transition matrix. Thus, it was necessary to employ a piecewise-constant DTMC model.
3. The number of faulty tanks in a path is defined by a Continuous Time Markov Chain (CTMC), modeled with a Generalized Stochastic Petri Net (GSPN).
4. The probability that a given faulty tank is in the path from a given source tank to the end of a chain (CR) needed an exhaustive combinatorial search over the entire canonical sector.

The Master Equation will now be introduced which combines results from each of the “sub-models” listed above to produce  $Q(T)$ , the unreliability of a message originating at tank  $T$ . In other words,  $Q(T)$  is the probability of message originating at tank id  $T$  is lost within a sector of tanks and does not reach ether CR. Equation 1 is a *simplified* Master Equation,

$$Q(T) = \sum_{i=0}^{f_{max}} (q_i(d(T)) * C_i(T)) \quad (1)$$

where:

- $Q(T)$  = the probability a message at tank  $T$  does not reach either CR.
- $T$  = the identity of the message *source* tank in a given sector.
- $i$  = the number of faulty tanks in the path from source tank  $T$  to the CR.
- $f_{max}$  = the maximum expected number of faults per sector.
- $C_i(T)$  = the probability that there are  $i$  faults are in tank  $T$ 's path.
- $d(T)$  = the source tank  $T$ 's depth. Where depth is defined as the number of tanks in the path from source tank  $T$  to the CR.
- $q_i(d(T))$  = the unreliability of a message given  $i$  faults in the path to the CR.

Equation 2 is the *complete* Master Equation with  $C_i(T)$  expanded to include the GSPN and Exhaustive Search sub-models. The variables of this equation are listed below.

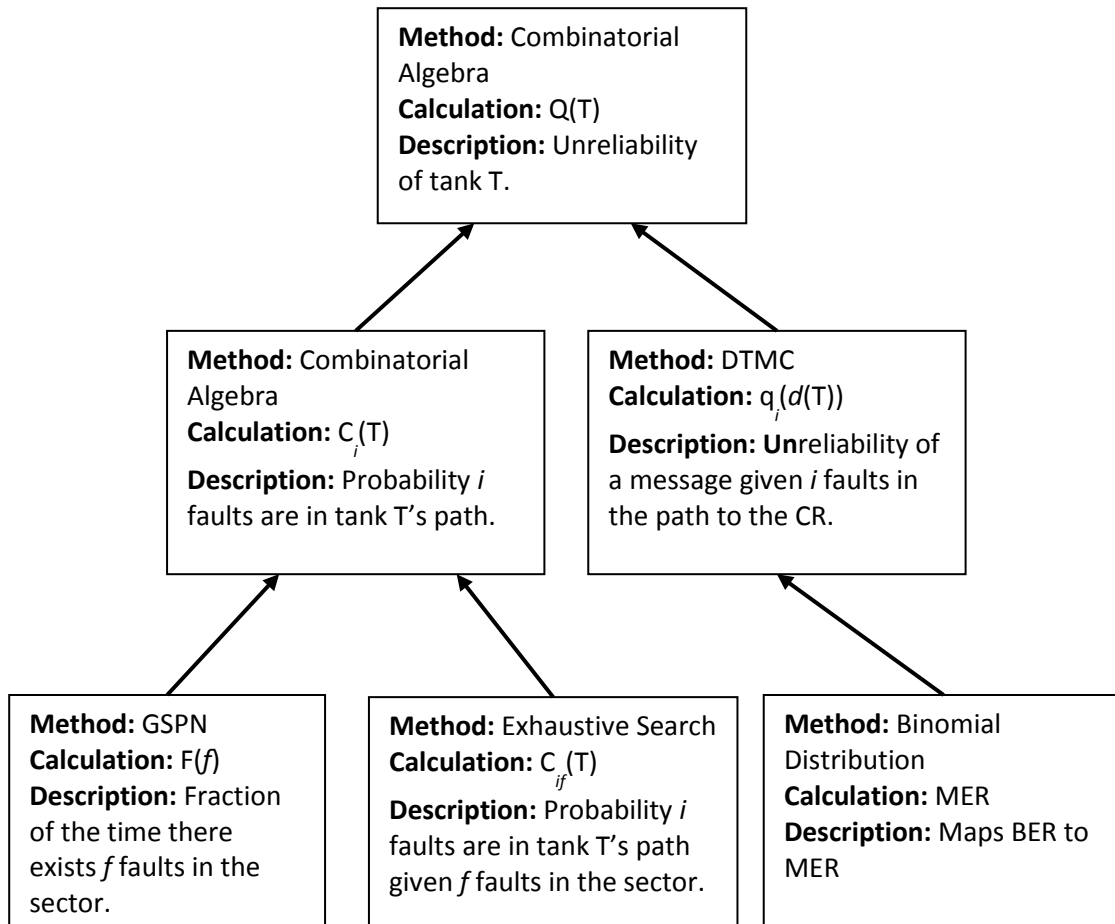
$$Q(T) = \sum_{i=0}^{f_{max}} (q_i(d(T)) * (\sum_{f=0}^{f_{max}} C_{if}(T) * F(f)) \quad (2)$$

where:

- $f$  = the number of faulty tanks in the sector.
- Let  $F(f)$  be the fraction of the time in which there exist  $f$  faults in the sector.
- Let  $C_{if}(T)$  be the probability that there are  $i$  faults are in tank  $T$ 's path, given  $f$  faults are in the sector.

Figure 12 illustrates how the different sub-models integrate to produce the desired end result of message loss for a given tank,  $T$ .





**Figure 12: Architecture of Multimodal Model and Sub-models**

The next sections describe each sub-model. Once these sub-models are defined, the following section walks through one tank's unreliability calculation in a detailed, step by step approach to help illustrate how the sub-model results are combined to determine the end result of one tank's unreliability.

## 4 GSPN Analysis of $f_{max}$ and $F(f)$

### 4.1 Context of $f_{max}$ and $F(f)$

As specified in Equation 2 and illustrated in Figure 12,  $F(f)$  is defined as the fraction of time in which there exists  $f$  faults in a sector. Then, Equation 2 defines  $f_{max}$ , as the maximum expected number of faults in a sector, beyond which the probability of additional faults becomes negligible. The following sections describe Generalized Stochastic Petri Nets (GSPN), and how they were used to calculate the value of  $f_{max}$ , and the values of  $F(f) \forall f \in \{0..f_{max}\}$ .

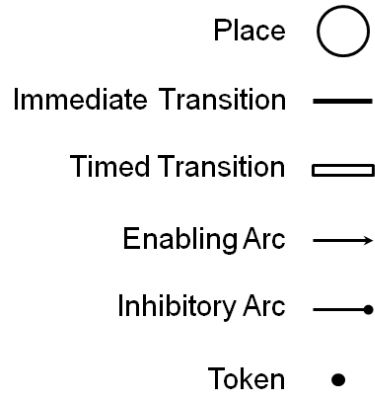
### 4.2 Introduction to GSPNs

Generalized Stochastic Petri Nets (GSPNs) are a common method to model Continuous Time Markov Chains (CTMCs) in reliability analysis. A CTMC is a nondeterministic, state-based stochastic process, in which the next state depends only on the current state (all previous states are irrelevant) [16]. In contrast to the DTMC being evaluated at “ticks” in time, time in a CTMC is a real number, and the time in state is exponentially distributed.

In this analysis, the objective of the GSPN sub-model is to determine unavailability of tank units, which can then be used to determine probability that a given number of tanks is down at the same time. This section starts with an explanation of Petri Nets, and then expands the discussion to GSPNs, and their application to the Auger North analysis.

#### 4.2.1 Petri Net Basics

Petri Nets were created by Carl Adam Petri in 1962 [17]. This thesis gives only a brief introduction to Petri Nets. For more detail the reader is referred to Marsan Chapters 1-5 [16]. Petri Nets have a bipartite graph structure, consisting of places, transitions (immediate and timed), arcs (enabling and inhibiting), and tokens (displays their commonly used symbols).



**Figure 13: Petri Net Symbols**

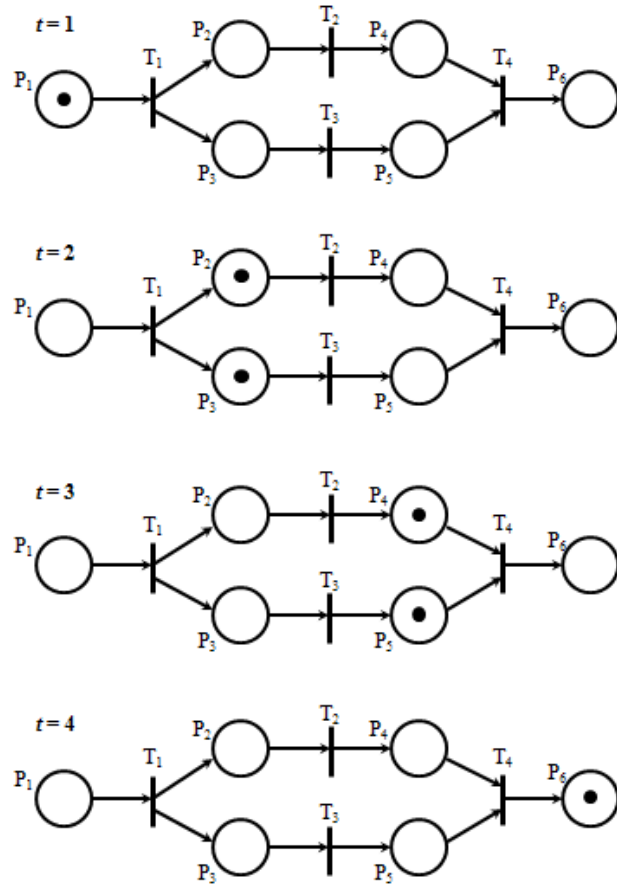
A *place* is a vertex that represents a condition or state variable. For a Boolean state variable, the condition is considered to be true if a token is in its place. An *immediate transition* represents an action which immediately occurs when all conditions needed to enable that action are all met. An *arc* is a directed edge in the graph. An *enabling arc* from a place  $P_i$  to transition  $T_j$  enables the action  $j$  when condition  $i$  is true. An arc from transition  $T_j$  to place  $P_i$  makes condition  $i$  true when action  $j$  occurs.

The *marking* of a Petri Net is the vector of the number of tokens in all places. Since the number of tokens in one place represents the value of one state *variable*, the marking of all places in the net defines the system state.

The term “a transition fires” means that an action took place. A transition fires when every enabling place contains a token (all enabling conditions are true). Firing consumes one token along each input arc, and creates one token along each output arc. In other words, the tokens are “consumed” and “created”, not “moved”. The number of tokens is not always conserved; in other words, the mapping of input tokens to output tokens does not have to be one-to-one.

Figure 14 uses a sequence of transition firings to illustrate the basic transition “firing rules” for a Petri Net. This figure displays a single Petri Net in four different sequential markings (states). The following describes each state in detail:

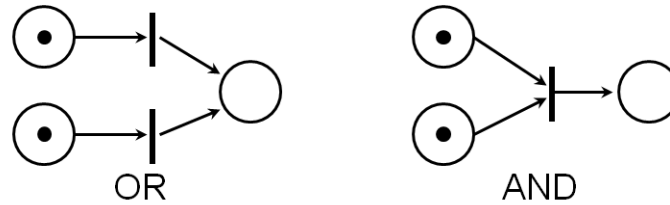
- The initial state of the Petri Net is shown at time  $t=1$ . There is a token in place P1, indicating that condition P1 is true.
- Transition T1 only has one incoming arc from P1. Since there is a token in P1, T1 fires (i.e. the action at T1 is taken). The token is consumed by T1, which produces two tokens. These tokens follow the two outgoing arcs and are placed into P2 and P3, as illustrated at time  $t=2$ .
- At time  $t=2$ , transition T2 is enabled by the token in P2, and T3 is enabled by the token in P3. Thus, T2 and T3 are both enabled. The token in P2 is consumed by T2, which produces a token in P4. Concurrently, the token in P3 is consumed by T3, which produces a token in P5, as illustrated at time  $t=3$ .
- At time  $t=3$ , there are tokens in *both* P4 and P5; at that point, all conditions for the firing of T4 have been met. T4 consumes both tokens, and produces one token which is placed in P6. Thus, at time  $t=4$ , no transitions are enabled, and the Petri Net is now considered “dead”.



**Figure 14: Basic Petri Net Transition Firing**

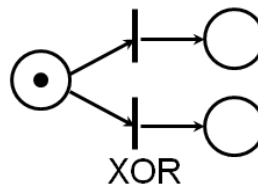
Any combinatorial relationship between events and states can be represented by a Petri Net.

**OR and AND:** Boolean OR and AND relationships can be implemented via simple constructs as illustrated in Figure 15. In the OR construct, the arrival of a token at *either* input place enables its own transition, which then creates a token in the output place (allowing the process to proceed). In the AND construct, *both* input places must contain a token in order to enable the single transition; only the presence of both tokens allows the process to proceed.



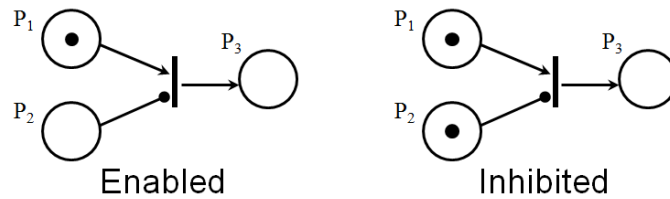
**Figure 15: AND and OR Equivalents in Petri Nets**

**XOR:** A “conflict” is a place with multiple output transitions. If a place has only one token, then one transition will eventually fire. This transaction will consume the token, and disable the other transition (XOR). Firing order is indeterminate (unknown), representing mutually exclusive processes. Different weights can be placed on an outgoing arc to indicate the probability of taking that arc. If no weight is indicated on the different arcs, then the probability of taking each arc is evenly distributed between them. Figure 16 shows an example of a conflict in a Petri Net with equal probability of taking each arc.



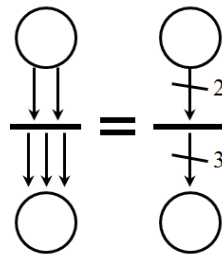
**Figure 16: Petri Net Conflict**

**Inversion:** A standard enabling arc enables a transition if there is a token in its input place. By contrast, an *inhibitory* arc does the opposite of an enabling arc; it *prevents* a transition from firing if its input place contains a token. Specifically, the transition can fire only if the inhibiting condition is *false*. Graphically, an inhibiting arc is identified by a dot instead of an arrow head at its terminal end. Figure 17 shows how a transition can be enabled or inhibited dependent on the tokens in multiple input places. Specifically, the transition is enabled only if P1 is true and P2 is false ( $P_1 \wedge \sim P_2$ ).



**Figure 17: Inhibitory Arcs in Petri Nets**

**Integer Conditions:** A transition may require more than one token in an enabling place in order to be enabled, and may produce more than one token in an output place. The consumption and creation of multiple tokens are denoted with multiple arcs, or with one arc marked with a slash and number. Figure 18 shows a transition that requires two tokens to be enabled, and creates three output tokens when it fires.



**Figure 18: Multiple Arcs in a Petri Net**

#### 4.2.2 Generalized Stochastic Petri Nets (GSPNs)

The *timed transition* acts in a similar manner to the immediate transition *with one exception*. Once the timed transition becomes enabled, the transition waits a specified delay; *then* it fires, consuming and creating tokens in one atomic action. The symbol for a timed transition is shown in Figure 13.

In a *Stochastic Petri Net* (SPN), all timed transitions have an exponentially distributed random firing delay, rather than a constant firing delay. Mean firing rates are represented by lambda ( $\lambda$ ), where  $\lambda = 1/\text{mean time-to-fire}$ .

A Generalized Stochastic Petri Net (GSPN) is identical to a Stochastic Petri Net (SPN) with the exception that it allows *both* instantaneous and exponential transitions in the same net. The instantaneous transitions represent changes of state or logical processes, while timed transitions represent real time-consuming physical processes. GSPNs are isomorphic to CTMCs, and can thus represent any continuous time Markov process.

### 4.2.3 State Space Explosion and Truncation

Dependability and availability modeling using Markovian techniques suffers from state space explosions, often referred to in literature as the “largeness problem” [18]. Most real world systems being modeled produce extremely large CTMCs. One *state* of the CTMC is represented by the positions of *all* tokens in the entire net. Since each place represents one state variable, rather than the entire state, the visual complexity of a GSPN grows as  $O(\text{number of state variables})$  instead of  $O(\text{number of states})$ . Thus, a relatively simple GSPN can represent a system whose CTMC would contain millions of states.

*State Truncation* is the technique used in this analysis to limit the state space of the Auger North CTMC. The state truncation technique eliminates system states that have a low probability of occurring. Due to its importance and relevance to this thesis, excerpts from Mondal’s work describing truncation are provided below [19].

*Typically in highly reliable systems, the occurrence of many failures is of low probability, and the system spends most of its lifetime in a small subset of highly probable states [20][21]. If a system has  $N$  components and all of these components have a very low failure rate (i.e. very long Mean Time to Failure), then the likelihood of reaching the system states with more than a few faulty components is very low. Therefore, it is convenient to stop the state space generation after a small number of faults ( $L$ ) have occurred. In that case, we implicitly assume that the probability of reaching states after  $L$  or more faults is very low and we can neglect those states by truncating the original state space.*



### 4.3 Stochastic Petri Net Package (SPNP)

For this study, all GSPN models were entered into the commercially available *Stochastic Petri Net Package* (SPNP) [16] for analysis and evaluation. SPNP generates an analytical solution to the system differential equations, rather than running a discrete event simulation. The two primary advantages to this approach are: (1) an analytical solution yields exact results, rather than the approximate answers with confidence levels and confidence intervals that would be produced by a simulation, (2) an analytical solution evaluates *all* possible states, whereas discrete event simulations can easily miss highly unlikely (but catastrophic) states. In addition, SPNP has proven itself to be remarkably stable and efficient, even for systems comprising thousands of states.

### 4.4 Modeling of Tank Availability

A detailed continuous time Markov model was created as a GSPN to model one Auger North sector's unavailability, and analyzed using SPNP. A graphical representation of the sector's model is provided in Appendix A, and the C code input to SPNP for this model is available in Appendix B.

The fault rates, repair rates, and other parameters for various components were gleaned from the best available sources, including vendor-supplied data, Auger North specifications, and Auger South experience. The following assumptions and constraints were applied when developing the GSPN for the Auger North Sector:

- All station tank faults are *Fail Silent*, due to the presence of the independent Channel Guardian chip on the tank radio's baseband circuit board.
- The system specification for the number of tanks in a canonical sector is 128 [22].
- The fraction of battery faults and charger faults that give a warning was set to 90%. This was conservative because experience has shown at least 90% of batteries and chargers give a warning. Due to the similarity to the Auger South Observatory, the mean warning time was empirically chosen based on data from the Auger South Observatory site. Specifically:

- A 30 day mean warning time was selected for battery failures, i.e. a failing battery starts to exhibit symptoms about 30 days before complete failure.
- The specification for battery capacity calls for the battery to last 10 days without charging. Hence, the mean charger warning time was set to 10 days.
- Based on data available in the literature, the battery physical fault rate (1/MTTF) was set at  $\lambda = 5.7 \times 10^{-8}/\text{hr}$  [23].
- Based on vendor-supplied data, the Charge Controller's physical fault rate was set at  $\lambda = 4.6 \times 10^{-8}/\text{hr}$  [24].
- There was no data on the solar cell  $\lambda$  [25]. Thus, due to the solar cell's simplicity, it was assumed its failure rate would be no worse than the charge controller.
- Consistent with circuit boards this size and complexity, the system specification for radio electronics physical fault rate was set at  $\lambda = 1.0 \times 10^{-6}/\text{hr}$ .
- Based on experience at Auger South, the Mean Time to Repair (MTTR) was specified to be 4 days [22]. In addition, it is assumed that there is no more than one repair vehicle per sector.
- Correlated faults, or common mode faults, are events (e.g. tornado, blizzard), that would take down more than one tank simultaneously. Annual correlated faults are estimated to be 1 per sector per year.
- To deal with correlated faults, Dynamic Re-routing was enabled, as described in Subsection 2.4.4.
- A truncation limit of 8 proved sufficient to limit the number of states, without losing any accuracy of the data.

## 4.5 Numerical Results

Using the base conditions and assumptions listed above, the results of the GSPN established unavailability are displayed in

Table 1, which can be described as the fraction of the time there exists  $f$  faults ( $F(f)$ ). The maximum value of  $f$  ( $f_{max}$ ) was set at 4 faults, because the probability of having  $f=5$  or more faults was  $4.0 \times 10^{-9}$ , or 2 orders of magnitude less than the probability for  $f=4$ . Indeed, given  $3.16 \times 10^7$  seconds in one year, it is expected that *more than* 4 faults would exist for much less than 1 second per year. To make the analysis slightly conservative, the value of  $F(f)$  listed in

Table 1 for  $f=4+$ , is the probability of having 4 or more faults active at a given time (Appendix C).

**Table 1: Fault Probabilities per Sector**

$f$	$F(f)$
0	9.8577E-01
1	1.3741E-02
2	4.8157E-04
3	8.3417E-06
4+	1.9908E-07

## 5 Exhaustive Search for $C_{if}(T)$

### 5.1 Context of $C_{if}(T)$

As specified in Equation 2 and illustrated in Figure 12,  $C_{if}(T)$  is defined as the probability that  $i$  faults are in tank  $T$ 's path, given  $f$  faults in a sector. A tank's path is defined as the subset of tanks that messages pass through between that originating tank and the CR.

One counter-intuitive factor is that  $C_{if}(T)$  is not simply a function of tank depth  $d$ .

Multiple tanks can share the same depth, yet have a unique placement in the canonical sector due to being further along the sector's backbone. Thus, an exhaustive search was needed to determine  $C_{if}(T)$  for each unique tank position. The following section gives a detailed explanation for the variation in  $C_{if}(T)$  at a given depth.

#### 5.1.1 Variation in $C_{if}(T)$ at a Given Depth

As discussed in the Subsection 3.1, the flow of messages between tanks follows a Mobius fold as messages are forwarded from the tanks in a side chain to the tanks in the backbone. Because of the Mobius fold's nature of making a tank's location in the chain (side chain, or backbone) transparent to communications, it seems intuitive that regardless of where the tank is located, that it should have the same fault in-path probabilities as another tank at the same depth. This is not the case.

The probability a given fault is in the path of a given source tank,  $C_f$ , has been calculated with a restriction that no two faults can be adjacent tanks in the chain. This is considered invalid as it would cause a systemic failure, which is beyond the scope of this study. The restriction of there being no two adjacent faulty tanks caused an inconsistency in  $C_f$  at a certain depth. LRs in the canonical sector are numbered with Tank IDs from 2 to 128. Stations 0 and 1 are the sector's CRs.

Figure 10 displays the tank ID layout in a canonical sector, and Figure 11 displays the tanks' depths in the sector at each tank location.

A tank may have a different number of neighbors dependent on where it is in the chain, even if it is at the same depth ( $d$ ) as another tank. A tank at depth  $d$  in the middle of a side chain will have two immediate (first-order) neighbors, i.e. at depth  $(d-1)$  in its chain, and one at depth  $(d+1)$  in its chain. Tanks at the end of the chain (leaf tanks) will have only one neighbor due to not having a tank with a higher depth  $(d+1)$  in its chain. There are two backbone tanks at each intersection with a side chain. The outer backbone tank has three neighbors, the tank with one less depth  $(d-1)$  in the backbone, a tank at depth  $(d+1)$  in the backbone, and a tank at depth  $(d+1)$  in the adjacent side chain.

The inconsistency in  $C_f$  at a given depth is caused by a change in legal fault combinations when some, but not all faults are in the chain of a given source tank. In other words, when given  $f$  faults in a sector, and  $i$  are in the source tank's path, but not all faults in the sector are in its path ( $0 < i < f$ ).

To help illustrate the reasoning behind the inconsistency in  $C_f$ , we will examine two tanks at the same depth, tank 11 and tank 15, with two faults in the sector. When given two tanks at the same depth  $d$  ( $d=9$  in this case), one tank will be further down a side chain (tank 11), and the other will be further down the backbone (tank 15). Let one fault be in the tanks path, and focus the legal positions of a second fault. Most tanks have one illegal position for the second fault (the tank one hop further along the chain). If the first fault is at the outer backbone intersection position, it has two illegal positions for the second fault (one tank further along the backbone and one tank further along the side chain).

Therefore, as a tank gets further down the backbone, it gains more tanks in its path that have three neighbors (tank 13 in this example). Taking into account the restriction of neighboring tank faults not being valid fault scenarios, and tank 13 has an additional neighbor, there is one less valid fault scenario for tank 15 when compared to tank 11.

									29
									28
								19	27
							18	17	26
				10	11		16	15	25
			5	8	9		14	13	24
		4	3	6	7		12	20	23
	1	2							22
0									21

Figure 19: Tank  $C_f$  Illustration

## 5.2 Methodology

The probability that a fault is in tank  $T$ 's path,  $C_{if}(T)$ , was determined by exhaustively searching all legal fault locations for each tank within the array. The exhaustive search, or “brute force” approach, was accomplished by creating a program in C language (Appendix D). The program started with an array of tank IDs for one canonical sector. It created an array of Boolean values to indicate which tank IDs were neighbors, and an array of Boolean values for each tank to show which tanks IDs were in its path to the CR. Recording neighboring tanks was necessary to account for the illegal fault scenario of two adjacent faults.

The algorithm to analyze  $C_{if}(T)$  includes nested *for* statements to iterate through each tank position, and each possible fault position in that tank’s path to the CRs. For each additional fault introduced to the sector, an additional inner loop was added to account for each of the new fault positions. Illegal fault scenarios were excluded by checking to see if two faults were adjacent. Probabilities were calculated by counting the number of fault scenarios per each tank (zero faults in path, one fault in path, ect.) and dividing by the total number of fault scenarios for that tank.

## 5.3 Numerical Results

The probability  $C_{if}(T)$  from the exhaustive search was determined for each tank  $T$  from  $T = [2 \dots 129]$ , faults in the sector  $f = [1 \dots 4]$ , and faults in tank  $T$ 's path to the CR  $i = [0 \dots f]$  (Appendix E).

Figure 20 through Figure 24 show  $C_{if}(T)$  for an example set of tank IDs 2, 7, 50, 75, 100, and 129 as faults are introduced to the sector. The general trends of the different tanks and scenarios were as expected. The closer a tank is to the CRs, the less likely a fault is to be in its path (due to there being fewer hops to the CRs).

For example, the Figure 20 shows that, for tank ID 2 the probability that there are zero faults in its path is 1.0. This is due to the fact that tank 2 is adjacent to the CRs, so there are no tanks that could be faulty in the path to the CR. Tank ID 129 has a much higher probability that faults will be in its path because it has a larger depth, meaning it has more hops to get to the CRs. Tank IDs 50 and 75 share similar values for  $C_{if}(T)$  because they are at depths of 19 and 20, respectively.

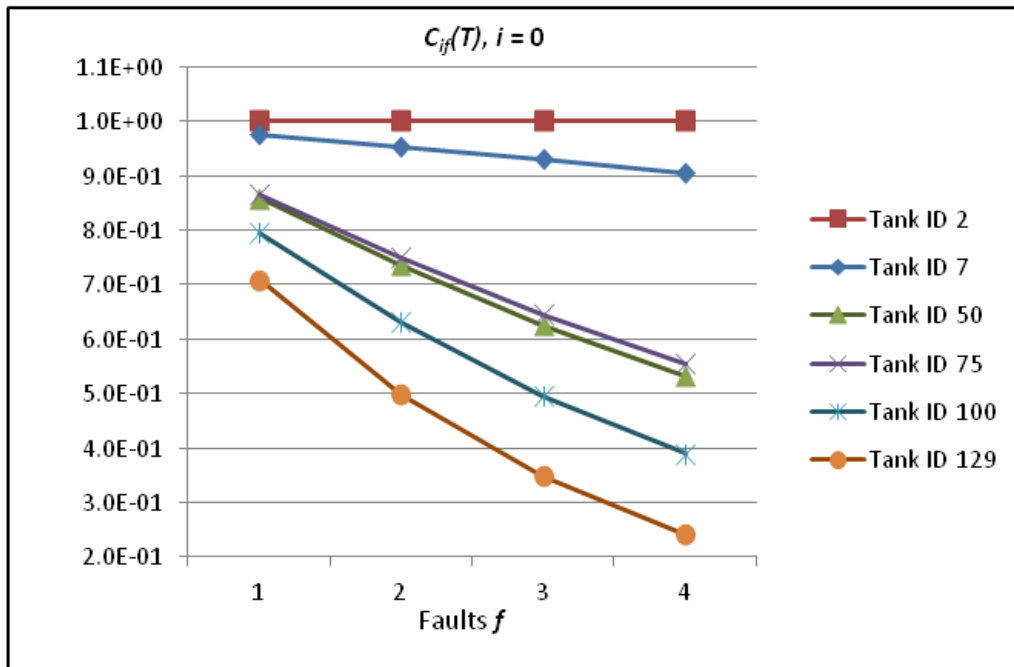


Figure 20:  $C_{if}(T)$  Results,  $i = 0$

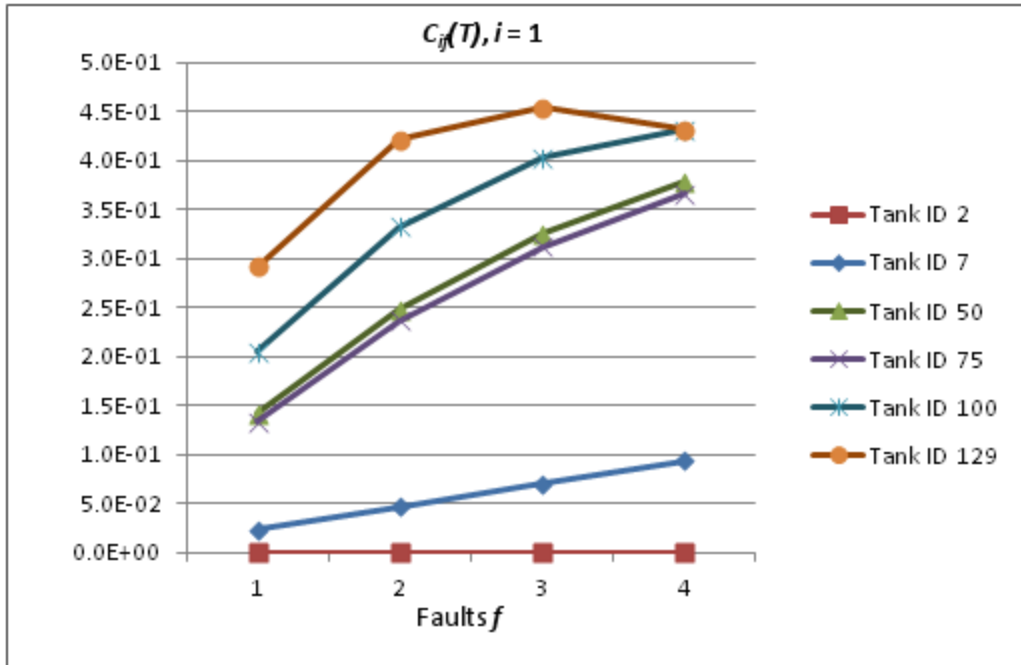


Figure 21:  $C_{ij}(T)$  Results,  $i = 1$

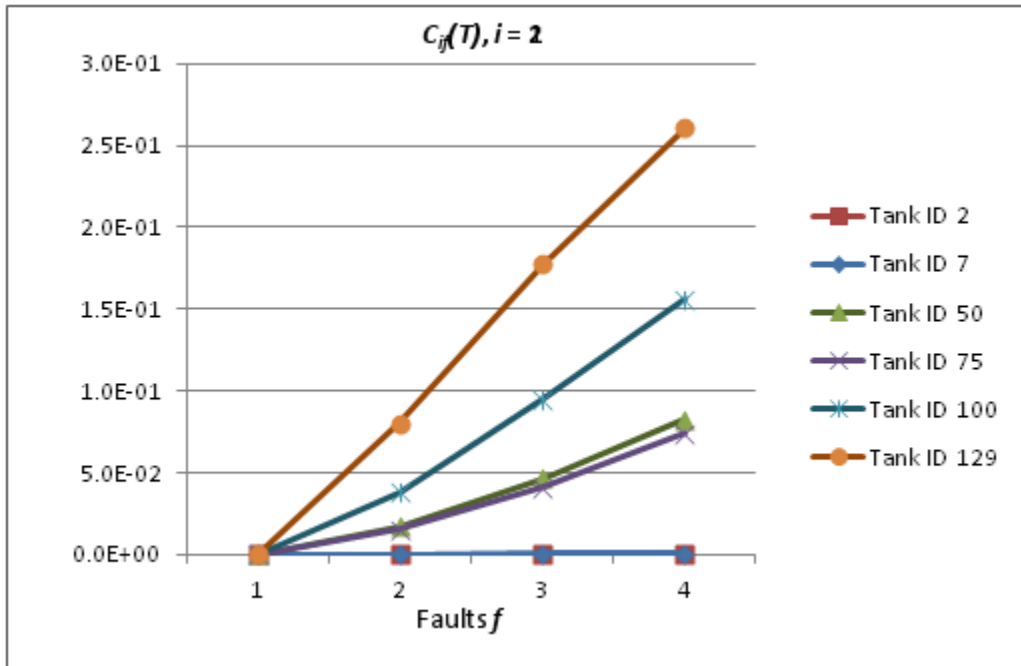


Figure 22:  $C_{ij}(T)$  Results,  $i = 2$



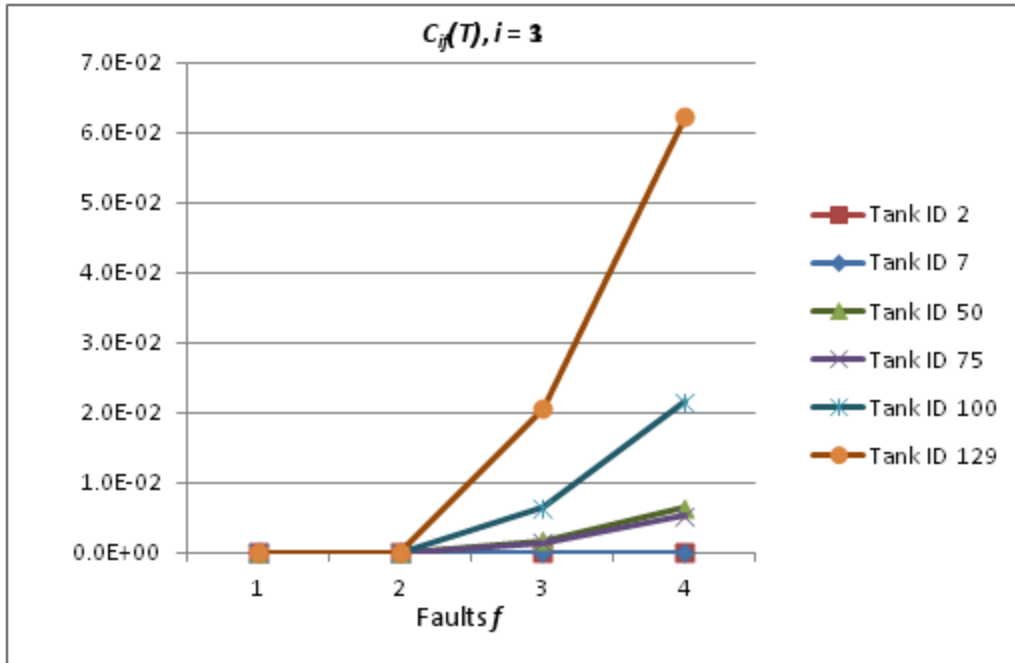


Figure 23:  $C_{ij}(T)$  Results,  $i = 3$

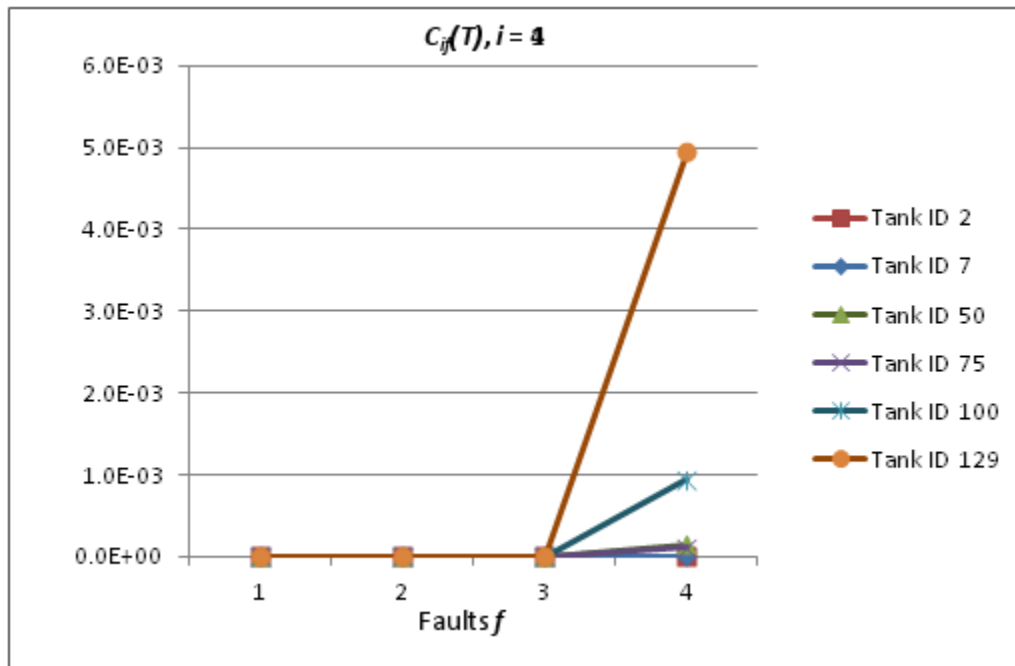


Figure 24:  $C_{ij}(T)$  Results,  $i = 4$

## 6 Combinatorial Analysis of $C_i(T)$

### 6.1 Context of $C_i(T)$

As specified in Equation 2 and illustrated in Figure 12,  $C_i(T)$  is defined as follows. For a given tank  $T$ ,  $C_i(T)$  is the *unconditional* probability there are  $i$  faults in the path from tank  $T$  to the CR.

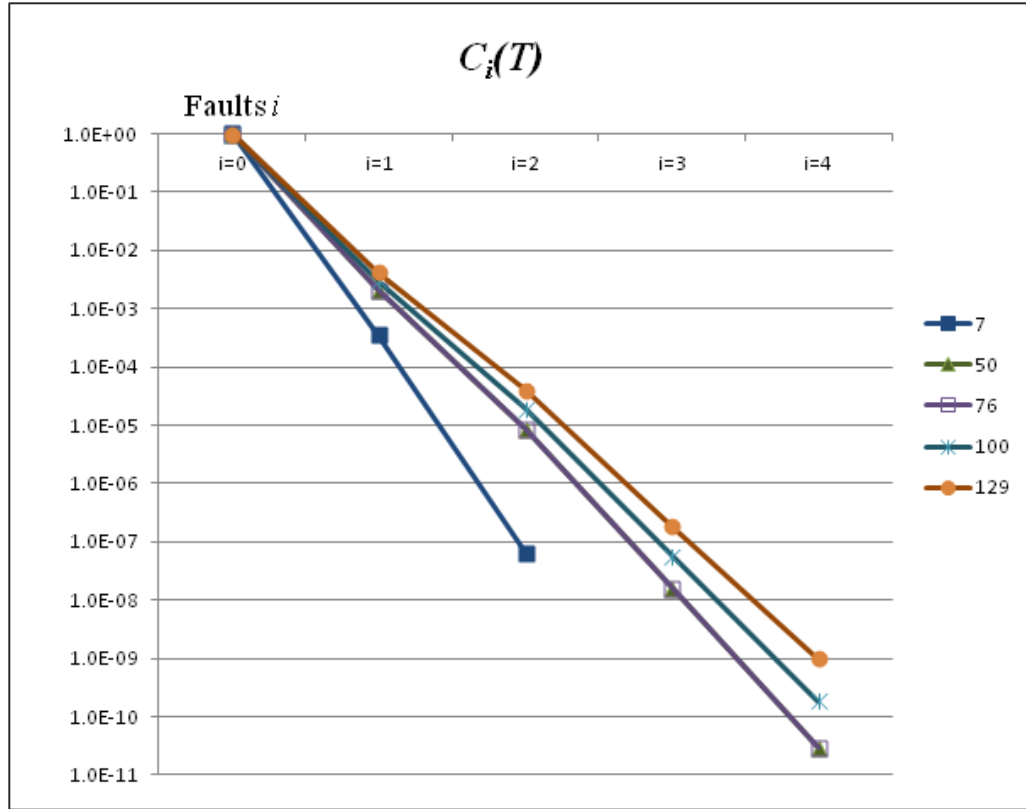
### 6.2 Methodology

The probability of  $f$  faults in the sector ( $F(f)$ ) is found by multiplying the probability of having  $f$  faults in the sector ( $F(f)$  from the GSPN modeling defined in Section 4) with the conditional probabilities that given  $f$  faults,  $i$  would be in tank  $T$ 's path ( $C_{if}(T)$  from the exhaustive search in Section 5). The result,  $C_i(T)$ , is the unconditional probability  $i$  faults are in tank  $T$ 's path.

$$C_i(T) = \sum_{f=0}^4 C_{i,f}(T) * F_f \quad (3)$$

### 6.3 Numerical Results

Figure 25 gives a sampling of the  $C_i(T)$  results for example tank IDs 2, 7, 50, 75, 100, and 129 (a complete table of results can be found in Appendix F). Tank  $T=2$ , the probability that there are zero faults is 1, and the probability there are one or more faults is zero. This is again due to the fact that Tank 2 is adjacent to the CRs, leaving no chance for a fault to occur before it reaches the CRs. As the tanks get further away from the CRs, and increasing the number of hops to the CR, the probability there is a fault in the tanks path to the CRs increases. Tank  $T=50$  and  $T=75$  share very similar probabilities, because they have similar depths ( $d(50) = 20$ , and  $d(75) = 19$ ).



**Figure 25:  $C_i(T)$  Results**

Table 2 shows the  $C_i(T)$  results of two tanks at who share the same depth but have a different placement in the sector (tank  $T = 11$  and  $T = 15$ ). The Table shows that though the tanks share the same depth ( $d(11) = 9 = d(15) = 9$ ), they have slightly different values for  $C_i(T)$ . This is due to the different paths and legal fault scenarios for the tanks, as described in Section 5.1.1.

**Table 2:  $C_i(T)$  Comparison of Tanks at same depth**

<b>Tank ID</b>	<b><math>i = 0</math></b>	<b><math>i = 1</math></b>	<b><math>i = 2</math></b>	<b><math>i = 3</math></b>	<b><math>i = 4</math></b>
<b>11</b>	9.9919E-01	8.0996E-04	9.6653E-07	2.8783E-10	2.1219E-14
<b>15</b>	9.9919E-01	8.0989E-04	9.6628E-07	2.8765E-10	2.1202E-14

## 7 Binomial Analysis of MER

### 7.1 Context of MER

As illustrated in Figure 12, binomial distribution is used to calculate the Message Error Rate (MER), which is then used to calculate  $q_i(d(T))$ , the unreliability of a message given  $i$  faults in the path to the CR. Individual bit errors in a wireless channel are random, and tend to be independent and identically distributed (IID) – at least over a period of time that is long with respect to the length of one message. Therefore, the binomial distribution can be used to map random BER to MER [26].

### 7.2 Methodology

As implemented in the Auger Protocol, WAHREN messages employ significant error detection, but no error correction methods. The design decision not to include error correction was due the additional bandwidth and computation it would require, as error correction can be expensive. Without error correction, one flipped bit could corrupt an entire  $N$ -bit message. Due to this 1-of- $N$  fail property, the Binomial distribution reduces to the Product of Reliability Law, stating that the total reliability of combined components (the message) is the product of reliabilities the individual components (bits). Equation 4 applies the Product of Reliabilities Law to calculate message reliability ( $R_{msg}$ ) from bit reliability ( $R_i$ ), with a total of  $N$  bits.

$$R_{msg} = \prod_{i=1}^N R_i \quad (4)$$

The Bit Error Rate (BER) is the *unreliability* of a bit, and the desired value is the *unreliability* of a message, MER. Using the Definition of Unreliability ( $Q$ ) shown in Equation 5, and applying it to a full message of length  $N$  yields the expression for MER ( $Q_{msg}$ ) as a function of BER ( $Q_i$ ), as shown in Equation 6.

$$Q = 1 - R \quad (5)$$

$$Q_{msg} = 1 - \prod_{i=1}^N (1 - Q_i) = 1 - (1 - Q_i)^N \quad (6)$$

The target BER for Auger North is  $10^{-6}$ , similar to that of Auger South. This BER was determined to be a conservative value based on a Radio Frequency (RF) site survey of the RDA test location in Colorado [27]. The analysis was based on a 2.4 GHz RF design employing the MAX 2822 Transceiver chip, which has a specified BER =  $10^{-5}$  at its receiving sensitivity of -95 dBm at 1Mbps [28]. The survey showed that the vast majority of inter-station links could achieve a Received Signal Strength (RSS) of at least 12 dB above the receiver sensitivity without resorting to taller antenna masts. This 12 dB *fade margin* can significantly reduce the BER of the system.

A common rule of thumb, the one-decade-per-decade rule, states that one can expect a factor of 10 reduction in BER for every 10 dB increase in RSS. However, empirical studies have shown an even stronger relationship; in some cases, a factor of 10 improvement in BER can be achieved with only a few dB improvement in RSS [29][30][31]. Thus, given an expected 12 dB fade margin for most links, we can safely assume a BER =  $10^{-6}$  for the 2.45 GHz version of the hardware. Although hard numbers are currently unavailable for the as yet untested 4.6 GHz hardware, similar sensitivities can be expected. Therefore, our calculations for MER assume BER =  $10^{-6}$ .

### 7.3 Numerical Results

There are two types of messages relevant to this analysis. The first is the original Data message containing T2 Triggers and T3 Trigger response data. The second type is a request for retransmission, or Macro-NACK message from the CS back to the originating station in the event that an expected data message is not received. At the time of this calculation, an original message contained 2,944 bits<sup>2</sup>, and the Macro-NACK message contained 288 bits. The resulting values for calculated MERs are shown in Table 3.

---

In the WAHREN protocol, as implemented for Auger North, the Trigger message payload is actually broken into two adjacent messages. The message length listed here includes the combined payloads of both messages and all necessary overhead bits.

**Table 3: MERs for Relevant Message Types**

<b>Message Type</b>	<b>BER (<math>Q_i</math>)</b>	<b>Length (N)</b>	<b>MER (<math>Q_{msg}</math>)</b>
Trigger Data	1E-06	2,944	2.94E-03
Macro-Nack	1E-06	288	2.88E-04

## 8 DTMC Analysis of $q_i(d(T))$

### 8.1 Context of $q_i(d(T))$

As specified in Equation 2 and illustrated in Figure 11,  $q_i(d(T))$  is defined as follows. For a given tank (T), at depth  $d$ , with  $i$  faults in the path from tank  $T$  to the CR,  $q_i(d(T))$  is the probability that a message originated by Tank ( $T$ ) is *not* delivered to the CS because it was corrupted enroute to the CRs.

Discrete Time Markov Chains (DTMCs) were used to model  $q_i(d(T))$ . In general, DTMCs are used to model Markov (memoryless) processes where state transitions are functions of discrete events, rather than continuous real-time. Each message transmission between tanks is a discrete event, and therefore represents one step in discrete Markov process.

Four re-transmission scenarios were modeled to analyze which features of WAHREN provide the most improvement in reliability: Case-0 (with no on-demand retransmissions of lost messages), Macro-Retransmission, Micro-Retransmission, and Combined-Retransmission (which combines Macro and Micro retransmissions).

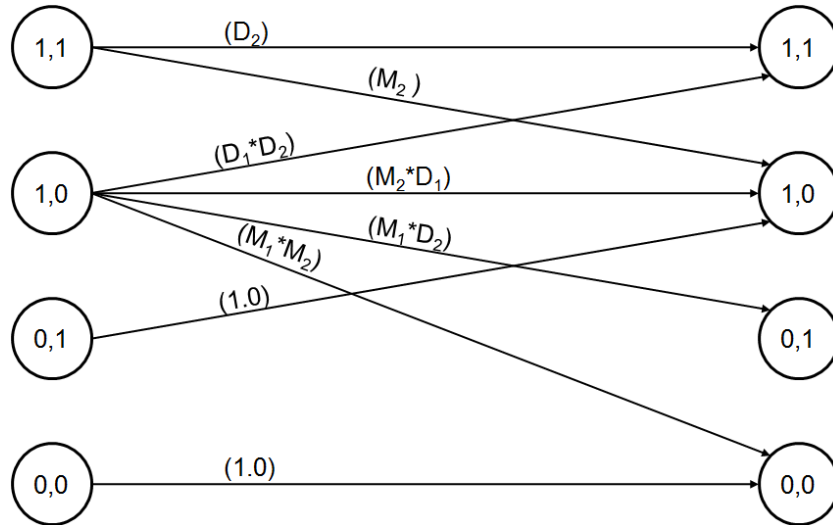
### 8.2 Methodology

#### 8.2.1 Case-0

The base case transmission scenario is Case-0. In this scenario, there is no on-demand retransmission for messages that were not received. Messages are sent from their originating tank, up the chain to the Concentrator Station (CS). With each “hop” along the chain, a message is transmitted from tank  $T$  to the next two tanks in the path to the CR ( $T-1$  and  $T-2$ ). Figure 26 shows a *before-and-after* representation of the potential states and transitions when a message is transmitted one hop along the chain.

The states on the *left* side of Figure 26 represent each possible state before one message transmission. The first number inside the state represents whether the *currently*

transmitting tank  $T$  has received the message, and the second number represents whether tank  $T-1$  (the next tank along the chain) has received the message.



**Figure 26: Case-0 Before-and-After State Diagram**

The states on the *right* side of Figure 26 represent tanks  $T-1$  (the left number in the state) and  $T-2$  (the right number in the state). Notice that  $T-1$  is the second number in the states on the left (before) side of Figure 26, and the first number in the states on the left side of Figure 26.

To clarify, the left hand side of the figure represents the states of tanks ( $T, T-1$ ) *before* tank  $T$  transmits (or relays) the message, while the right hand side represents the states of tanks ( $T-1, T-2$ ) *after* tank  $T$  transmitted the message. Thus, in all states, the first number listed represents the state of the *next tank to transmit*.

Each state variable is a Boolean value representing whether a tank is in possession of the message (“1” indicates it has the message, and “0” indicates it does not have the message). The arcs (transitions) represent potential state changes. The values corresponding to the arcs are the probabilities of changing from one state to the next. The values are represented as follows:

- $D_1$  = Probability the message was successfully delivered to tank  $T-1$ ,

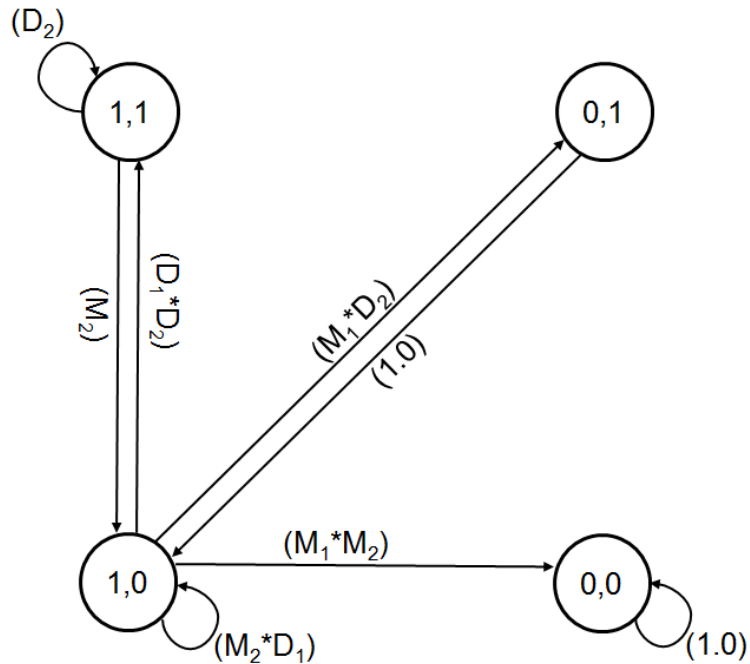


- $D_2$  = Probability the message was successfully delivered to tank  $T-2$ ,
- $M_1 = 1-D_1$  = Probability the message was *not* successfully delivered to tank  $T-1$ ,
- $M_2 = 1-D_2$  = Probability the message was *not* successfully delivered to tank  $T-2$ .

For a non-faulty receiving tank  $x$ ,  $M_x = \text{MER}$ ; but for a faulty receiving tank  $x$ ,  $M_x = 0$ . Thus, if both tanks  $T-1$  and  $T-2$  are non-faulty, then  $M_1 = M_2 = \text{MER}$ . However, if tank  $T-1$  is faulty, then  $M_1=1$  and  $D_1=0$  (indicating the message was never received by tank  $T-1$ ); similarly, if tank  $T-2$  is faulty,  $M_2=1$  and  $D_2=0$ .

Before-and-after diagrams such as Figure 26 are useful for visualizing message loss probabilities when going from one state the next, and for deriving the corresponding DTMC state transition diagram.

Figure 27 shows the DTMC state transition diagram derived from Figure 19 for Case-0. In Figure 27, each circle represents one of four possible states, where the *first* number in the state represents the next tank to transmit, while the second number represents the message status of the tank one hop along the chain (closer to the CS) from the next transmitting tank.



**Figure 27: Case-0 DTMC State Transition Diagram**

Equations for determining the probability that a message has proceeded to the next state were derived from the DTMC in Figure 27. The Next State Probabilities and their corresponding equations, based on the current state, are shown in Table 4. Similar to the state transition diagrams, the probability ( $P$ ) subscript numbers are Boolean values representing whether a tank has message to transmit (“1” indicates it has the message, and “0” indicates it does not have the message) for this tank and its neighbor.

**Table 4: Case-0 DTMC Equations**

Next State Probability	Probability Base on Current State
$P_{11}^+$	$P_{11}[D_2] + P_{10}[(D_1)(D_2)]$
$P_{10}^+$	$P_{11}[M_2] + P_{10}[(M_2)(D_1)] + P_{01}$
$P_{01}^+$	$P_{10}[(M_1)(D_2)]$
$P_{00}^+$	$P_{00} + P_{10}[(M_1)(M_2)]$

With  $M_1$  equal to the MER of tank  $T-1$ ,  $M_2$  equal to the MER of tank  $T-2$ , and their corresponding  $D_x = 1 - M_x$ , the next state probabilities are dependent on the probabilities of being in each current state as defined in the equations in Table 4.

To help demonstrate how the DTMC equation is applied to a tank, an example focused on tank ID 129 (the furthest tank away from the CS in a canonical sector) is provided for each of the retransmission scenarios. Setting the value of  $M$  to the MER of a T2 message, as listed in Table 1, and the probability of being in each current state  $P_{xx}$ , the next state probability,  $P_{xx}^+$  is calculated using the equations in Table 4.

Table 5 shows the next state probability for a message arriving at the CRs from tank 129, given no faults in the path. Tank ID 129 is at a depth of  $d(129) = 39$ , meaning the message must be transmitted 38 times along the chain before it is received by the CRs. The probability that in the next state, one or both CRs (tank IDs 0 and 1) will receive the message from tank 129 is  $P_{11}^+ + P_{10}^+ + P_{01}^+ = 0.999987$ . Similarly, the probability that *neither* CR received the message is  $q_0(39) = P_{00}^+ = 9.6095\text{E-}06$ . Due to the message traveling over dual paths for *every* hop, the probability neither CR receives the message is on the order of  $10^{-6}$ , or  $O(\text{MER}^2)$ .

**Table 5: Case 0 Arrival Probabilities at the Two Concentrator Radios (CRs) from Tank ID 129, with Zero Faults in the Path**

Probability	T2 Result	Unreliability $q_0(39)$
$P_{11}^+$	0.99703	9.6095E-06
$P_{10}^+$	2.9482E-03	
$P_{01}^+$	8.6414E-06	
$P_{00}^+$	9.6095E-06	

If there is a faulty tank in the path to the CRs, then the value of the faulty tank's MER is  $M = 1$ , and its value of  $D = (1 - M) = 0$ . It was empirically determined that the worst case scenario is for faults to be placed at the tanks closest to the message's originating tank.

Placing the faults near the originating tank was slightly worse than faults introduced in the middle of the chain, resulting in a mildly conservative value for  $P_{00}$  at the CRs.

As shown in Table 6, introducing one faulty tank in the path of a message originating at tank ID 129, decreases the probability one or both CRs have the message to 0.99410, increasing the probability that *neither* CR receive the message to  $q_1(39) = P_{00}^+ = 2.9492E-03$ . Introducing a faulty tank limits the transmission to follow one path around the faulty tank, resulting in a single non-redundant path for the message to travel. This single path increases the probability that a message is lost to the  $O(MER)$ , making it the dominant fault, as the rest of the path along the chain is still  $O(MER^2)$ . It was this dominant fault that was the motivation to implement retransmission into the WAHREN protocol.

**Table 6: Case 0 Arrival Probabilities at the Two Concentrator Radios (CRs) from Tank ID 129, with One Fault in the Path**

Probability	T2 Result	Unreliability $q_1(39)$
$P_{11}^+$	0.99410	2.9492E-03
$P_{10}^+$	2.9396E-06	
$P_{01}^+$	8.6160E-06	
$P_{00}^+$	2.9492E-03	

### 8.2.2 Macro-Retransmission

The second scenario analyzed is Macro-Retransmission. Macro-Retransmission only retransmits messages that the CS expected, but did not receive. If the CS does not receive an expected T2, it will send a Macro-Retransmission message (Macro-NACK) to the original source tank of the message. The Macro-NACK message is relayed down the chain to the source tank, which will then retransmit the original Trigger Data message up the chain to the CS. The CS will only request a Macro-Retransmission one time per expected message, meaning it will not send a second Macro-NACK if the retransmitted T2 was not received.

The Macro-Retransmission shares the same DTMC and Probability Equations diagrams as Case-0, as shown in Figure 26, Figure 27, and Table 4. Though they share the same DTMC diagram, they are numerically different due to the number of different message lengths. As shown previously in Table 3, message lengths are different for an original Trigger Data message and a Macro-NACK message. Macro-Retransmission will be more reliable overall because of the opportunity given for a retransmission of the original message, which was not provided with the Case-0 scenario.

Combinatorics are used to combine the DTMC results of the original T2 message, the Macro-NACK message, and the retransmitted T2 message to result in an overall unreliability  $q_f(h)$  of a message originating at a tank  $h$  hops away from the CR, with  $f$  faults in the path. With Macro-Retransmission, a message can be lost if and only if the original T2 message is lost, AND either the Macro-NACK message OR the retransmitted T2 message is lost. For clarity of the order of operations, reiterating the overall probability that *neither* CR receive the T2 message is the loss of the original T2 AND (the loss of the Macro-NACK OR the loss the retransmitted T2). Using the previously introduced Definition of Unreliability shown in Equation 5, the resulting overall unreliability  $q_f(h)$  is shown in Equation 7.

$$q_f(h) = P_{00_{T2}}^+ * (1 - ((1 - P_{00_{NACK}}^+) * (1 - P_{00_{T2}}^+))) \quad (7)$$

Table 7 shows the Macro-Retransmission arrival probabilities at the CRs, for a message originated by Tank 129.

Table 8 shows the values of the same probabilities with one fault in the chain. With the introduction of one fault in the chain, the probability *neither* CR receive the message increases from  $q_0(39) = 9.3150E-11$  to  $q_1(39) = 9.5450E-06$ . Enabling a second transmission of the T2 message decreased the single-fault probability that both messages are lost to  $O(MER^2)$ , which is comparable to Case-0 with no faults.

**Table 7: Macro-Retransmission Arrival Probabilities between the CRs and Tank ID 129, Zero Faults in the Path**

Probability	T2 Result	Macro-NACK Result	Unreliability $q_0(39)$
$P_{11}^+$	0.997034	0.999712	9.3150E-11
$P_{10}^+$	2.9482E-03	2.8804E-04	
$P_{01}^+$	8.6414E-06	8.2920E-08	
$P_{00}^+$	9.6095E-06	8.3828E-08	

**Table 8: Macro-Retransmission Arrival Probabilities between the CRs and Tank ID 129, with One Fault in the Path**

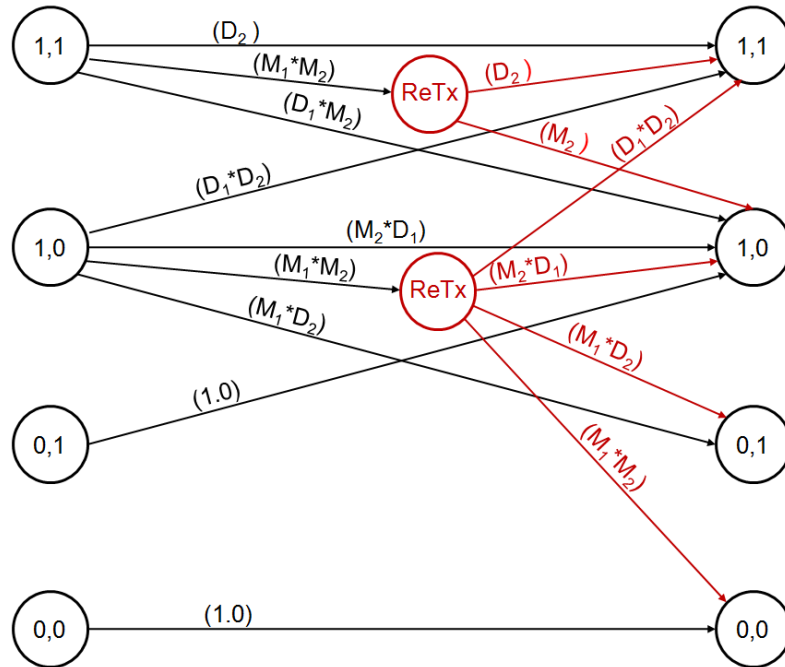
Probability	T2 Result	Macro-NACK Result	Unreliability $q_1(39)$
$P_{11}^+$	0.994103	0.999424	9.5450E-06
$P_{10}^+$	2.9396E-03	2.8796E-04	
$P_{01}^+$	8.6160E-06	8.2896E-08	
$P_{00}^+$	2.9492E-03	2.8804E-04	

### 8.2.3 Micro-Retransmission

The Micro-Retransmission scenario employs retransmission of messages lost between *neighboring tanks*. Once a receiving tank gets a message, it sends a positive acknowledgment (ACK). Due to the dual path, dual transmission of each message in the WAHREN protocol, every message transmitted by tank  $T$  should be received by two neighboring tanks ( $T-1$  and  $T-2$ ). The next time tanks  $T-1$  and  $T-2$  transmit along the chain, an acknowledgement to tank  $T$  is embedded into the forwarded message. Tank  $T$  eavesdrops on the message, checking a byte embedded in the header of the forwarded message which acknowledges receiving tank  $T$ 's message. Embedding the acknowledgements in the headers of the next messages forwarded by  $T-1$  and  $T-2$  minimizes the number of transmissions needed for the Micro-Retransmission scenario by not requiring a separate message solely for the Micro-ACK. If tank  $T$  does not receive a Micro-ACK from *either* tank ( $T-1$  or  $T-2$ ), it retransmits the T2 message (Micro-Retransmit). In other words, the *avoidance* of a retransmission by T requires a positive

ACK from at least one neighbor. Transmitting tanks will only Micro-Retransmit a message once.

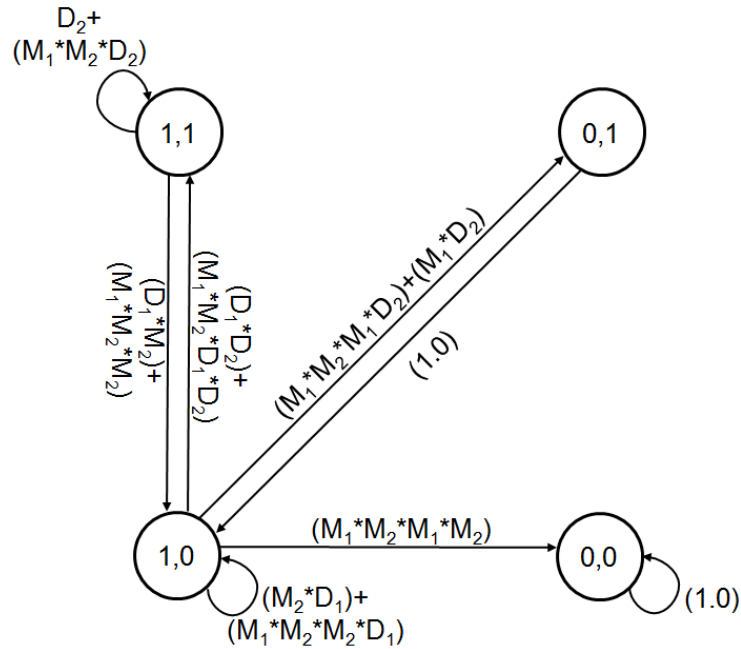
Figure 28 shows the Before-and-After relationships between the probabilities going from one state to the next in a similar manner as for Case-0 in Figure 26. Indicated with red color are the added actions imposed by a Micro-Retransmission between two tanks.



**Figure 28: Micro-Retransmission State Transition Diagram**

If tank  $T$ 's neighboring tank's ( $T-1$  or  $T-2$ ) message containing the ACK is lost, it is considered a missing ACK to tank  $T$ . This event is *not* built into the Before-and-After diagram, as it merely causes an unnecessary retransmission of a previously delivered message, and thus does not improve reliability.

Figure 29 is the DTMC state transition diagram of the Macro-Retransmission scenario derived from Figure 28.



**Figure 29: Micro-Retransmission DTMC State Transition Diagram**

The DTMC in Figure 29 was used to derive the Next State Probability equations for the Micro-Retransmission scenario, shown in Table 9.

**Table 9: Micro-Retransmit DTMC Equations**

Next State Probability	Probability Base on Current State
$P_{11}^+$	$P_{11}[D_2 + (M_1)(M_2)(D_2)] + P_{10}[(D_1)(D_2) + (M_1)(M_2)(D_1)(D_2)]$
$P_{10}^+$	$P_{11}[(D_1)(M_2) + (M_2)^2(M_1)] + P_{10}[(M_2)(D_1) + (M_2)^2(M_1)(D_1)] + P_{01}$
$P_{01}^+$	$P_{10}[(M_1)(D_2) + (M_1)^2(M_2)(D_2)]$
$P_{00}^+$	$P_{00} + P_{10}[(M_1)^2(M_2)^2]$

Again, let us focus on the example of a message originating at tank ID 129.

Table 10 displays the next state probabilities based on the values of the current state for the CRs receiving the message.



**Table 10: Micro-Retransmission Arrival Probabilities at the Two Concentrator Radios (CRs) from Tank ID 129, with Zero Faults in the Path**

Probability	T2 Result	Unreliability $q_0(39)$
$P_{11}^+$	0.99705	8.3019E-11
$P_{10}^+$	2.9396E-03	
$P_{01}^+$	8.6163E-06	
$P_{00}^+$	8.3019E-11	

In the Micro-Retransmission scenario, the probability that *neither* CR received the message is  $q_0(39) = P_{00}^+ = 8.3018E-11$  is comparable to that of the Macro-Retransmission scenario where  $q_0(39) = 9.135E-11$ . This is expected because in both scenarios, the T2 message has the opportunity to be *retransmitted* once. Both Macro- and Micro-Retransmission provide a better probability both CRs receive the T2 message when compared to Case-0 due to the possible second retransmission of the T2.

As shown in Table 11, introducing one faulty tank in the path of a message originating at tank ID 129 increases the probability that *neither* CR receive the message from  $q_0(39) = 8.3019E-11$  to  $q_1(39) = 8.6418E-06$ . Again, allowing a Micro-Retransmission of the T2 message decreased the probability that both messages are lost to the  $O(MER^2)$ , which is similar to that of Case-0 with no faults.

**Table 11: Micro-Retransmission Arrival Probabilities at the Two Concentrator Radios (CRs) from Tank ID 129, with One Fault in the Path**

Probability	T2 Result	Unreliability $q_1(39)$
$P_{11}^+$	0.99704	8.6418E-06
$P_{10}^+$	2.9396E-03	
$P_{01}^+$	8.6162E-06	
$P_{00}^+$	8.6418E-06	

#### 8.2.4 Combined-Retransmission

Combined-Retransmission uses both Micro-Retransmission and Macro-Retransmission to get an original Trigger Data message (T2) to the CS. The original T2 message is transmitted from tank  $T$  to two tanks up the chain ( $T-1$  and  $T-2$ ). If tank  $T$  does not receive an ACK by either tank  $T-1$  or  $T-2$ , it automatically retransmits the message (Micro-Retransmission). In addition, if the message does not make it all the way to the CS, a Macro-NACK is sent by the CS, along the chain to the message's originating tank, which initiates a Macro-Retransmission of the lost message. The original message is then retransmitted up the chain, again following the Micro-Retransmission protocol.

In Combined-Retransmission, a single message shares the same DTMC and Probability Equations as Micro-Retransmit, as shown in Figure 28, Figure 29, and Table 9. Although they share the same DTMC diagram, Micro-Retransmission and Combined-Retransmission are numerically different due to the number of messages transmitted and different message lengths.

In addition, the same combinatorics defined in Equation 7 are used to combine the DTMC results of the original T2 message, the Macro-NACK message, and the Macro-retransmitted T2, all with enroute Micro-Retransmission. The result is the overall unreliability  $q(h)$  of a message originating at a tank  $h$  hops away from the CR, with  $f$  faults.

Focusing again on tank ID 129, the probability of the next state depends on the current state. Because the next state probabilities share the same equations as Micro-retransmit, it also shares the values in

Table 10. The overall resulting unreliability  $q_f(h)$  it takes into account both the Micro- and the Macro-retransmission of the T2 message (Table 12).

**Table 12: Macro-Retransmission Arrival Probabilities at the Two Concentrator Radios (CRs) from Tank ID 129, Zero Faults in the Path**

Probability	T2 w/ Micro Result	Macro-NACK Result	Unreliability $q_0(39)$
$P_{11}^+$	0.99705	0.99971	6.893E-21
$P_{10}^+$	2.9396E-03	2.8796E-04	
$P_{01}^+$	8.6163E-06	8.2896E-08	
$P_{00}^+$	8.3019E-11	6.9510E-15	

Table 13 shows the Combined-Retransmission arrival probabilities with *one* fault in the chain. With the introduction of one fault in the chain, the probability *neither* CR receives the message increases from  $q_0(39) = 6.893E-21$  to  $q_1(39) = 7.504E-11$ . With one faulty tank in the chain, the probability that neither CR receive the T2 message goes from  $q_0(CASE0)(39) = 2.9492E-03$  to  $q_0(COMBO)(39) = 7.504E-11$ . Therefore, implementing the Macro- and Micro-Retransmission significantly decreases the probability that neither CR receive the T2 message.

**Table 13: Macro-Retransmission Arrival Probabilities at the Two Concentrator Radios (CRs) from Tank ID 129, One Fault in the Path**

Probability	T2 w/ Micro Result	Macro-NACK Result	Unreliability $q_1(39)$
$P_{11}^+$	0.99704	0.99971	7.504E-11
$P_{10}^+$	2.9396E-03	2.8796E-04	
$P_{01}^+$	8.6162E-06	8.2896E-08	
$P_{00}^+$	8.6418E-06	8.2920E-08	

### 8.3 Numerical Results

Because this model is a sequence of piecewise DTMCs, it was evaluated using a commercial spreadsheet program for each of the four retransmission scenarios. The spreadsheet simplified the insertion of faulty tanks into the chain, by allowing the *value* of MER at a faulty tank to be altered without altering the form of the state transition equations.

At the top of the each sheet, as shown in

Table 14, the MER is calculated based on the BER, and message length for the T2 and Macro-NACK messages, respectively (as described in Section 7). The T2 and Macro-NACK MER's are then used as global variables in subsequent calculations.

**Table 14: DTMC Spreadsheet: MER Calculation**

Fault	Hop	T2		Macro-NACK	
1	1	Pkt Len (B)	368	Pkt Len (B)	36 (Bytes)
2	99	Pkt Len (b)	2944	Pkt Len (b)	288 (Bits)
3	99	BER	1.0E-06	BER	1.0E-06
4	99				
5	99	Max Hops	38	Max Hops	38
6	99	MER	2.940E-03	MER	2.880E-04

The *Fault* column represents potential faults that the user can insert into the chain. The *Hop* column allows the user to enter the hop at which to insert a faulty tank. The value “99” is a placeholder that does not insert a fault, as here are only 38 possible hops.

Table 14 shows one fault in the chain at hop 1 (adjacent to the originating tank).

Below the MER calculation, is a table of next state probabilities for increasing numbers of hops away from the originating tank. One row is dedicated for each value of hops  $h$  (ranging from 0 to a maximum of 38 hops). The column labels  $\text{Pr}(00)$  through  $\text{Pr}(11)$  represent the state probabilities  $P_{00}^+$  through  $P_{01}^+$ , respectively. The initial condition is at zero hops, where only the *originating* tank  $T$  has the message. Thus, following the state definitions of Subsection 8.1, the values for the initial condition at zero hops are:  $P_{10}^+ = 1.0$  and  $P_{00}^+ = P_{01}^+ = P_{11}^+ = 0.0$ . The subsequent rows are the state probabilities of the message after  $h$  hops from originating tank  $T$  (Table 15).

**Table 15: DTMC Spreadsheet, Initial State**

MER	MDR	Hops (h)	Pr( 00 )	Pr( 01 )	Pr( 10 )	Pr( 11 )
<b>0.0029</b>	0.9971	<b>0</b>	<b>0.00</b>	<b>0.00</b>	<b>1.00</b>	<b>0.00</b>
<b>1.0000</b>	0.0000	<b>1</b>	2.9397E-03	9.9706E-01	0.0000E+00	0.000000
<b>0.0029</b>	0.9971	<b>2</b>	2.9397E-03	0.0000E+00	9.9706E-01	0.000000
<b>0.0029</b>	0.9971	<b>3</b>	2.9483E-03	2.9224E-03	2.9224E-03	0.991207
<b>0.0029</b>	0.9971	<b>4</b>	2.9483E-03	8.5657E-06	5.8448E-03	0.991198

The  $M_x$ , (represented by MER), and  $D_x$  ( $1-M_x$ , represented by MDR) values are listed in the two left-hand columns for each corresponding hop  $h$ . When a faulty tank is introduced using the Hop column, the MER at the specified hop becomes 1, resulting in that tank's  $MDR = 0$ , as shown at hop 1 in Table 16. Creating columns for the individual values of MER and MDR at each hop  $h$ , made it simple to add a fault, allowing the equations for  $P_{xx}$  to remain the same for all rows of a particular retransmission scenario.

This particular example is for Case-0 with one fault. The other retransmission scenarios use the same approach represented in this example, but have different results due to having different state transition equations for  $Pr(xx)$  embedded in the spreadsheet cells. Macro-Retransmit and Combined-Retransmit have more than one message, therefore two tables of cells were created side-by-side, with one table of cells representing the original T2 message, and a second table of cells (using the same  $P_{xx}$  equations) for the Macro-NACK message (Appendix G). Micro-Retransmit looks similar to Case-0 with one table of message probabilities per hop, with different  $Pr(xx)$  and  $Q(h)$  equations.

**Table 16: DTMC Spreadsheet, MER and MDR at Each Hop**

MER	MDR	Hops (h)	Pr(00)	1
0.0029	0.9971	0	0.00	
1.0000	0.0000	1	2.9397E-03	9
0.0029	0.9971	2	2.9397E-03	0
0.0029	0.9971	3	2.9483E-03	2
0.0029	0.9971	4	2.9483E-03	8

MER = 1, indicating a fault tank at hop 1.

➔

Non-faulty tank MER calculated from BER and message length.

➔

Starting with the initial state for  $h=0$  hops, each next row's equations were evaluated by using the previous row's state values. For each hop  $h$ , using the equations from Table 4 or Table 9 (depending on the retransmission scenario), each  $P_{xx}$  next state probability after  $h$  hops was based on the current state of its two neighboring tanks at  $(h-1)$  and  $(h-2)$ .

For each retransmission scenario, and for each hop  $h$ ,  $i$  faulty tanks were introduced to the scenario. As stated previously, empirical results showed that placement of faulty tanks nearest to the originating tank provide slightly worst case DTMC results. For multiple fault scenarios, faults were placed at every second tank, because if there were two adjacent faulty tanks, that would be a systemic failure, which is not the focus of this analysis. Thus, faulty tanks were placed at hops one, three, five, and seven. First evaluating zero faults, then adding one fault at a time starting with the first fault placed at hop one (second fault at hop three, third fault at hop five and fourth fault at hop seven),  $q_i(h)$  was calculated for each value of hops ( $h$ ) and each number of faults-in-path ( $i$ ). Using the depth equation ( $d(T)=h(T)+1$ ),  $q_i(d(T))$  was calculated from  $q_i(h)$ .

Table 17 shows the columns to the right of the Pr( xx ) columns. First is a column labeled *1-SUM of Probs*, which is used for error-checking. Since all four  $P_{xx}$  values at a particular hop should always sum to 1.0, the contents of this column should all be 0.0. The next column *Hops (h)* is repeated for easier visualization. The column  $n(h)$  states how many *faulty* tanks are at that row's path. The column furthest to the right,  $Q(h)$ , has the

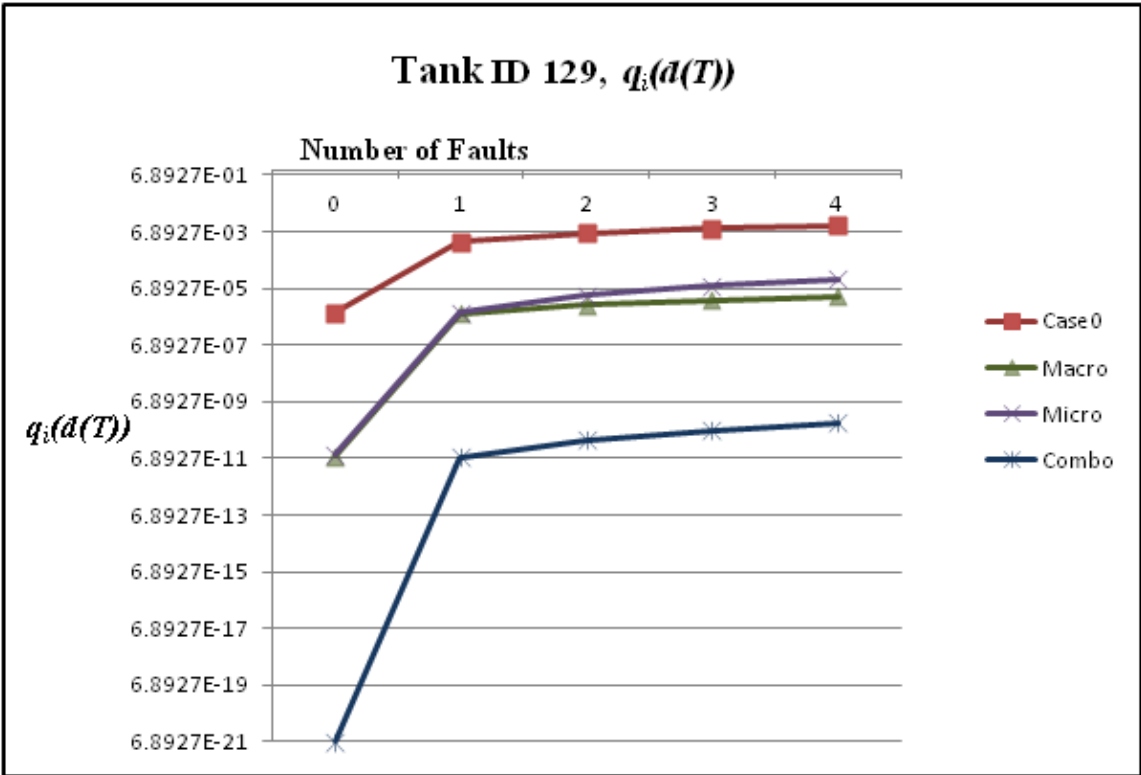
resulting  $q_i(h)$  with the variable  $i$  as determined by the number of faults inserted into the chain by the user with the *Hop* column from Table 14.

**Table 17: DTMC Spreadsheet, Resulting Q(h)**

<b>0 )</b>	<b>Pr( 11 )</b>	<b>1-SUM of Probs</b>	<b>Hops (h)</b>	<b>n(h)</b>	<b>Q(h)</b>
<b>.00</b>	<b>0.00</b>	0.00E+00	<b>0</b>	<b>0</b>	
3+00	0.000000	0.00E+00	<b>1</b>	1	2.9397E-03
E-01	0.000000	0.00E+00	<b>2</b>	1	2.9397E-03
E-03	0.991207	0.00E+00	<b>3</b>	2	2.9483E-03
E-03	0.991198	0.00E+00	<b>4</b>	2	2.9483E-03

The complete numerical results of  $q_i(d(T))$  for each of the retransmission scenarios per each tank  $T$ , given 0 through 4 faults, depth ranging from 1 to 39, are displayed in Appendix H. The results show how the different retransmission scenarios affect the probability that a message originated by Tank ( $T$ ) is *not* delivered to the CS because it was corrupted enroute to the CRs,  $q_i(d(T))$ .

To help illustrate the affect the retransmission scenario has on  $q_i(d(T))$ , Figure 30 displays the unreliability at the CR's of a message originating at tank ID 129. The graph shows that Macro-retransmission and Micro-retransmission can each decrease the probability of a message lost enroute to the CRs by 2 to 5 orders of magnitude. Similarly, combining Macro- and Micro-Retransmission decreases the unreliability range from 7 to 15 orders of magnitude. The difference in unreliability shows the effectiveness of adding either Micro-, Macro-, or Combined-Retransmissions.



**Figure 30: Originating Message from Tank ID 129, CR  $q_i(d(T))$**



## 9 Final Top-Level Results

A commercial spreadsheet was used to record the results of each sub-model shown in Figure 12 and combine them to calculate tank unreliability ( $Q(T)$ ) in accordance with Equation 2. Section 9.1 takes a step-by-step approach, explaining how the results of each sub-model were recorded and combined to get tank unreliability ( $Q(T)$ ). Section 9.2 explains how sector unreliability  $Q_{sect}$  was derived from tank unreliability ( $Q(T)$ ). Finally, Section 9.3 explains achieving the desired end result of system unreliability  $Q_{sys}$ , along with additional statistics for message loss.

### 9.1 Calculations Explained

Unreliability of the message flow from tank  $T$  to concentrator radios (CRs) has been calculated for all four retransmission scenarios. There were several calculations that lead to the desired probability that a message was lost. This section walks through how the results from the sub-models were combined to establish the reliability of each tank, as illustrated in Figure 12. In addition to describing the methodology of combining the sub-model results, this section takes a step-by-step approach describing one example tank's Case-0 calculations to help illustrate the process. Though this example focuses on tank  $T = 7$ , the methodology applies to all tanks.

The first sub-model evaluated  $F(f)$ , the fraction of time in which there exists  $f$  faults in a sector. As described in Section 4, Generalized Stochastic Petri Nets (GSPNs) were used to evaluate  $F(f)$ , using the commercially available *Stochastic Petri Net Package* (SPNP) for analysis and evaluation. The complete GSPN results for  $F(f)$ , summarized in Table 1, showed that the probability of there being more than 4 active faults in a sector was negligible.

The second sub-model evaluated  $C_{if}(T)$ , defined as the probability that  $i$  faults are in tank  $T$ 's path, given  $f$  faults in a sector.  $C_{if}(T)$  was determined by exhaustively searching all legal fault locations for each tank within the array.

Table 18 is a section of Appendix D, with  $(C_{if}(T))$  for tank ID 7 highlighted.

Table 18 only shows results for  $i = 0$  and part of  $i = 1$ ; Appendix D lists the full results for faults in path  $i = [0 \dots 4]$ , and existing faults  $f = [1 \dots 4]$ .

**Table 18:  $C_{if}(T)$  of Tank ID 7**

**$C_{if}(T)$**

<b>Tank</b>	<b>i = 0</b>	<b>i = 0</b>	<b>i = 0</b>	<b>i = 0</b>	<b>i = 1</b>	<b>i = 1</b>
<b>ID</b>	<b>f = 1</b>	<b>f = 2</b>	<b>f = 3</b>	<b>f = 4</b>	<b>f = 1</b>	<b>f = 2</b>
0	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00
1	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00
2	1.000E+00	1.000E+00	1.000E+00	1.000E+00	0.000E+00	0.000E+00
3	9.921E-01	9.840E-01	9.756E-01	9.670E-01	7.874E-03	1.600E-02
4	9.843E-01	9.684E-01	9.524E-01	9.363E-01	1.575E-02	3.162E-02
5	9.764E-01	9.528E-01	9.292E-01	9.056E-01	2.362E-02	4.711E-02
6	9.843E-01	9.684E-01	9.524E-01	9.363E-01	1.575E-02	3.162E-02
7	9.764E-01	9.528E-01	9.292E-01	9.056E-01	2.362E-02	4.710E-02
8	9.685E-01	9.374E-01	9.067E-01	8.764E-01	3.150E-02	6.221E-02
9	9.606E-01	9.220E-01	8.842E-01	8.472E-01	3.937E-02	7.720E-02

The third sub-model uses combinatorial algebra to apply the results of the first two sub-models,  $F(f)$  and  $C_{if}(T)$ , to determine  $C_i(T)$ , the unconditional probability  $i$  faults are in tank  $T$ 's path to the CR. The probabilities from the GSPN sub-model results for  $F(f)$  (Appendix C) are multiplied, using Equation 3, with Exhaustive Search sub-model results for  $C_{if}(T)$  (Appendix D), resulting in  $C_i(T)$  (Appendix F). The value of  $C_i(T)$  for tank ID 7 is highlighted in Table 19.

Table 19:  $C_i(T)$  of Tank ID 7

$C_i(T)$

Tank	i=0	i=1	i=2	i=3	i=4
0	9.858E-01	0.000E+00	0.000E+00	0.000E+00	0.000E+00
1	9.858E-01	0.000E+00	0.000E+00	0.000E+00	0.000E+00
2	1.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00
3	9.999E-01	1.161E-04	0.000E+00	0.000E+00	0.000E+00
4	9.998E-01	2.320E-04	0.000E+00	0.000E+00	0.000E+00
5	9.997E-01	3.479E-04	6.457E-08	0.000E+00	0.000E+00
6	9.998E-01	2.320E-04	0.000E+00	0.000E+00	0.000E+00
7	9.997E-01	3.479E-04	6.455E-08	0.000E+00	0.000E+00
8	9.995E-01	4.635E-04	1.935E-07	0.000E+00	0.000E+00
9	9.994E-01	5.791E-04	3.869E-07	2.883E-11	0.000E+00

The fourth sub-model determined the Message Error Rate (MER) for both original Data messages containing T2 Triggers and T3 Trigger response data, and the Macro-NACK message which requests a Macro-retransmission. Due to the nature of individual bits in a wireless channel being independent and identically distributed, binomial distribution was used to map Bit Error Rate (BER) to MER (results in Table 3).

The fifth sub-model evaluated  $q_i(d(T))$ , defined as the probability that a message originated by tank ( $T$ ), at depth  $d$ , with  $i$  faults in the path from tank  $T$  to the CR, is *not* delivered to the CS because it was corrupted enroute to the CRs. Each message transmission is a discrete event, therefore Discrete Time Markov Chains (DTMCs) were used to model  $q_i(d(T))$ , using the binomial distribution sub-model resulted MER as the probability a message was not delivered. Each of the four re-transmission scenarios were evaluated (Case-0-, Micro-, Macro-, and Combined-retransmission), for each tank  $T = [0 \dots 129]$  (includes both CRs) and fault in path  $i = [0 \dots 4]$  combination (results in Appendix H).

Tank ID 7's depth  $d(7) = 5$ , which is defined as the number of hops to the CR plus one (layout previously shown in Figure 11). Table 20 is a section of Appendix H (the DTMC sub-model results) highlighting the  $q_i(d(T))$  results for tank ID 7, in the Case-0-retransmission scenario.

**Table 20:  $q_i(d(T))$  of Tank ID 7**

			<b>Case0 <math>q_i(d(f))</math></b>				
<b>Tank</b>	<b>Hops</b>	<b>Depth</b>	<b>0 Faults</b>	<b>1 Fault</b>	<b>2 Faults</b>	<b>3 Faults</b>	<b>4 Faults</b>
0	0	0	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00
1	0	1	0.000E+00	0.000E+00	0.000E+00	0.000E+00	0.000E+00
2	1	2	8.642E-06	2.940E-03	2.940E-03	2.940E-03	2.940E-03
3	2	3	8.667E-06	2.940E-03	2.940E-03	2.940E-03	2.940E-03
4	3	4	8.718E-06	2.948E-03	5.871E-03	5.871E-03	5.871E-03
5	4	5	8.743E-06	2.948E-03	5.871E-03	5.871E-03	5.871E-03
6	3	4	8.718E-06	2.948E-03	5.871E-03	5.871E-03	5.871E-03
7	4	5	8.743E-06	2.948E-03	5.871E-03	5.871E-03	5.871E-03
8	5	6	8.769E-06	2.948E-03	5.879E-03	8.793E-03	8.793E-03

The sixth and final sub-model evaluates the unreliability of tank  $T$ ,  $Q(T)$ . The probability  $i$  faults are in tank  $T$ 's path ( $C_i(T)$  Combinatorial Algebra sub-model results), are combined with the unreliability of a message given  $i$  faults in the path to the CR ( $q_i(d(T))$  DTMC sub-model results) using Equation 8.

$$Q(T) = \sum_{i=0}^4 q_i(d(T)) * C_i(T) \quad (8)$$

$Q(T)$  was calculated for each tank  $T$  in each of the four retransmission scenarios, Case-0-, Micro-, Macro- and Combined-Retransmit (Appendix I). Table 21 is a section of Appendix I, highlighting the  $Q(T)$  calculation for tank ID 7.

Table 21:  $Q(T)$  of Tank ID 7

$Q(T)$

Tank	Case0	Micro	Macro	Combo
0	0.000E+00	0.000E+00	0.000E+00	0.000E+00
1	0.000E+00	0.000E+00	0.000E+00	0.000E+00
2	8.642E-06	7.468E-11	7.540E-11	5.577E-21
3	9.007E-06	1.078E-09	1.177E-09	8.754E-15
4	9.400E-06	2.080E-09	2.290E-09	1.749E-14
5	9.766E-06	3.083E-09	3.398E-09	2.625E-14
6	9.400E-06	2.080E-09	2.290E-09	1.749E-14
7	9.766E-06	3.083E-09	3.398E-09	2.625E-14
8	1.013E-05	4.085E-09	4.507E-09	3.501E-14
9	1.050E-05	5.097E-09	5.619E-09	4.279E-14

Finally, the reliability of tank  $T$  can be calculated using the Definition of Reliability Law shown in Equation 9. The results for  $R(T)$  for each tank in the sector are in Appendix J.

$$R(T) = 1 - Q(T) \quad (9)$$

## 9.2 Sector Results

The previous section calculated the unreliability (probability of non-delivery) for a message originating at any given tank. This section, calculates Sector Unreliability ( $Q_{sect}$ ) as the combined unreliability for all 128 tanks in a canonical sector. Since each tank issues one original message per second, the resulting value of  $Q_{sect}$  is the probability that a message is lost *per-second*. This value will be scaled to more useful metrics, including the mean time between messages, and the expected number of messages lost per year.

One sector comprises a 1-of- $N$  fail system, which means if one message out of  $N$  total messages does not get forwarded from its originating tank to the CR, then the system has failed. Therefore, it is not necessary for all component  $R$ 's to be identical; i.e. each component can have an individual value of  $R$ . Due to the 1-of- $N$  fail property, the Product of Reliability Law applies, stating that the total reliability of combined parts is the product of the each part's reliabilities. Thus, the Reliability of the sector  $R_{sect}$  is the product over all tanks,  $T$ , of the tank's reliabilities  $R_T$ , as shown in Equation 10.

$$R_{sect} = \prod R_T \quad (10)$$

Using the Definition of Unreliability (Equation 5), the sector unreliability  $Q_{sect}$  was calculated for each retransmission scenario based upon the sector's reliability  $R_{sect}$  (Equation 11).

$$Q_{sect} = 1 - R_{sect} \quad (11)$$

Additional statistics on the sector include the time between faults in one sector (Equation 12), and the expected number of messages lost per year (Equation 13), given one original message per tank per second, and 31,556,926 seconds in one year.

$$\text{Mean Time between lost messages}_{sect} = MTBL_{sect} = \frac{1}{Q_{sect}} \quad (12)$$

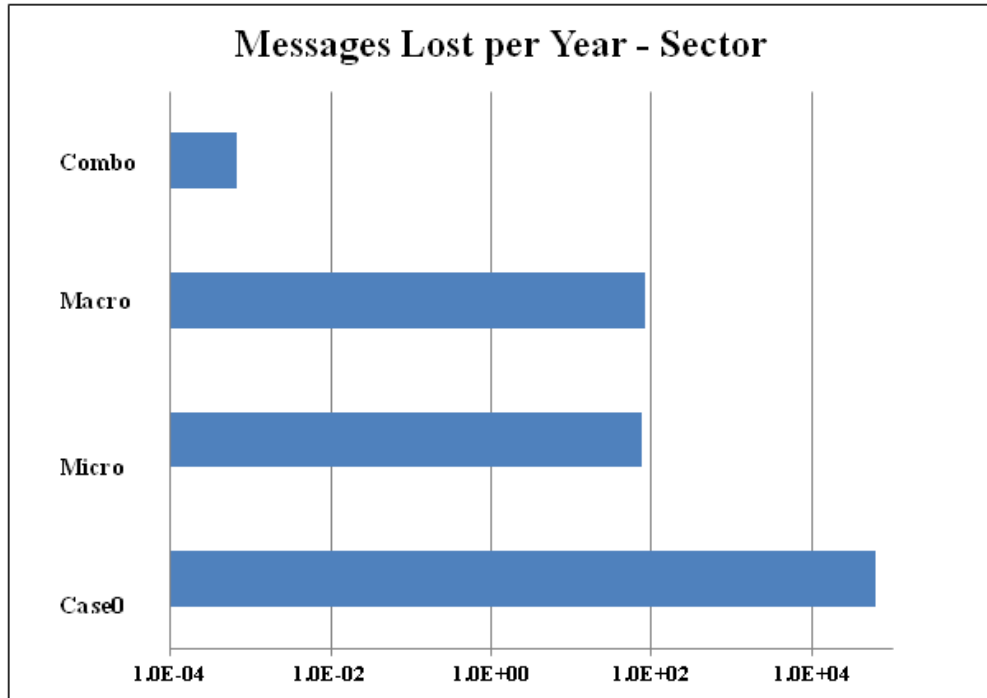
$$\text{Messages lost per year}_{sect} = \frac{31,556,926}{MTBL_{sect}} \quad (13)$$

The results of Equations 10 through 13 are shown in Table 22.

**Table 22: Sector Results**

Sector	Case0	Micro	Macro	Combo
$R_{sect} = \prod(R_T)$	9.980E-01	1.000E+00	1.000E+00	1.000E+00
$Q_{sect} = 1 - R_{sect}$	1.975E-03	2.383E-06	2.660E-06	2.093E-11
Mean Time Between Lost msgs (sec) = $Q_{sect}^{-1}$	5.063E+02	4.196E+05	3.760E+05	4.777E+10
Msgs lost in a year = $(31,556,926/Q_{sect}^{-1})$	6.232E+04	7.521E+01	8.393E+01	6.606E-04

Without any retransmission, Case-0, over 62,000 messages per year will be lost. Adding either Macro or Micro-retransmission decreases that amount by three orders of magnitude. Combining both Macro- and Micro-Retransmission reduces the message losses by another five orders of magnitude. Results for each retransmission scenario are displayed in Figure 31.



**Figure 31: Messages Lost per Year Sector Results**

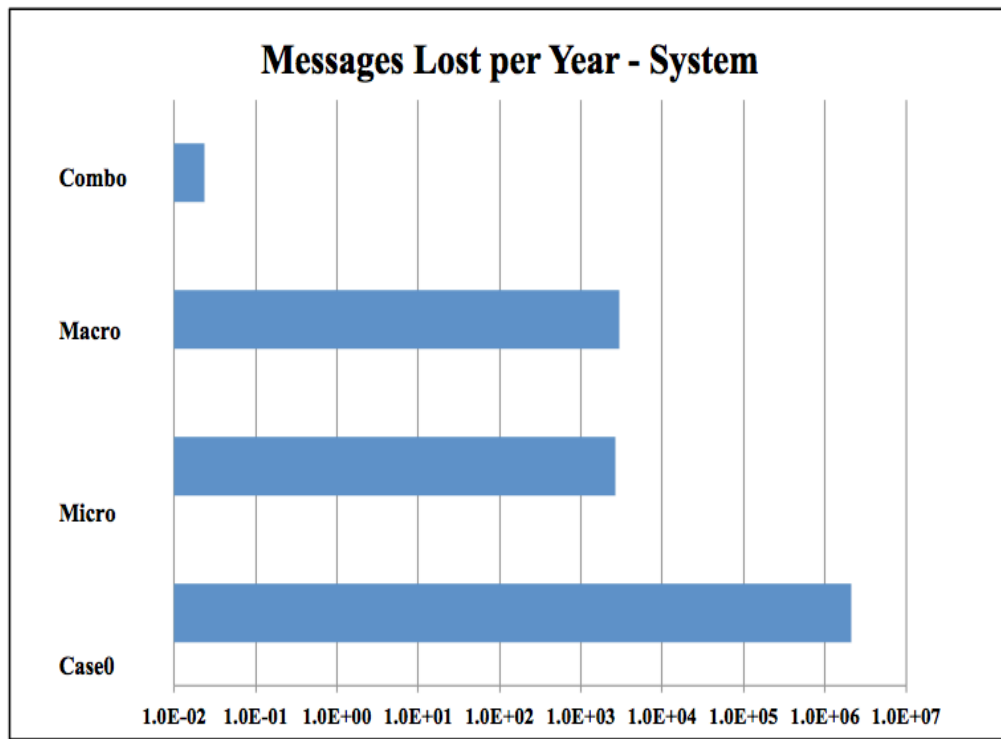
### 9.3 System Results

The same principles used to calculate sector results, can be used to calculate system results. The Product of Reliability Law allows the calculation of system reliability  $R_{sys}$ , by taking the product of the reliabilities of all sectors. As previously stated, there are a total of 4,400 tanks in the Auger North site, and divided by 128 (the average number of tanks per sector) the result is 34.4 sectors. For purposes of this study we can round this up to an even 35 sectors, adding a conservative error of less than 1.1%. Equations 11 through 13 can be applied to system results by simply using the system variables instead of sector. Table 23 displays the results for the entire Auger North as a system.

**Table 23: System Results**

System = 35 Sectors	Case0	Micro	Macro	Combo
$R_{sys} = \prod(R_{sect})$	9.3315E-01	9.9992E-01	9.9991E-01	1.0000E+00
$Q_{sys} = 1 - R_{sys}$	6.6851E-02	8.3409E-05	9.3086E-05	7.3262E-10
Mean Time Between Lost msgs (sec) = $Q_{sys}^{-1}$	1.4959E+01	1.1989E+04	1.0743E+04	1.3650E+09
Msgs lost in a year = $(31,556,926/Q_{sys}^{-1})$	2.1096E+06	2.6321E+03	2.9375E+03	2.3119E-02

The results in Table 23 mirror those for a single sector. In the best case of Combine Retransmission, only an average of 2.312E-02 messages are lost per year. Results for each retransmission scenario are displayed in Figure 32.



**Figure 32: Messages Lost per Year System Results**



## 10 Conclusions

The Pierre Auger Cosmic Ray Observatory North site employs a large array of surface detector stations (tanks) to detect the secondary particle showers generated by ultra-high energy cosmic rays. Due to the rare nature of ultra-high energy cosmic rays, it is important to have a high reliability on tank communications, ensuring no valuable data is lost. The Auger North site employs a peer-to-peer paradigm, the *Wireless Architecture for Hard Real-Time Embedded Networks* (WAHREN), designed specifically for highly reliable message delivery over fixed networks, under hard real-time deadlines. This thesis used a multimodal approach to prove that a performance and reliability analysis of WAHREN was possible, and provides the results of the analysis.

The WAHREN design includes two retransmission protocols, Micro- and Macro-retransmission. To fully understand how each retransmission protocol increased the reliability of communications, this analysis evaluated the system without using either retransmission protocol (Case-0), both Micro- and Macro-retransmission individually (Micro and Macro), and Micro- and Macro-retransmission combined (Combo).

The system of 4,400 tanks without retransmission will miss  $2.110E+6$  messages per year. When using only one retransmission protocol, both Micro- and Macro-retransmission had similar results of missing  $2.632E+03$  and  $2.938E+03$  messages per year. Adding both Micro- and Macro-retransmission, the number of messages lost in a year drastically decreases to  $2.312E-02$ . These results can help to inform the design decision process for the Auger North communication system.

## 11 Future Work

This thesis developed a multimodal modeling framework to estimate the rate of message loss in a sector of the Auger North array that has not suffered systemic failure. Now that this framework exists, it can be calibrated and integrated into other models to complete the end-to-end analysis of Auger North dependability. Specifically:

1. The values used for Message Error Rate (MER) herein were derived from best estimates of the Bit Error Rate (BER), based on a specific transceiver, common assumptions about BER vs. receiver sensitivity, and a previous in-situ signal loss survey conducted over part of the Auger North array site. The only way to achieve accurate values for BER is to do so empirically. The Research and Development Array (RDA) currently under construction can provide this information.
2. This study assumed that a sector has not suffered a systemic failure, i.e. no two faults are in adjacent tanks; thus, the peer-to-peer chain has not been broken. The results of this study can be integrated with earlier studies on the probability of systemic failure in order to estimate the total number of messages expected to be lost in all operating states.
3. This study assumed that the Concentrator Radios (CRs) and Concentrator Station (CS) were non-faulty. Once a detailed design study and reliability assessment of the CR/CS complex has been completed, the results of that study can be integrated with the array dependability results presented herein.

## References

- [1] Pierre Auger Collaboration (undated), *Auger Observatory: A new Astrophysics Facility Rises from the Pampa* [online]. Available: <http://www.auger.org/observatory/>
- [2] Zbigniew Szadkowski, "Triggers for the Pierre Auger Observatory, the Current Status and Plans for the Future" Wydawnictwo Politechniki, Warszawskiej, 2009.
- [3] R.M. Kieckhafer, "Hard Real-Time wireless communication in the northern Pierre Auger Observatory," Real Time Conference (RT), 2010 17th IEEE-NPSS , vol., no., pp.1-8, 24-28 May 2010.
- [4] R.M. Kieckhafer, et al, "A New Paradigm for Reliable Hard-Real-Time WSNs", *IEEE Real-Time Syst. Symp. (RTSS-06), Work in progress Session*, Dec 2006.
- [5] D. Nitz and the Pierre Auger Collaboration "Instrumentation for the Northern Site of the Pierre Auger Observatory", *Nucl. Instruments and Methods* (to appear).
- [6] J. Blumer and the Pierre Auger Collaboration, "The Northern Site of the Pierre Auger Observaroty", *New J. of Physics* vol. 12, Mar 2010.
- [7] R. Kieckhafer, D. Nitz, "Auger North SD Communication System: Analysis of Reliability to Systemic Faults," GAP Note 2008-131, 2008.
- [8] B. Effantin and H. Kheddouci, "The b-Chromatic Number of Some Power Graphs", *Discrete Math and Theoretical Computer Science*, vol. 6, 2003, pp. 45-54.
- [9] B. Hall, et al, "Ringin Out Fault Tolerance: A New Ring Network for Superior Low-Cost Dependability", *Proc. 2005 International Conference on Dependable Systems and Networks (DSN'05)*, pp 298-307, July 2005.
- [10] F. Bogenberger, "FlexRay, and Where It Stands Today", *SAE World Congress*, Mar 2002.
- [11] *Road Vehicles – Controller Area Network (CAN) – Part 4: Time Triggered Communications*, ISO Standard 11898-4, 2004.
- [12] *Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LRWPANs)*, IEEE Standard 802.15.4, 2003.
- [13] M.H. Azadmanesh, R.M. Kieckhafer, "Exploiting Omissive Faults in Synchronous Approximate Agreement," *Computers, IEEE Transactions on* , vol.49, no.10, pp.1031-1042, Oct 2000.

- [14] D. Huber, et al, "SILK: An Implementation of a Buffer Insertion Ring," *Selected Areas in Communications, IEEE Journal on* , vol.1, no.5, pp. 766- 774, Nov 1983.
- [15] M.K. Simon, S. M. Hinedi, W.C. Lindsey, "Differentially Coherent Communication with Waveforms," in *Digital Communication Techniques*, Englewood Cliffs, NJ, PTR Prentice Hall, 1995, ch. 7, sec. 7.1.2-7.2.3, pp. 446-477.
- [16] M.A. Marsan, et al, *Modeling with Generalized Stochastic Petri Nets*, (Ch 1-5); Wiley, 1995.
- [17] C.A. Petri, "Communication with Automata," thesis, Darmstadt Institute of Technology, Bonn, Germany, 1962.
- [18] V. Mainkar and K. Trivedi, "Fixed Point Iteration Using Stochastic Reward Nets," presented at Proceedings of the Sixth International Workshop on Petri Nets and Performance Models, Los Alamitos, CA 1995.
- [19] Mondal, S. (2002), "Evaluation of a State Truncation Method in GSPN Dependability Modeling," Masters Thesis for Michigan Technological University.
- [20] J.B. Dugan, "Fault Trees and Imperfect Coverage," *IEEE Transactions on Reliability*, vol. 38, pp. 177-185, 1989.
- [21] J. Muppala, C. Lin, "Dependability Analysis of Large-Scale Distributed Systems Using Stochastic Petri Nets," presented at IEEE International Conference on Systems, Man and Cybernatics, 1996.
- [22] R.M. Kieckhafer, for the Pierre Auger Collaboration, "Design of a Real-Time Wireless Network for the Northern Pierre Auger Observatory", *IEEE Trans. Nuclear Science*, vol. 58, no. 4, Aug 2011, pp. 1784-1792.
- [23] Kramm, F.; "Field Experiences with VRLA-gel-batteries in Stationary Applications Over a Period of More than 20 Years," *Telecommunications Energy Conference*, 2001. INTELEC 2001. Twenty-Third International, pp 603-607, 18-18 Oct. 2001.
- [24] *Sunsaver Operator's Manual*, Morningstar Corp, (undated).
- [25] *Solar Module SM50-HData Sheet*, Siemens Solar GmbH, (undated).
- [26] R.E. Walpole, et al, "Probability & Statistics for Engineers & Scientists", Pearson, 2007, Section 5.3.

[27] J. Hollenbeck, R. Kieckhafer, P. Mendez, D. Nitz, A. Palmgren, F. Sarazin, M. Weiss, O. Wolf, "Report of the RF Survey of the Auger North RDA Site," GAP Note 2008-113, 2008.

[28] Maxim, *MAX2822 Datasheet*; Maxim Integrated Product, 2003.  
<<http://www.maxim-ic.com/datasheet/index.mvp/id/3938>>

[29] j. Zyren, A. Petrick, "Tutorial on Basic Link Budget Analysis," Intersil; Application Note AN9804.1, 1998.

[30] A. Jow, C. Schurgers, D. Palmer, "CalRadio: A Portable, Flexible 802.11 Wireless Research Platform," in *1<sup>st</sup> International Workshop on System Evaluation for Mobile Platforms*, Jun, 2007.

[31] A.Lackpour, M. Kavehrad and S. Thompson, "Architecture and Predicted Performance of an IEEE 802.11b-like WMAN Transceiver at 5.8GHz," Proceedings of the Third IEEE Workshop on Wireless Local Area Networks, Boston-Mass., September 2001.

[32] R. M. Kieckhafer, D. Nitz, P. Mishra, J. Tan, M. Weiss, F. Sarazin, "Application of the Wireless Infrastructure for High Assurance Real-Time Sensor Nets in Auger North," 2007.

[33] R.M. Kieckhafer, D. Nitz, A. Palmgren, "Auger North SD Communication System: Wireless Protocol Validation and Verification," GAP Note 2008-162, 2008.

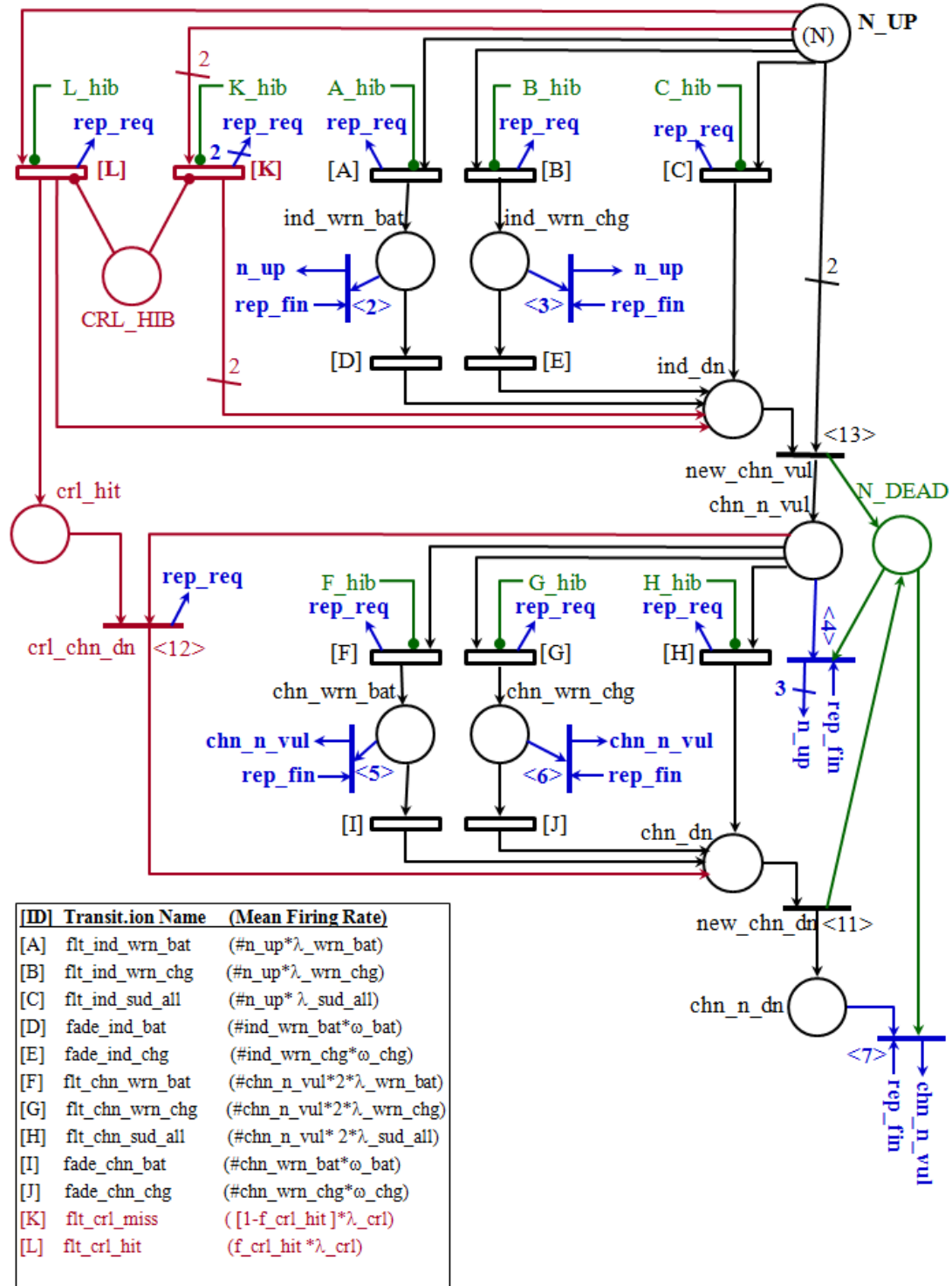
[34] M. Kleifges, H. Gemmeke, "UHE Cosmic Ray Detection With the Pierre Auger Observatory," *IEEE Transactions on Nuclear Science*, Vol. 52, No. 6, 2005.

[35] K.S. Trivedi, *SPNP User's Manual*, Ver. 6.0, Center for Advanced Computing (CACC), Duke University, Sep. 1999.

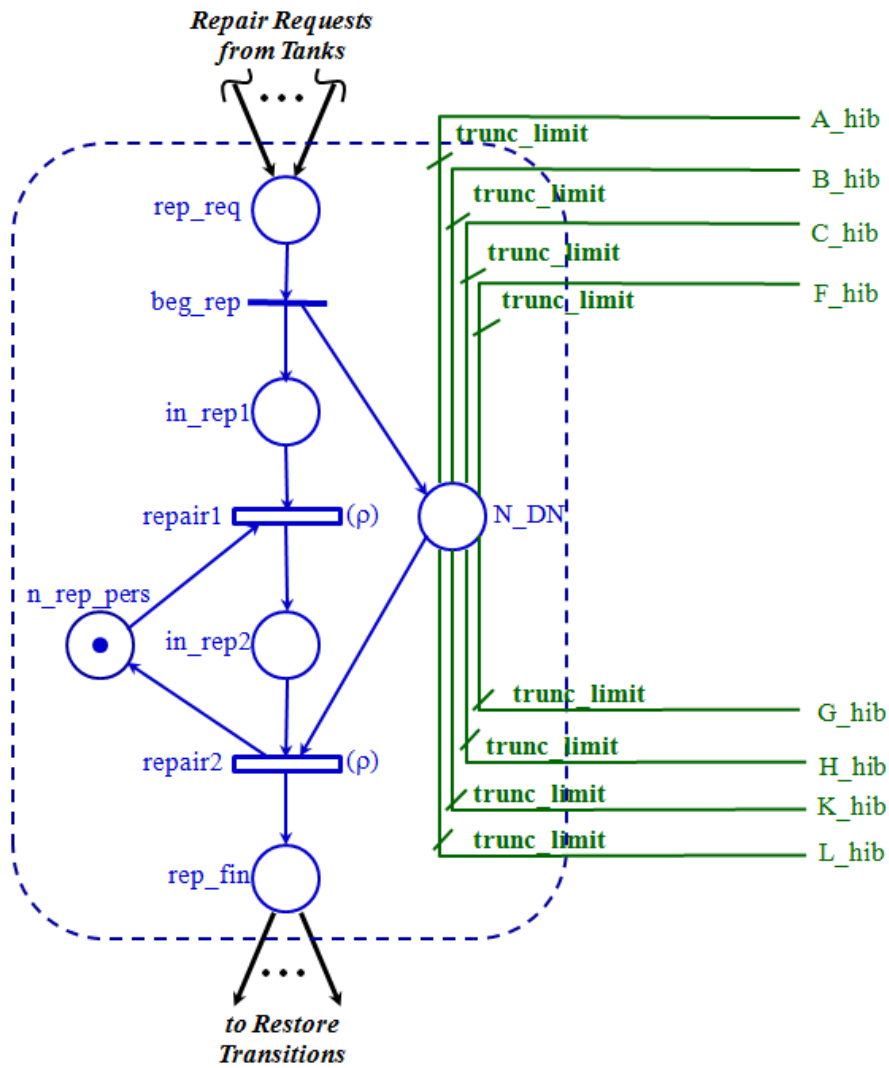
[36] Shannon, C.E. "Communication in the presence of noise." *Proceedings of the IRE*, vol. 37 no.1 pp 10-21, 19-49.

# Appendix A – GSPN

## A.a GSPN, Sector Fault Model – with Correlated Tank Faults



## A.b GSPN, Sector Repair & Truncation Subnet



## Appendix B – SPNP Code

```
/* ##### */
#include "user.h"
char AUTHOR[] = "R.M. Kieckhafer & M. Maskevics";
char DESCRIP[] = "Auger WAHREN Sector Avail - w/ rerouting option";
char REP_DST[] = "Erlang Repair - 2 stages - 1 annual phase";
char FLT_MOD[] = "b = Benign faults";
char FLT_DEL[] = "w = Warning given on some fract of faults";
char FLT_DEP[] = "c = Correlated (crl) Double Faults Allowed";
/* 20-Sep-2010 MM U-sect-bwc6 Original version. */
/* Model derived from Q-sect-bwc6. */
/* Modified to model unavailability rather than */
/* unreliability. */
/* ----- */
/* Variable definitions */
int N; /* total num of nodes per sector */
int route_red; /* route redundancy (1 = no re-routing) */
int trunc_limit = 8; /* marking of N_DN to truncate PNet */
int rep_pers; /* number of parallel repair teams */
int hib_crl; /* >=1 : inhibit correlated faults */
double f_wrn_bat, f_wrn_chg; /* flt mode frac */
double t_wrn_bat, t_wrn_chg, omega_bat, omega_chg; /* warning durs */
double lam_bat, lam_chg, lam_elx; /* phys lambdas */
double lam_wrn_bat, lam_wrn_chg, lam_sud_all; /* funct lambdas */
double ann_crl, lam_crl, f_crl_hit ; /* correl flts */
double mu, rho; /* rep rate, & per-stage rep rate */
double MTTR = 4; /* MTTR */

/* ----- */
void options()
{
    pr_message("\nEXECUTED ON: ");
    pr_message(__DATE__);
    pr_message(" ");pr_message(__TIME__);
    pr_message("\nAUTHOR NAME: ");pr_message(AUTHOR);
    pr_message("\nINPUT FILE: ");pr_message(__FILE__);
    pr_message("\n");
    pr_message("\nDESCRIPTION: ");pr_message(DESCRIP);
    pr_message("\n REP DST: ");pr_message(REP_DST);
    pr_message("\n FLT MOD: ");pr_message(FLT_MOD);
    pr_message("\n FLT DEL: ");pr_message(FLT_DEL);
    pr_message("\n FLT DEP: ");pr_message(FLT_DEP);
    pr_message("\n=====");
    pr_message("\n");
    /* Alter iopts and fopts as needed */
    /* pick transient solution method */
    iopt(IOP_TSMETHOD, VAL_TSUNIF);
    /* allow absorbing states */
    iopt(IOP_OK_ABSMARK, VAL_NO);
    /* set convergence criterion */
    fopt(FOP_PRECISION, 1.0E-6);
    /* set max number of iterations */
    iopt(IOP_ITERATIONS, 500000);
}
```



```

N = 128; /* Number of tanks per sector */
rep_pers = 1; /* only 1 technician per sector*/
f_wrn_bat = 0.9; /* frac of batt flts that give warning*/
f_wrn_chg = 0.9; /* frac of char flts that give warning*/
t_wrn_bat = 30.0; /* mean batt warning time, days */
t_wrn_chg = 10.0; /* mean char warning time, days */
omega_bat = 1.0 / (24.0 * t_wrn_bat);
omega_chg = 1.0 / (24.0 * t_wrn_chg);
lam_bat = 5.0e-4 / 8765.0; /* Battery physical lambda */
lam_chg = 4.0e-4 / 8765.0; /* Charger physical lambda */
lam_elx = 1.0e-6; /* Electronics physical lambda*/
lam_wrn_bat = lam_bat * f_wrn_bat;
lam_wrn_chg = lam_chg * f_wrn_chg;
lam_sud_all = lam_elx+(lam_bat-lam_wrn_bat)+(lam_chg-lam_wrn_chg);
ann_crl = 1.0; /*default placeholder*/
route_red = input (" Re-routing redundancy (1 or 2) ");
hib_crl = input (" Inhibit correlated faults? (1=yes, 0=no) ");
if (hib_crl == 0) {
    ann_crl = finput(" Num of correlated events per array-year ");}
lam_crl = ann_crl / (8765.0 * 32.0);
f_crl_hit = 3.0 / 8.0;
mu = 1.0 / (24.0*MTTR); /* total repair rate */
rho = 2.0*mu; /* firing rate for each Erlang stage */
}

/* ----- */
void net()
{
/* ----- Places ----- SYSTEM-LEVEL */
place ("N_UP"); init("N_UP", N);
place ("N_DN");
place ("N_DEAD");
/* ----- Places ----- REPAIR SUBNET */
place ("rep_req");
place ("in_rep1");
place ("in_rep2");
place ("rep_fin");
place ("n_rep_pers"); init("n_rep_pers", rep_pers);
/* ----- Places ----- INDEPENDENT FAULTS */
place ("ind_wrn_bat");
place ("ind_wrn_chg");
place ("ind_dn");
place ("chn_n_vul");
place ("chn_wrn_bat");
place ("chn_wrn_chg");
place ("chn_dn");
place ("chn_n_dn");
/* ----- Places ----- CORRELATED FAULTS */
place ("CRL_HIB"); init("CRL_HIB", hib_crl);
place ("crl_hit");
/* ----- Transitions ----- REPAIR SUBNET */
rateval ("stg1_rep", rho);
iarc ("stg1_rep", "in_rep1");
iarc ("stg1_rep", "n_rep_pers");

```

```

    oarc ("stg1_rep", "in_rep2");
rateval ("stg2_rep", rho);
    iarc ("stg2_rep", "in_rep2");
    iarc ("stg2_rep", "N_DN");
    oarc ("stg2_rep", "n_rep_pers");
    oarc ("stg2_rep", "rep_fin");
imm ("beg_rep");
    iarc ("beg_rep", "rep_req");
    oarc ("beg_rep", "in_repl");
    oarc ("beg_rep", "N_DN");
imm ("RST_ind_wrn_bat"); priority("RST_ind_wrn_bat",2);
    iarc ("RST_ind_wrn_bat", "rep_fin");
    iarc ("RST_ind_wrn_bat", "ind_wrn_bat");
    oarc ("RST_ind_wrn_bat", "N_UP");
imm ("RST_ind_wrn_chg"); priority("RST_ind_wrn_chg",3);
    iarc ("RST_ind_wrn_chg", "rep_fin");
    iarc ("RST_ind_wrn_chg", "ind_wrn_chg");
    oarc ("RST_ind_wrn_chg", "N_UP");
imm ("RST_chn_n_vul"); priority("RST_chn_n_vul",4);
    iarc ("RST_chn_n_vul", "rep_fin");
    iarc ("RST_chn_n_vul", "chn_n_vul");
    iarc ("RST_chn_n_vul", "N_DEAD");
    moarc ("RST_chn_n_vul", "N_UP", 3);
imm ("RST_chn_wrn_bat"); priority("RST_chn_wrn_bat",5);
    iarc ("RST_chn_wrn_bat", "rep_fin");
    iarc ("RST_chn_wrn_bat", "chn_wrn_bat");
    oarc ("RST_chn_wrn_bat", "chn_n_vul");
imm ("RST_chn_wrn_chg"); priority("RST_chn_wrn_chg",6);
    iarc ("RST_chn_wrn_chg", "rep_fin");
    iarc ("RST_chn_wrn_chg", "chn_wrn_chg");
    oarc ("RST_chn_wrn_chg", "chn_n_vul");
imm ("RST_chn_n_dn"); priority("RST_chn_n_dn",7);
    iarc ("RST_chn_n_dn", "rep_fin");
    iarc ("RST_chn_n_dn", "chn_n_dn");
    iarc ("RST_chn_n_dn", "N_DEAD");
    oarc ("RST_chn_n_dn", "chn_n_vul");
/* ----- Transitions ----- INDEPENDENT NODES FAULTS */
ratedep ("flt_ind_wrn_bat", lam_wrn_bat, "N_UP");
    mharc ("flt_ind_wrn_bat", "N_DN", trunc_limit);
    iarc ("flt_ind_wrn_bat", "N_UP");
    oarc ("flt_ind_wrn_bat", "ind_wrn_bat");
    oarc ("flt_ind_wrn_bat", "rep_req");
ratedep ("flt_ind_wrn_chg", lam_wrn_chg, "N_UP");
    mharc ("flt_ind_wrn_chg", "N_DN", trunc_limit);
    iarc ("flt_ind_wrn_chg", "N_UP");
    oarc ("flt_ind_wrn_chg", "ind_wrn_chg");
    oarc ("flt_ind_wrn_chg", "rep_req");
ratedep ("flt_ind_sud_all", lam_sud_all, "N_UP");
    mharc ("flt_ind_sud_all", "N_DN", trunc_limit);
    iarc ("flt_ind_sud_all", "N_UP");
    oarc ("flt_ind_sud_all", "ind_dn");
    oarc ("flt_ind_sud_all", "rep_req");
ratedep ("fade_ind_bat", omega_bat, "ind_wrn_bat");
    iarc ("fade_ind_bat", "ind_wrn_bat");

```

```

    oarc ("fade_ind_bat", "ind_dn");
ratedep ("fade_ind_chg", omega_chg, "ind_wrn_chg");
    iarc ("fade_ind_chg", "ind_wrn_chg");
    oarc ("fade_ind_chg", "ind_dn");
imm ("new_chn_vul"); priority("new_chn_vul",13);
    iarc ("new_chn_vul", "ind_dn");
    miarc ("new_chn_vul", "N_UP", 2);
    oarc ("new_chn_vul", "chn_n_vul");
    oarc ("new_chn_vul", "N_DEAD");
/* ----- Transitions ----- INDEPENDENT CHAIN FAULTS*/
ratedep ("flt_chn_wrn_bat", lam_wrn_bat, "chn_n_vul");
    mharc ("flt_chn_wrn_bat", "N_DN", trunc_limit);
    iarc ("flt_chn_wrn_bat", "chn_n_vul");
    oarc ("flt_chn_wrn_bat", "chn_wrn_bat");
    oarc ("flt_chn_wrn_bat", "rep_req");
ratedep ("flt_chn_wrn_chg", lam_wrn_chg, "chn_n_vul");
    mharc ("flt_chn_wrn_chg", "N_DN", trunc_limit);
    iarc ("flt_chn_wrn_chg", "chn_n_vul");
    oarc ("flt_chn_wrn_chg", "chn_wrn_chg");
    oarc ("flt_chn_wrn_chg", "rep_req");
ratedep ("flt_chn_sud_all", lam_sud_all, "chn_n_vul");
    mharc ("flt_chn_sud_all", "N_DN", trunc_limit);
    iarc ("flt_chn_sud_all", "chn_n_vul");
    oarc ("flt_chn_sud_all", "chn_dn");
    oarc ("flt_chn_sud_all", "rep_req");
ratedep ("fade_chn_bat", omega_bat, "chn_wrn_bat");
    iarc ("fade_chn_bat", "chn_wrn_bat");
    oarc ("fade_chn_bat", "chn_dn");
ratedep ("fade_chn_chg", omega_chg, "chn_wrn_chg");
    iarc ("fade_chn_chg", "chn_wrn_chg");
    oarc ("fade_chn_chg", "chn_dn");
imm ("new_chn_dn"); priority("new_chn_dn",11);
    iarc ("new_chn_dn", "chn_dn");
    oarc ("new_chn_dn", "chn_n_dn");
    oarc ("new_chn_dn", "N_DEAD");
/* ----- Transitions ----- CORRELATED FAULTS*/
rateval ("flt_crl_miss", (1-f_crl_hit)*lam_crl);
    harc ("flt_crl_miss", "CRL_HIB");
    mharc ("flt_crl_miss", "N_DN", trunc_limit);
    miarc ("flt_crl_miss", "N_UP", 2);
    moarc ("flt_crl_miss", "rep_req", 2);
    moarc ("flt_crl_miss", "ind_dn", 2);
rateval ("flt_crl_hit", f_crl_hit*lam_crl);
    harc ("flt_crl_hit", "CRL_HIB");
    mharc ("flt_crl_hit", "N_DN", trunc_limit);
    iarc ("flt_crl_hit", "N_UP");
    oarc ("flt_crl_hit", "rep_req");
    oarc ("flt_crl_hit", "crl_hit");
    oarc ("flt_crl_hit", "ind_dn");
imm ("crl_chn_dn"); priority("crl_chn_dn",12);
    iarc ("crl_chn_dn", "crl_hit");
    iarc ("crl_chn_dn", "chn_n_vul");
    oarc ("crl_chn_dn", "rep_req");
    oarc ("crl_chn_dn", "chn_dn");

```

```

}

/* ----- */
int assert()
{
    if ( (mark("N_UP") + mark("N_DN") > N )
        || (mark("N_DN") > (mark("in_rep1") + mark("in_rep2"))))
        return(RES_ERROR);
    else
        return(RES_NOERR);
}

/* ----- */
void ac_init()
{
    fprintf(stderr, "\nSTART AC_INIT #####\n");
}

/* ----- */
void ac_reach()
{
    fprintf(stderr, "\nSTART AC_REACH #####\n");
}

/* ----- */
/* variables */
double mission_time, Mean_dn, Q_upper, Q_lower, Rel_Error;
double P_num_dn, P_zero_dn, P_one_dn, P_two_dn;
double P_three_dn, P_four_dn;
double P_any_dn, P_mor_dn;
double P_gr_1_dn, P_gr_2_dn, P_gr_3_dn, P_gr_4_dn, P_gr_5_dn;
double P_gr_6_dn, P_gr_7_dn, P_gr_8_dn;
double U_lower_exp, U_upper_exp;
double U_Upper_test;
double Trunc_Prob() {return(mark("N_DN")>=trunc_limit);}
double U_Lower() {return(mark("chn_n_dn")>=route_red);}
double U_Upper() {return(mark("N_DN")>=trunc_limit)
    ||(mark("chn_n_dn")>=route_red);}
double Num_dn() {return(mark("N_DN"));}
double Zero_dn() {return(mark("N_DN")==0);}
double One_dn() {return(mark("N_DN")==1);}
double Two_dn() {return(mark("N_DN")==2);}
double Three_dn() {return(mark("N_DN")==3);}
double Four_dn() {return(mark("N_DN")==4);}
double Gr_1_dn() {return(mark("N_DN")>=1);}
double Gr_2_dn() {return(mark("N_DN")>=2);}
double Gr_3_dn() {return(mark("N_DN")>=3);}
double Gr_4_dn() {return(mark("N_DN")>=4);}
double Gr_5_dn() {return(mark("N_DN")>=5);}
double Gr_6_dn() {return(mark("N_DN")>=6);}
double Gr_7_dn() {return(mark("N_DN")>=7);}
double Gr_8_dn() {return(mark("N_DN")>=8);}
double Any_dn() {return(mark("N_DN")>0);}

```

```

void ac_final()
{
    fprintf(stderr, "\nSTART AC_FINAL #####\n");
    mission_time = 100.0 * 8765.0;
    solve(mission_time);

    /* CALCULATE UNAVAILABILITY */

    /* Pr(system was up at time t) */
    /* Fraction of time system is up during this run */
    U_lower_exp = expected(U_Lower);
    U_upper_exp = expected(U_Upper);
    Rel_Error = ((U_upper_exp - U_lower_exp) / U_lower_exp);

    /* CALCULATE EXPECTED DN'S */
    P_num_dn = expected(Num_dn);
    P_zero_dn = expected(Zero_dn);
    P_one_dn = expected(One_dn);
    P_two_dn = expected(Two_dn);
    P_three_dn = expected(Three_dn);
    P_four_dn = expected(Four_dn);
    P_any_dn = expected(Any_dn);
    P_gr_1_dn = expected(Gr_1_dn);
    P_gr_2_dn = expected(Gr_2_dn);
    P_gr_3_dn = expected(Gr_3_dn);
    P_gr_4_dn = expected(Gr_4_dn);
    P_gr_5_dn = expected(Gr_5_dn);
    P_gr_6_dn = expected(Gr_6_dn);
    P_gr_7_dn = expected(Gr_7_dn);
    P_gr_8_dn = expected(Gr_8_dn);

    /* Sanity Check */
    U_Upper_test = (expected(U_Lower) + expected(Trunc_Prob));
    pr_message ("\nSTATS: _____ FOR ONE SECTOR");
    pr_message ("\n ");
    pr_value("___ Trunc Limit (#N_DN) ", trunc_limit);
    pr_value("___ UnAvail Relative Error ", Rel_Error);
    pr_message ("\n ");
    pr_value("___ UnAvail Lower Bound U(t)", U_lower_exp );
    pr_value("___ UnAvail Upper Bound U(t)", U_upper_exp );
    pr_message ("\n ");
    pr_value("___ Upper bound test (should be higher than the upper
        bound:",
        U_Upper_test);
    pr_message ("\n ");
    pr_message ("\n ");
    pr_value("___ Expected Num_dn ", P_num_dn);
    pr_value("___ Expected Zero_dn ", P_zero_dn);
    pr_value("___ Expected One_dn ", P_one_dn);
    pr_value("___ Expected Two_dn ", P_two_dn);
    pr_value("___ Expected Three_dn ", P_three_dn);
    pr_value("___ Expected Four_dn ", P_four_dn);
    pr_value("___ Expected >= 1_dn ", P_gr_1_dn);
    pr_value("___ Expected >= 2_dn ", P_gr_2_dn);

```

```

pr_value("__ Expected >= 3_dn ", P_gr_3_dn);
pr_value("__ Expected >= 4_dn ", P_gr_4_dn);
pr_value("__ Expected >= 5_dn ", P_gr_5_dn);
pr_value("__ Expected >= 6_dn ", P_gr_6_dn);
pr_value("__ Expected >= 7_dn ", P_gr_7_dn);
pr_value("__ Expected >= 8_dn ", P_gr_8_dn);
pr_value("__ Expected Any_dn ", P_any_dn);

pr_message("\n=====");
pr_message("\nINPUT FILE : ");pr_message(__FILE__);
pr_message("\nEXECUTED ON: ");
pr_message(__DATE__);pr_message(" ");pr_message(__TIME__);
pr_message("\n");
}

```

### Appendix C – GSPN Results, $F(f)$

$f$	$F(f)$
0	9.8577E-01
1	1.3741E-02
2	4.8157E-04
3	8.3417E-06
4+	1.9908E-07

## Appendix D – C-Program, Exhaustive Search, $C_{if}(T)$

```
// Evaluation of Auger WAHREN tank faults.
// Probability the fault is in a tanks path.

#include <stdlib.h>
#include <sys/types.h>
#include <stdio.h>
#include <math.h>
#include <stdlib.h>

//Definitions
#define ALGO1
//#define ALGO2

// Constants
int N_TANKS = 130;
int TRUE = 1;
int FALSE = 0;
int N_SIDECHAINS = 10;

// set up tank structure
typedef struct {
    int a; // tank location on the a axis
    int b; // tank location on the b axis
    int d; // tank depth
    int h; // number of hops to get to CR
}Tank_T;

// Arrays
Tank_T tanks[130]; // sector of tanks, index 0-129 is the tank id

// Prototypes of other functions used in this program
int IsBackbone(int);
int IsEven(int);

// This is the main function of the program. It determines
// the probability a fault is in a tank's path.
int main()
{
    // Variables
    int i, j, tt, row_print; // used to iterate through loops
    int sc; // used to iterate through sidechains
    int source_t; // tank being evaluated
    int f1, f2, f3, f4; // fault number
    int a_temp = 1; // used to calculate a location of tank
    int b_temp = 1; // used to calculate b location of tank
    int n_tank_sidechain = 4; // init # of tanks in 1st sidechain
    int temp_tank_id = 2; // tank # of 1st tank in sidechain
    int def_error_check = 0; // check for an error in #defines
    int total_count, fault_count; // counters for tank faults

    // Arrays
    int inpath[130][130]; // tracks if a tank is in the path of another
```



```

int neighbor[130][130]; // tracks which tanks are neighbors
double f1_inpath[130][2]; // tracks fault probability of my path
// when there is 1 fault in they sector
// # patterns in my path / total patterns
// column 0 = prob no fault is in my path
// column 1 = prob 1 fault is in my path
double f2_inpath[130][3]; // tracks fault prob of my path
// when there are 2 faults in my sector
// # patterns in my path / total patterns
// column 0 = prob no fault is in my path
// column 1 = prob 1 fault is in my path
// column 2 = prob 2 faults is in my path
double f3_inpath[130][4]; // tracks fault prob of my path
// when there are 3 faults in my sector
// # patterns in my path / total patterns
// column 0 = prob no fault is in my path
// column 1 = prob 1 fault is in my path
// column 2 = prob 2 faults is in my path
// column 3 = prob 3 faults is in my path
double f4_inpath[130][5]; // tracks fault prob of my path
// when there are 4 faults in my sector
// # patterns in my path / total patterns
// column 0 = prob no fault is in my path
// column 1 = prob 1 fault is in my path
// column 2 = prob 2 faults is in my path
// column 3 = prob 3 faults is in my path
// column 4 = prob 4 faults is in my path

// set inpath and neighbor arrays to all FALSE values
for (i = 0; i < N_TANKS; i++) {
    f1_inpath[i][0] = FALSE;
    f1_inpath[i][1] = FALSE;
    f2_inpath[i][0] = FALSE;
    f2_inpath[i][1] = FALSE;
    f2_inpath[i][2] = FALSE;
    f3_inpath[i][0] = FALSE;
    f3_inpath[i][1] = FALSE;
    f3_inpath[i][2] = FALSE;
    f3_inpath[i][3] = FALSE;
    f4_inpath[i][0] = FALSE;
    f4_inpath[i][1] = FALSE;
    f4_inpath[i][2] = FALSE;
    f4_inpath[i][3] = FALSE;
    f4_inpath[i][4] = FALSE;
    for (j = 0; j < N_TANKS; j++) {
        inpath[i][j] = FALSE;
        neighbor[i][j] = FALSE;
    }
}

// Verify exactly one algorithm is defined
#ifdef ALG01
printf("ALG01 \n");
def_error_check++;

```

```

#endif
#ifdef ALGO2
printf("ALGO2 \n");
def_error_check++;
#endif
if (def_error_check != 1){
    printf("ERROR IN DEFINITIONS \n");
    return 0;
}

// Set up concentrator radios
tanks[0].a = 0;
tanks[0].b = 1;
tanks[0].d = 0;
tanks[0].h = 0;
tanks[1].a = 0;
tanks[1].b = 2;
tanks[1].d = 1;
tanks[1].h = 0;
inpath[0][0] = TRUE;
inpath[1][0] = TRUE;
inpath[1][1] = TRUE;
neighbor[0][0] = TRUE;
neighbor[0][1] = TRUE;
neighbor[1][0] = TRUE;
neighbor[1][1] = TRUE;

// Set up entire sectors tank's parameters =====
for (sc = 1; sc <= N_SIDECHAINS; sc++) {
    for (source_t = temp_tank_id;
        source_t < (temp_tank_id+n_tank_sidechain); source_t++) {
        // determine tank's a location -----
        if (IsEven(source_t) == FALSE) {
            tanks[source_t].a = a_temp;
        }
        else{
            tanks[source_t].a = a_temp + 1;
        }
        // determine tank's b location -----
        tanks[source_t].b = b_temp;
        b_temp++;
        // determine tank depth -----
        tanks[source_t].d = (tanks[source_t].a+1) +
            (((tanks[source_t].b-1)>>1)<<1);
        // determine tank's number of hops to CR -----
        tanks[source_t].h = tanks[source_t].d -1;
        // determine tank's inpath values -----
        // include tanks in my sidechain
        for(i = tanks[source_t].b; i > 0; i--){
            inpath[source_t][temp_tank_id+(i-1)] = TRUE;
        }
        // include tanks in the backbone
        for (i = 0; i <= source_t; i++) {
            if (IsBackbone(i) == TRUE) {

```

```

        inpath[source_t][i] = TRUE;
    }
}
// determine tank's neighbors -----
// set my value to TRUE
neighbor[source_t][source_t]= TRUE;
// if not end tank in sidechain, include next tank id
if(tanks[source_t].b != n_tank_sidechain)
    neighbor[source_t][source_t+1] = TRUE;
// if 2nd tank in this sidechain,
// include 1st tank in next sidechain
if (tanks[source_t].b == 2){
    // include 1st tank in next row (unless in last sidechain)
    if(sc!=10)
        neighbor[source_t][source_t+(n_tank_sidechain-1)]=TRUE;
}
// if 1st tank in this sidechain,
// include 2nd tank in prev sidechain
if (tanks[source_t].b == 1){
    if(sc==10)
        neighbor[source_t][source_t-(n_tank_sidechain)+1]=TRUE;
    else
        neighbor[source_t][source_t-(n_tank_sidechain-2)+1]=TRUE;
}
// otherwise include tank with one lower tank id
else neighbor[source_t][source_t-1] = TRUE;
}
// set up variables for next loop
printf("Row number %i\n", sc);
temp_tank_id = source_t;
a_temp += 2;
b_temp = 1;
// each sidechain has two tanks more than the last,
// except for the last sidechain
if (sc < (N_SIDECHAINS-1)) n_tank_sidechain +=2;
}

// Evaluation: One Faulty tank in the sector =====
#ifdef ALGO1
for (source_t = 2; source_t < N_TANKS; source_t++) {
    total_count = 0;
    // count the number of fault patterns in my path
    for (f1 = 2; f1 < N_TANKS; f1++) {
        // don't count if the source tank is the faulty tank
        if (source_t == f1) continue;
        total_count++;
        if (inpath[source_t][f1] == TRUE )
            f1_inpath[source_t][1] = f1_inpath[source_t][1] + 1;
        else f1_inpath[source_t][0] = f1_inpath[source_t][0] + 1;
    }
    // divide number of faults patterns in my path by total patterns
    f1_inpath[source_t][0] = f1_inpath[source_t][0] / total_count;
    f1_inpath[source_t][1] = f1_inpath[source_t][1] / total_count;
}
}

```

```

#endif

#ifdef ALGO2
for (f1 = 2; f1 < (N_TANKS); f1++){
    for (source_t = 2; source_t < N_TANKS; source_t++){
        if (source_t == f1) continue;
        fault_count = 0;
        if (inpath[source_t][f1] == TRUE) fault_count++;
        f1_inpath[source_t][fault_count] =
            f1_inpath[source_t][fault_count]+1;
    }
}
// Divide by total count, calculate tank's Cf
for (source_t = 2; source_t < N_TANKS; source_t++){
    total_count = f1_inpath[source_t][0] + f1_inpath[source_t][1];
    f1_inpath[source_t][0] = f1_inpath[source_t][0] / total_count;
    f1_inpath[source_t][1] = f1_inpath[source_t][1] / total_count;
}
#endif

// Evaluation: Two Faulty tanks in the sector =====
#ifdef ALGO1
for (source_t = 2; source_t < N_TANKS; source_t++) {
    total_count = 0;
    // count the number of fault patterns in my path
    for (f1 = 2; f1 < N_TANKS; f1++) {
        // don't count if the source tank is the faulty tank
        if (source_t == f1) continue;
        for (f2 = f1+1; f2 < N_TANKS; f2++) {
            // don't count if the source tank is the faulty tank
            if (source_t == f2) continue;
            // don't count it there are two faults next to each other
            // (system failure)
            if (neighbor[f1][f2] == TRUE) continue;
            // set up variables
            fault_count = 0;
            total_count++;
            // count the number of faults in my path
            if (inpath[source_t][f1] == TRUE) fault_count++;
            if (inpath[source_t][f2] == TRUE) fault_count++;
            // record number of faults in f2_inpath array
            if (fault_count == 0)
                f2_inpath[source_t][0] = f2_inpath[source_t][0] + 1;
            else if (fault_count == 1)
                f2_inpath[source_t][1] = f2_inpath[source_t][1] + 1;
            else if (fault_count == 2)
                f2_inpath[source_t][2] = f2_inpath[source_t][2] + 1;
            else printf("ERROR IN 2 FAULT LOOP \n");
        }
    }
}
// divide number of faults patterns in my path by total patterns
f2_inpath[source_t][0] = f2_inpath[source_t][0] / total_count;
f2_inpath[source_t][1] = f2_inpath[source_t][1] / total_count;
f2_inpath[source_t][2] = f2_inpath[source_t][2] / total_count;

```

```

}
#endif

#ifdef ALGO2
for (f1 = 2; f1 < (N_TANKS - 1); f1++){
  for (f2 = f1+1; f2 < N_TANKS; f2++){
    if (neighbor[f1][f2] == TRUE) continue;
    for (source_t = 2; source_t < N_TANKS; source_t++){
      if ((source_t == f1) || (source_t == f2)) continue;
      fault_count = 0;
      if (inpath[source_t][f1] == TRUE) fault_count++;
      if (inpath[source_t][f2] == TRUE) fault_count++;
      f2_inpath[source_t][fault_count] =
        f2_inpath[source_t][fault_count]+1;
    }
  }
}
// Divide by total count, calculate tank's Cf
for (source_t = 2; source_t < N_TANKS; source_t++){
  total_count = f2_inpath[source_t][0] + f2_inpath[source_t][1]
    + f2_inpath[source_t][2];
  f2_inpath[source_t][0] = f2_inpath[source_t][0] / total_count;
  f2_inpath[source_t][1] = f2_inpath[source_t][1] / total_count;
  f2_inpath[source_t][2] = f2_inpath[source_t][2] / total_count;
}
#endif

// Evaluation: Three Faulty tanks in the sector =====
#ifdef ALGO1
for (source_t = 2; source_t < N_TANKS; source_t++) {
  total_count = 0;
  // count the number of fault patterns in my path
  for (f1 = 2; f1 < N_TANKS; f1++) {
    // don't count if the source tank is the faulty tank
    if (source_t == f1) continue;
    for (f2 = f1+1; f2 < N_TANKS; f2++) {
      // don't count if the source tank is the faulty tank
      if (source_t == f2) continue;
      for(f3 = f2+1; f3 < N_TANKS; f3++){
        // don't count if the source tank is the faulty tank
        if (source_t == f3) continue;
        // don't count it there are two faults next to each other
        // (system failure)
        if ((neighbor[f1][f2] == TRUE) | (neighbor[f2][f3] == TRUE)
          | (neighbor[f1][f3] == TRUE)) continue;
      }
      // set up variables
      fault_count = 0;
      total_count++;
      // count the number of faults in my path
      if (inpath[source_t][f1] == TRUE) fault_count++;
      if (inpath[source_t][f2] == TRUE) fault_count++;
      if (inpath[source_t][f3] == TRUE) fault_count++;
      // record number of faults in f3_inpath array
      if (fault_count == 0)

```

```

        f3_inpath[source_t][0] = f3_inpath[source_t][0] + 1;
    else if (fault_count == 1)
        f3_inpath[source_t][1] = f3_inpath[source_t][1] + 1;
    else if (fault_count == 2)
        f3_inpath[source_t][2] = f3_inpath[source_t][2] + 1;
    else if (fault_count == 3)
        f3_inpath[source_t][3] = f3_inpath[source_t][3] + 1;
    else printf("ERROR IN 3 FAULT LOOP \n");
    }
}
}
// divide num of faults patterns in my path by total patterns
f3_inpath[source_t][0] = f3_inpath[source_t][0] / total_count;
f3_inpath[source_t][1] = f3_inpath[source_t][1] / total_count;
f3_inpath[source_t][2] = f3_inpath[source_t][2] / total_count;
f3_inpath[source_t][3] = f3_inpath[source_t][3] / total_count;
}
#endif

#ifdef ALGO2
for (f1 = 2; f1 < (N_TANKS - 2); f1++){
    for (f2 = f1+1; f2 < (N_TANKS - 1); f2++){
        for (f3 = f2+1; f3 < N_TANKS; f3++){
            if (neighbor[f1][f2] == TRUE) continue;
            if (neighbor[f2][f3] == TRUE) continue;
            if (neighbor[f1][f3] == TRUE) continue;
            for (source_t = 2; source_t < N_TANKS; source_t++){
                if ((source_t == f1) || (source_t == f2) ||
                    (source_t == f3)) continue;
                fault_count = 0;
                if (inpath[source_t][f1] == TRUE) fault_count++;
                if (inpath[source_t][f2] == TRUE) fault_count++;
                if (inpath[source_t][f3] == TRUE) fault_count++;
                f3_inpath[source_t][fault_count] =
                    f3_inpath[source_t][fault_count]+1;
            }
        }
    }
}
// Divide by total count, calculate tank's Cf
for (source_t = 2; source_t < N_TANKS; source_t++){
    total_count = f3_inpath[source_t][0] + f3_inpath[source_t][1]
        + f3_inpath[source_t][2] + f3_inpath[source_t][3];
    f3_inpath[source_t][0] = f3_inpath[source_t][0] / total_count;
    f3_inpath[source_t][1] = f3_inpath[source_t][1] / total_count;
    f3_inpath[source_t][2] = f3_inpath[source_t][2] / total_count;
    f3_inpath[source_t][3] = f3_inpath[source_t][3] / total_count;
}
#endif

// Evaluation: Four Faulty tanks in the sector =====
#ifdef ALGO1
for (source_t = 2; source_t < N_TANKS; source_t++) {
    total_count = 0;

```

```

// count the number of fault patterns in my path
for (f1 = 2; f1 < N_TANKS; f1++) {
    // don't count if the source tank is the faulty tank
    if (source_t == f1) continue;
    for (f2 = f1+1; f2 < N_TANKS; f2++) {
        // don't count if the source tank is the faulty tank
        if (source_t == f2) continue;
        for (f3 = f2+1; f3 < N_TANKS; f3++) {
            // don't count if the source tank is the faulty tank
            if (source_t == f3) continue;
            for (f4 = f3+1; f4 < N_TANKS; f4++) {
                // don't count if the source tank is the faulty tank
                if (source_t == f4) continue;
                // don't count it there are two faults next to
                // each other (system failure)
                if ((neighbor[f1][f2] == TRUE) |
                    (neighbor[f1][f3] == TRUE) |
                    (neighbor[f1][f4] == TRUE) |
                    (neighbor[f2][f3] == TRUE) |
                    (neighbor[f2][f4] == TRUE) |
                    (neighbor[f3][f4] == TRUE))
                    continue;
                // set up variables
                fault_count = 0;
                total_count++;
                // count the number of faults in my path
                if (inpath[source_t][f1] == TRUE) fault_count++;
                if (inpath[source_t][f2] == TRUE) fault_count++;
                if (inpath[source_t][f3] == TRUE) fault_count++;
                if (inpath[source_t][f4] == TRUE) fault_count++;
                // record number of faults in f4_inpath array
                if (fault_count == 0)
                    f4_inpath[source_t][0] = f4_inpath[source_t][0] + 1;
                else if (fault_count == 1)
                    f4_inpath[source_t][1] = f4_inpath[source_t][1] + 1;
                else if (fault_count == 2)
                    f4_inpath[source_t][2] = f4_inpath[source_t][2] + 1;
                else if (fault_count == 3)
                    f4_inpath[source_t][3] = f4_inpath[source_t][3] + 1;
                else if (fault_count == 4)
                    f4_inpath[source_t][4] = f4_inpath[source_t][4] + 1;
                else printf("ERROR IN 4 FAULT LOOP \n");
            }
        }
    }
}
// divide number of faults patterns in my path by total patterns
f4_inpath[source_t][0] = f4_inpath[source_t][0] / total_count;
f4_inpath[source_t][1] = f4_inpath[source_t][1] / total_count;
f4_inpath[source_t][2] = f4_inpath[source_t][2] / total_count;
f4_inpath[source_t][3] = f4_inpath[source_t][3] / total_count;
f4_inpath[source_t][4] = f4_inpath[source_t][4] / total_count;
}
#endif

```

```

#ifdef ALGO2
for (f1 = 2; f1 < (N_TANKS - 3); f1++){
  for (f2 = f1+1; f2 < (N_TANKS - 2); f2++){
    for (f3 = f2+1; f3 < (N_TANKS - 1); f3++){
      for (f4 = f3+1; f4 < N_TANKS; f4++){
        if (neighbor[f1][f2] == TRUE) continue;
        if (neighbor[f1][f3] == TRUE) continue;
        if (neighbor[f1][f4] == TRUE) continue;
        if (neighbor[f2][f3] == TRUE) continue;
        if (neighbor[f2][f4] == TRUE) continue;
        if (neighbor[f3][f4] == TRUE) continue;
        for (source_t = 2; source_t < N_TANKS; source_t++){
          if ((source_t == f1) || (source_t == f2) ||
              (source_t == f3) || (source_t == f4)) continue;
          fault_count = 0;
          if (inpath[source_t][f1] == TRUE) fault_count++;
          if (inpath[source_t][f2] == TRUE) fault_count++;
          if (inpath[source_t][f3] == TRUE) fault_count++;
          if (inpath[source_t][f4] == TRUE) fault_count++;
          f4_inpath[source_t][fault_count] =
            f4_inpath[source_t][fault_count]+1;
        }
      }
    }
  }
}
// Divide by total count, calculate tank's Cf
for (source_t = 2; source_t < N_TANKS; source_t++){
  total_count = f4_inpath[source_t][0] + f4_inpath[source_t][1]
    + f4_inpath[source_t][2] + f4_inpath[source_t][3]
    + f4_inpath[source_t][4];
  f4_inpath[source_t][0] = f4_inpath[source_t][0] / total_count;
  f4_inpath[source_t][1] = f4_inpath[source_t][1] / total_count;
  f4_inpath[source_t][2] = f4_inpath[source_t][2] / total_count;
  f4_inpath[source_t][3] = f4_inpath[source_t][3] / total_count;
  f4_inpath[source_t][4] = f4_inpath[source_t][4] / total_count;
}
#endif

// Output Data
printf("Prob 0 faults in tanks path\n");
for (source_t = 0; source_t < N_TANKS; source_t++){
  printf("Tank %i ");
  printf("%e %e %e %e \n", f1_inpath[source_t][0],
    f2_inpath[source_t][0], f3_inpath[source_t][0],
    f4_inpath[source_t][0]);
}
printf("Prob 1 fault in tanks path\n");
for (source_t = 0; source_t < N_TANKS; source_t++){
  printf("Tank %i ");
  printf("%e %e %e %e \n", f1_inpath[source_t][1],
    f2_inpath[source_t][1], f3_inpath[source_t][1],
    f4_inpath[source_t][1]);
}

```



```

}
printf("Prob 2 faults in tanks path\n");
for (source_t = 0; source_t < N_TANKS; source_t++){
    printf("Tank %i ");
    printf("0 %e %e %e \n",f2_inpath[source_t][2],
        f3_inpath[source_t][2], f4_inpath[source_t][2]);
}
printf("Prob 3 faults in tanks path\n");
for (source_t = 0; source_t < N_TANKS; source_t++){
    printf("Tank %i ");
    printf("0 0 %e %e \n",f3_inpath[source_t][3],
        f4_inpath[source_t][3]);
}
printf("Prob 4 faults in tanks path\n");
for (source_t = 0; source_t < N_TANKS; source_t++){
    printf("Tank %i ");
    printf("0 0 0 %e \n", f4_inpath[source_t][4]);
}
return 0;
}

// This function returns TRUE if b is a backbone tank
// else returns FALSE;
// Input int b = id of tank in question
// Output int = TRUE if backbone tank, otherwise FALSE
int IsBackbone(int b){
    if (tanks[b].b <= 2) return TRUE;
    else return FALSE;
}

// This function returns TRUE if e is an even number,
// else returns FALSE;
// Input int e = integer in question
// Output int = TRUE if even, otherwise FALSE
int IsEven(int e){
    if (e % 2 == 0) return FALSE;
    else return TRUE;
}

```

## Appendix E – Exhaustive Search Results, $C_{ij}(T)$

### E.a $C_{ij}(T)$ , $i=0$

Tank ID	$i = 0$ $f = 1$	$i = 0$ $f = 2$	$i = 0$ $f = 3$	$i = 0$ $f = 4$
0	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
1	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
2	1.0000E+00	1.0000E+00	1.0000E+00	1.0000E+00
3	9.9213E-01	9.8400E-01	9.7563E-01	9.6698E-01
4	9.8425E-01	9.6839E-01	9.5240E-01	9.3630E-01
5	9.7638E-01	9.5276E-01	9.2916E-01	9.0557E-01
6	9.8425E-01	9.6839E-01	9.5240E-01	9.3630E-01
7	9.7638E-01	9.5277E-01	9.2919E-01	9.0564E-01
8	9.6850E-01	9.3740E-01	9.0671E-01	8.7642E-01
9	9.6063E-01	9.2204E-01	8.8424E-01	8.4722E-01
10	9.5276E-01	9.0681E-01	8.6214E-01	8.1877E-01
11	9.4488E-01	8.9168E-01	8.4038E-01	7.9096E-01
12	9.6850E-01	9.3740E-01	9.0671E-01	8.7641E-01
13	9.6063E-01	9.2205E-01	8.8427E-01	8.4727E-01
14	9.5276E-01	9.0693E-01	8.6251E-01	8.1947E-01
15	9.4488E-01	8.9182E-01	8.4078E-01	7.9172E-01
16	9.3701E-01	8.7684E-01	8.1943E-01	7.6469E-01
17	9.2913E-01	8.6199E-01	7.9843E-01	7.3836E-01
18	9.2126E-01	8.4726E-01	7.7781E-01	7.1272E-01
19	9.1339E-01	8.3263E-01	7.5747E-01	6.8763E-01
20	9.5276E-01	9.0693E-01	8.6251E-01	8.1947E-01
21	9.4488E-01	8.9184E-01	8.4082E-01	7.9180E-01
22	9.3701E-01	8.7697E-01	8.1979E-01	7.6538E-01
23	9.2913E-01	8.6211E-01	7.9879E-01	7.3902E-01
24	9.2126E-01	8.4738E-01	7.7816E-01	7.1337E-01
25	9.1339E-01	8.3278E-01	7.5788E-01	6.8838E-01
26	9.0551E-01	8.1831E-01	7.3796E-01	6.6406E-01
27	8.9764E-01	8.0396E-01	7.1840E-01	6.4039E-01
28	8.8976E-01	7.8974E-01	6.9918E-01	6.1736E-01
29	8.8189E-01	7.7562E-01	6.8023E-01	5.9482E-01
30	9.3701E-01	8.7697E-01	8.1979E-01	7.6537E-01

$C_{if}(T), i=0$ (continued)				
31	9.2913E-01	8.6213E-01	7.9884E-01	7.3911E-01
32	9.2126E-01	8.4751E-01	7.7851E-01	7.1403E-01
33	9.1339E-01	8.3291E-01	7.5823E-01	6.8902E-01
34	9.0551E-01	8.1844E-01	7.3831E-01	6.6469E-01
35	8.9764E-01	8.0409E-01	7.1874E-01	6.4101E-01
36	8.8976E-01	7.8987E-01	6.9952E-01	6.1796E-01
37	8.8189E-01	7.7577E-01	6.8064E-01	5.9555E-01
38	8.7402E-01	7.6181E-01	6.6211E-01	5.7375E-01
39	8.6614E-01	7.4797E-01	6.4392E-01	5.5256E-01
40	8.5827E-01	7.3426E-01	6.2606E-01	5.3196E-01
41	8.5039E-01	7.2063E-01	6.0845E-01	5.1178E-01
42	9.2126E-01	8.4751E-01	7.7851E-01	7.1402E-01
43	9.1339E-01	8.3293E-01	7.5829E-01	6.8913E-01
44	9.0551E-01	8.1856E-01	7.3866E-01	6.6533E-01
45	8.9764E-01	8.0422E-01	7.1908E-01	6.4162E-01
46	8.8976E-01	7.8999E-01	6.9986E-01	6.1857E-01
47	8.8189E-01	7.7590E-01	6.8098E-01	5.9614E-01
48	8.7402E-01	7.6194E-01	6.6244E-01	5.7433E-01
49	8.6614E-01	7.4810E-01	6.4425E-01	5.5313E-01
50	8.5827E-01	7.3438E-01	6.2639E-01	5.3251E-01
51	8.5039E-01	7.2080E-01	6.0886E-01	5.1248E-01
52	8.4252E-01	7.0734E-01	5.9167E-01	4.9302E-01
53	8.3465E-01	6.9401E-01	5.7480E-01	4.7412E-01
54	8.2677E-01	6.8080E-01	5.5826E-01	4.5578E-01
55	8.1890E-01	6.6768E-01	5.4193E-01	4.3779E-01
56	9.0551E-01	8.1856E-01	7.3866E-01	6.6532E-01
57	8.9764E-01	8.0424E-01	7.1915E-01	6.4174E-01
58	8.8976E-01	7.9012E-01	7.0020E-01	6.1918E-01
59	8.8189E-01	7.7603E-01	6.8132E-01	5.9674E-01
60	8.7402E-01	7.6206E-01	6.6278E-01	5.7491E-01
61	8.6614E-01	7.4822E-01	6.4458E-01	5.5370E-01
62	8.5827E-01	7.3451E-01	6.2672E-01	5.3307E-01
63	8.5039E-01	7.2092E-01	6.0919E-01	5.1303E-01
64	8.4252E-01	7.0747E-01	5.9199E-01	4.9356E-01
65	8.3465E-01	6.9413E-01	5.7512E-01	4.7465E-01
66	8.2677E-01	6.8093E-01	5.5857E-01	4.5629E-01

$C_{if}(T), i=0$ (continued)				
67	8.1890E-01	6.6785E-01	5.4235E-01	4.3847E-01
68	8.1102E-01	6.5490E-01	5.2644E-01	4.2118E-01
69	8.0315E-01	6.4208E-01	5.1084E-01	4.0440E-01
70	7.9528E-01	6.2938E-01	4.9556E-01	3.8813E-01
71	7.8740E-01	6.1676E-01	4.8047E-01	3.7217E-01
72	8.8976E-01	7.9012E-01	7.0020E-01	6.1918E-01
73	8.8189E-01	7.7606E-01	6.8139E-01	5.9687E-01
74	8.7402E-01	7.6219E-01	6.6311E-01	5.7550E-01
75	8.6614E-01	7.4835E-01	6.4491E-01	5.5427E-01
76	8.5827E-01	7.3464E-01	6.2704E-01	5.3363E-01
77	8.5039E-01	7.2105E-01	6.0951E-01	5.1358E-01
78	8.4252E-01	7.0759E-01	5.9231E-01	4.9410E-01
79	8.3465E-01	6.9426E-01	5.7544E-01	4.7518E-01
80	8.2677E-01	6.8106E-01	5.5889E-01	4.5681E-01
81	8.1890E-01	6.6798E-01	5.4266E-01	4.3898E-01
82	8.1102E-01	6.5503E-01	5.2675E-01	4.2167E-01
83	8.0315E-01	6.4220E-01	5.1115E-01	4.0489E-01
84	7.9528E-01	6.2951E-01	4.9586E-01	3.8861E-01
85	7.8740E-01	6.1694E-01	4.8089E-01	3.7282E-01
86	7.7953E-01	6.0449E-01	4.6621E-01	3.5752E-01
87	7.7165E-01	5.9218E-01	4.5184E-01	3.4270E-01
88	7.6378E-01	5.7999E-01	4.3777E-01	3.2836E-01
89	7.5591E-01	5.6787E-01	4.2387E-01	3.1426E-01
90	8.7402E-01	7.6219E-01	6.6311E-01	5.7550E-01
91	8.6614E-01	7.4838E-01	6.4499E-01	5.5441E-01
92	8.5827E-01	7.3476E-01	6.2737E-01	5.3420E-01
93	8.5039E-01	7.2118E-01	6.0984E-01	5.1413E-01
94	8.4252E-01	7.0772E-01	5.9263E-01	4.9464E-01
95	8.3465E-01	6.9439E-01	5.7575E-01	4.7571E-01
96	8.2677E-01	6.8118E-01	5.5920E-01	4.5733E-01
97	8.1890E-01	6.6811E-01	5.4297E-01	4.3948E-01
98	8.1102E-01	6.5515E-01	5.2705E-01	4.2217E-01
99	8.0315E-01	6.4233E-01	5.1145E-01	4.0537E-01
100	7.9528E-01	6.2963E-01	4.9617E-01	3.8908E-01
101	7.8740E-01	6.1706E-01	4.8118E-01	3.7329E-01
102	7.7953E-01	6.0462E-01	4.6651E-01	3.5798E-01

$C_{if}(T), i=0$ (continued)				
103	7.7165E-01	5.9231E-01	4.5214E-01	3.4315E-01
104	7.6378E-01	5.8012E-01	4.3806E-01	3.2879E-01
105	7.5591E-01	5.6805E-01	4.2428E-01	3.1488E-01
106	7.4803E-01	5.5612E-01	4.1080E-01	3.0142E-01
107	7.4016E-01	5.4431E-01	3.9760E-01	2.8840E-01
108	7.3228E-01	5.3263E-01	3.8469E-01	2.7581E-01
109	7.2441E-01	5.2102E-01	3.7193E-01	2.6343E-01
110	8.5827E-01	7.3476E-01	6.2737E-01	5.3420E-01
111	8.5039E-01	7.2118E-01	6.0984E-01	5.1413E-01
112	8.4252E-01	7.0772E-01	5.9263E-01	4.9464E-01
113	8.3465E-01	6.9439E-01	5.7575E-01	4.7571E-01
114	8.2677E-01	6.8118E-01	5.5920E-01	4.5733E-01
115	8.1890E-01	6.6811E-01	5.4297E-01	4.3948E-01
116	8.1102E-01	6.5515E-01	5.2705E-01	4.2217E-01
117	8.0315E-01	6.4233E-01	5.1145E-01	4.0537E-01
118	7.9528E-01	6.2963E-01	4.9617E-01	3.8908E-01
119	7.8740E-01	6.1706E-01	4.8118E-01	3.7329E-01
120	7.7953E-01	6.0462E-01	4.6651E-01	3.5798E-01
121	7.7165E-01	5.9231E-01	4.5214E-01	3.4315E-01
122	7.6378E-01	5.8012E-01	4.3806E-01	3.2879E-01
123	7.5591E-01	5.6805E-01	4.2428E-01	3.1488E-01
124	7.4803E-01	5.5612E-01	4.1080E-01	3.0142E-01
125	7.4016E-01	5.4431E-01	3.9760E-01	2.8840E-01
126	7.3228E-01	5.3263E-01	3.8469E-01	2.7580E-01
127	7.2441E-01	5.2108E-01	3.7206E-01	2.6362E-01
128	7.1654E-01	5.0965E-01	3.5972E-01	2.5186E-01
129	7.0866E-01	4.9829E-01	3.4751E-01	2.4029E-01

*E.b C<sub>ij</sub>(T), i=1*

Tank	i = 1	i = 1	i = 1	i = 1
ID	f = 1	f = 2	f = 3	f = 4
0	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
1	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
2	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
3	7.8740E-03	1.5996E-02	2.4374E-02	3.3016E-02
4	1.5748E-02	3.1615E-02	4.7599E-02	6.3699E-02
5	2.3622E-02	4.7111E-02	7.0453E-02	9.3632E-02
6	1.5748E-02	3.1615E-02	4.7600E-02	6.3701E-02
7	2.3622E-02	4.7099E-02	7.0417E-02	9.3562E-02
8	3.1496E-02	6.2214E-02	9.2141E-02	1.2126E-01
9	3.9370E-02	7.7197E-02	1.1347E-01	1.4818E-01
10	4.7244E-02	9.1925E-02	1.3406E-01	1.7365E-01
11	5.5118E-02	1.0641E-01	1.5395E-01	1.9779E-01
12	3.1496E-02	6.2214E-02	9.2142E-02	1.2126E-01
13	3.9370E-02	7.7187E-02	1.1344E-01	1.4813E-01
14	4.7244E-02	9.1798E-02	1.3370E-01	1.7299E-01
15	5.5118E-02	1.0627E-01	1.5356E-01	1.9708E-01
16	6.2992E-02	1.2049E-01	1.7270E-01	2.1980E-01
17	7.0866E-02	1.3446E-01	1.9112E-01	2.4118E-01
18	7.8740E-02	1.4817E-01	2.0883E-01	2.6126E-01
19	8.6614E-02	1.6165E-01	2.2591E-01	2.8017E-01
20	4.7244E-02	9.1798E-02	1.3370E-01	1.7300E-01
21	5.5118E-02	1.0626E-01	1.5352E-01	1.9701E-01
22	6.2992E-02	1.2037E-01	1.7235E-01	2.1919E-01
23	7.0866E-02	1.3433E-01	1.9078E-01	2.4059E-01
24	7.8740E-02	1.4804E-01	2.0851E-01	2.6069E-01
25	8.6614E-02	1.6150E-01	2.2553E-01	2.7954E-01
26	9.4488E-02	1.7471E-01	2.4188E-01	2.9715E-01
27	1.0236E-01	1.8766E-01	2.5754E-01	3.1357E-01
28	1.1024E-01	2.0036E-01	2.7254E-01	3.2882E-01
29	1.1811E-01	2.1283E-01	2.8695E-01	3.4306E-01
30	6.2992E-02	1.2037E-01	1.7235E-01	2.1919E-01
31	7.0866E-02	1.3432E-01	1.9074E-01	2.4051E-01
32	7.8740E-02	1.4792E-01	2.0818E-01	2.6013E-01
33	8.6614E-02	1.6138E-01	2.2521E-01	2.7899E-01

$C_{if}(T), i=1$ (continued)				
34	9.4488E-02	1.7458E-01	2.4156E-01	2.9663E-01
35	1.0236E-01	1.8753E-01	2.5723E-01	3.1307E-01
36	1.1024E-01	2.0023E-01	2.7224E-01	3.2835E-01
37	1.1811E-01	2.1267E-01	2.8658E-01	3.4250E-01
38	1.2598E-01	2.2486E-01	3.0028E-01	3.5556E-01
39	1.3386E-01	2.3680E-01	3.1335E-01	3.6757E-01
40	1.4173E-01	2.4848E-01	3.2578E-01	3.7854E-01
41	1.4961E-01	2.5994E-01	3.3766E-01	3.8862E-01
42	7.8740E-02	1.4792E-01	2.0818E-01	2.6013E-01
43	8.6614E-02	1.6136E-01	2.2516E-01	2.7890E-01
44	9.4488E-02	1.7445E-01	2.4124E-01	2.9611E-01
45	1.0236E-01	1.8740E-01	2.5692E-01	3.1257E-01
46	1.1024E-01	2.0010E-01	2.7193E-01	3.2787E-01
47	1.1811E-01	2.1254E-01	2.8629E-01	3.4205E-01
48	1.2598E-01	2.2473E-01	2.9999E-01	3.5513E-01
49	1.3386E-01	2.3667E-01	3.1306E-01	3.6715E-01
50	1.4173E-01	2.4835E-01	3.2550E-01	3.7815E-01
51	1.4961E-01	2.5978E-01	3.3732E-01	3.8815E-01
52	1.5748E-01	2.7095E-01	3.4853E-01	3.9718E-01
53	1.6535E-01	2.8187E-01	3.5913E-01	4.0529E-01
54	1.7323E-01	2.9253E-01	3.6915E-01	4.1249E-01
55	1.8110E-01	3.0298E-01	3.7866E-01	4.1891E-01
56	9.4488E-02	1.7445E-01	2.4124E-01	2.9611E-01
57	1.0236E-01	1.8738E-01	2.5686E-01	3.1248E-01
58	1.1024E-01	1.9997E-01	2.7163E-01	3.2740E-01
59	1.1811E-01	2.1242E-01	2.8599E-01	3.4159E-01
60	1.2598E-01	2.2461E-01	2.9970E-01	3.5469E-01
61	1.3386E-01	2.3654E-01	3.1277E-01	3.6674E-01
62	1.4173E-01	2.4822E-01	3.2522E-01	3.7775E-01
63	1.4961E-01	2.5965E-01	3.3704E-01	3.8777E-01
64	1.5748E-01	2.7082E-01	3.4826E-01	3.9683E-01
65	1.6535E-01	2.8174E-01	3.5887E-01	4.0495E-01
66	1.7323E-01	2.9241E-01	3.6889E-01	4.1218E-01
67	1.8110E-01	3.0282E-01	3.7834E-01	4.1853E-01
68	1.8898E-01	3.1298E-01	3.8720E-01	4.2404E-01
69	1.9685E-01	3.2288E-01	3.9551E-01	4.2873E-01

$C_{if}(T), i=1$ (continued)				
70	2.0472E-01	3.3253E-01	4.0326E-01	4.3265E-01
71	2.1260E-01	3.4197E-01	4.1054E-01	4.3588E-01
72	1.1024E-01	1.9997E-01	2.7163E-01	3.2740E-01
73	1.1811E-01	2.1239E-01	2.8592E-01	3.4149E-01
74	1.2598E-01	2.2448E-01	2.9941E-01	3.5426E-01
75	1.3386E-01	2.3641E-01	3.1249E-01	3.6632E-01
76	1.4173E-01	2.4810E-01	3.2494E-01	3.7736E-01
77	1.4961E-01	2.5952E-01	3.3677E-01	3.8740E-01
78	1.5748E-01	2.7070E-01	3.4799E-01	3.9648E-01
79	1.6535E-01	2.8162E-01	3.5861E-01	4.0462E-01
80	1.7323E-01	2.9228E-01	3.6864E-01	4.1186E-01
81	1.8110E-01	3.0269E-01	3.7809E-01	4.1823E-01
82	1.8898E-01	3.1285E-01	3.8696E-01	4.2376E-01
83	1.9685E-01	3.2275E-01	3.9527E-01	4.2847E-01
84	2.0472E-01	3.3240E-01	4.0303E-01	4.3241E-01
85	2.1260E-01	3.4180E-01	4.1024E-01	4.3559E-01
86	2.2047E-01	3.5094E-01	4.1692E-01	4.3804E-01
87	2.2835E-01	3.5983E-01	4.2308E-01	4.3980E-01
88	2.3622E-01	3.6846E-01	4.2871E-01	4.4088E-01
89	2.4409E-01	3.7689E-01	4.3391E-01	4.4137E-01
90	1.2598E-01	2.2448E-01	2.9941E-01	3.5426E-01
91	1.3386E-01	2.3638E-01	3.1242E-01	3.6623E-01
92	1.4173E-01	2.4797E-01	3.2466E-01	3.7697E-01
93	1.4961E-01	2.5940E-01	3.3650E-01	3.8702E-01
94	1.5748E-01	2.7057E-01	3.4772E-01	3.9612E-01
95	1.6535E-01	2.8149E-01	3.5835E-01	4.0428E-01
96	1.7323E-01	2.9215E-01	3.6839E-01	4.1154E-01
97	1.8110E-01	3.0256E-01	3.7784E-01	4.1793E-01
98	1.8898E-01	3.1272E-01	3.8672E-01	4.2348E-01
99	1.9685E-01	3.2263E-01	3.9504E-01	4.2821E-01
100	2.0472E-01	3.3228E-01	4.0280E-01	4.3217E-01
101	2.1260E-01	3.4167E-01	4.1002E-01	4.3536E-01
102	2.2047E-01	3.5081E-01	4.1670E-01	4.3783E-01
103	2.2835E-01	3.5970E-01	4.2286E-01	4.3961E-01
104	2.3622E-01	3.6833E-01	4.2851E-01	4.4071E-01
105	2.4409E-01	3.7671E-01	4.3365E-01	4.4117E-01



$C_{if}(T), i=1$ (continued)				
106	2.5197E-01	3.8484E-01	4.3829E-01	4.4101E-01
107	2.5984E-01	3.9271E-01	4.4244E-01	4.4026E-01
108	2.6772E-01	4.0033E-01	4.4611E-01	4.3895E-01
109	2.7559E-01	4.0775E-01	4.4938E-01	4.3711E-01
110	1.4173E-01	2.4797E-01	3.2466E-01	3.7697E-01
111	1.4961E-01	2.5940E-01	3.3650E-01	3.8702E-01
112	1.5748E-01	2.7057E-01	3.4772E-01	3.9612E-01
113	1.6535E-01	2.8149E-01	3.5835E-01	4.0428E-01
114	1.7323E-01	2.9215E-01	3.6839E-01	4.1154E-01
115	1.8110E-01	3.0256E-01	3.7784E-01	4.1793E-01
116	1.8898E-01	3.1272E-01	3.8672E-01	4.2348E-01
117	1.9685E-01	3.2263E-01	3.9504E-01	4.2821E-01
118	2.0472E-01	3.3228E-01	4.0280E-01	4.3217E-01
119	2.1260E-01	3.4167E-01	4.1002E-01	4.3536E-01
120	2.2047E-01	3.5081E-01	4.1670E-01	4.3783E-01
121	2.2835E-01	3.5970E-01	4.2286E-01	4.3961E-01
122	2.3622E-01	3.6833E-01	4.2851E-01	4.4071E-01
123	2.4409E-01	3.7671E-01	4.3365E-01	4.4117E-01
124	2.5197E-01	3.8484E-01	4.3829E-01	4.4101E-01
125	2.5984E-01	3.9271E-01	4.4244E-01	4.4026E-01
126	2.6772E-01	4.0033E-01	4.4611E-01	4.3895E-01
127	2.7559E-01	4.0769E-01	4.4932E-01	4.3709E-01
128	2.8346E-01	4.1480E-01	4.5206E-01	4.3473E-01
129	2.9134E-01	4.2171E-01	4.5442E-01	4.3187E-01

*E.c C<sub>ij</sub>(T), i=2*

Tank ID	i = 2 f = 1	i = 2 f = 2	i = 2 f = 3	i = 2 f = 4
0	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
1	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
2	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
3	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
4	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
5	0.0000E+00	1.2698E-04	3.9023E-04	7.9969E-04
6	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
7	0.0000E+00	1.2695E-04	3.8993E-04	7.9866E-04
8	0.0000E+00	3.8090E-04	1.1514E-03	2.3203E-03
9	0.0000E+00	7.6181E-04	2.2902E-03	4.5894E-03
10	0.0000E+00	1.2697E-03	3.7876E-03	7.5302E-03
11	0.0000E+00	1.9048E-03	5.6395E-03	1.1126E-02
12	0.0000E+00	3.8090E-04	1.1514E-03	2.3204E-03
13	0.0000E+00	7.6171E-04	2.2893E-03	4.5865E-03
14	0.0000E+00	1.2697E-03	3.7750E-03	7.4809E-03
15	0.0000E+00	1.9045E-03	5.6247E-03	1.1070E-02
16	0.0000E+00	2.6663E-03	7.8142E-03	1.5259E-02
17	0.0000E+00	3.5551E-03	1.0337E-02	2.0023E-02
18	0.0000E+00	4.5708E-03	1.3184E-02	2.5327E-02
19	0.0000E+00	5.7143E-03	1.6352E-02	3.1158E-02
20	0.0000E+00	1.2697E-03	3.7750E-03	7.4810E-03
21	0.0000E+00	1.9043E-03	5.6226E-03	1.1063E-02
22	0.0000E+00	2.6663E-03	7.7954E-03	1.5187E-02
23	0.0000E+00	3.5551E-03	1.0318E-02	1.9952E-02
24	0.0000E+00	4.5708E-03	1.3162E-02	2.5246E-02
25	0.0000E+00	5.7136E-03	1.6321E-02	3.1046E-02
26	0.0000E+00	6.9832E-03	1.9784E-02	3.7319E-02
27	0.0000E+00	8.3799E-03	2.3543E-02	4.4031E-02
28	0.0000E+00	9.9035E-03	2.7589E-02	5.1149E-02
29	0.0000E+00	1.1556E-02	3.1924E-02	5.8680E-02
30	0.0000E+00	2.6663E-03	7.7954E-03	1.5187E-02
31	0.0000E+00	3.5547E-03	1.0314E-02	1.9939E-02
32	0.0000E+00	4.5708E-03	1.3137E-02	2.5152E-02
33	0.0000E+00	5.7136E-03	1.6295E-02	3.0954E-02

$C_{if}(T), i=2$ (continued)				
34	0.0000E+00	6.9832E-03	1.9756E-02	3.7217E-02
35	0.0000E+00	8.3799E-03	2.3512E-02	4.3919E-02
36	0.0000E+00	9.9035E-03	2.7554E-02	5.1029E-02
37	0.0000E+00	1.1554E-02	3.1873E-02	5.8513E-02
38	0.0000E+00	1.3332E-02	3.6460E-02	6.6341E-02
39	0.0000E+00	1.5236E-02	4.1305E-02	7.4483E-02
40	0.0000E+00	1.7268E-02	4.6398E-02	8.2906E-02
41	0.0000E+00	1.9429E-02	5.1750E-02	9.1641E-02
42	0.0000E+00	4.5708E-03	1.3137E-02	2.5153E-02
43	0.0000E+00	5.7128E-03	1.6289E-02	3.0935E-02
44	0.0000E+00	6.9832E-03	1.9724E-02	3.7103E-02
45	0.0000E+00	8.3799E-03	2.3480E-02	4.3807E-02
46	0.0000E+00	9.9035E-03	2.7520E-02	5.0907E-02
47	0.0000E+00	1.1554E-02	3.1836E-02	5.8383E-02
48	0.0000E+00	1.3332E-02	3.6419E-02	6.6202E-02
49	0.0000E+00	1.5236E-02	4.1261E-02	7.4335E-02
50	0.0000E+00	1.7268E-02	4.6351E-02	8.2751E-02
51	0.0000E+00	1.9426E-02	5.1680E-02	9.1421E-02
52	0.0000E+00	2.1712E-02	5.7238E-02	1.0032E-01
53	0.0000E+00	2.4124E-02	6.3017E-02	1.0941E-01
54	0.0000E+00	2.6663E-02	6.9007E-02	1.1867E-01
55	0.0000E+00	2.9333E-02	7.5227E-02	1.2815E-01
56	0.0000E+00	6.9832E-03	1.9724E-02	3.7104E-02
57	0.0000E+00	8.3788E-03	2.3471E-02	4.3781E-02
58	0.0000E+00	9.9035E-03	2.7482E-02	5.0775E-02
59	0.0000E+00	1.1554E-02	3.1798E-02	5.8252E-02
60	0.0000E+00	1.3332E-02	3.6378E-02	6.6063E-02
61	0.0000E+00	1.5236E-02	4.1217E-02	7.4187E-02
62	0.0000E+00	1.7268E-02	4.6303E-02	8.2595E-02
63	0.0000E+00	1.9426E-02	5.1629E-02	9.1258E-02
64	0.0000E+00	2.1712E-02	5.7185E-02	1.0015E-01
65	0.0000E+00	2.4124E-02	6.2960E-02	1.0923E-01
66	0.0000E+00	2.6663E-02	6.8947E-02	1.1849E-01
67	0.0000E+00	2.9330E-02	7.5135E-02	1.2788E-01
68	0.0000E+00	3.2123E-02	8.1515E-02	1.3740E-01
69	0.0000E+00	3.5043E-02	8.8077E-02	1.4701E-01

$C_{if}(T), i=2$ (continued)				
70	0.0000E+00	3.8090E-02	9.4812E-02	1.5668E-01
71	0.0000E+00	4.1270E-02	1.0175E-01	1.6649E-01
72	0.0000E+00	9.9035E-03	2.7482E-02	5.0776E-02
73	0.0000E+00	1.1553E-02	3.1786E-02	5.8217E-02
74	0.0000E+00	1.3332E-02	3.6334E-02	6.5913E-02
75	0.0000E+00	1.5236E-02	4.1173E-02	7.4040E-02
76	0.0000E+00	1.7268E-02	4.6256E-02	8.2440E-02
77	0.0000E+00	1.9426E-02	5.1579E-02	9.1094E-02
78	0.0000E+00	2.1712E-02	5.7131E-02	9.9974E-02
79	0.0000E+00	2.4124E-02	6.2904E-02	1.0905E-01
80	0.0000E+00	2.6663E-02	6.8887E-02	1.1830E-01
81	0.0000E+00	2.9330E-02	7.5072E-02	1.2769E-01
82	0.0000E+00	3.2123E-02	8.1448E-02	1.3720E-01
83	0.0000E+00	3.5043E-02	8.8008E-02	1.4680E-01
84	0.0000E+00	3.8090E-02	9.4740E-02	1.5647E-01
85	0.0000E+00	4.1265E-02	1.0164E-01	1.6618E-01
86	0.0000E+00	4.4566E-02	1.0869E-01	1.7591E-01
87	0.0000E+00	4.7994E-02	1.1588E-01	1.8563E-01
88	0.0000E+00	5.1549E-02	1.2321E-01	1.9533E-01
89	0.0000E+00	5.5238E-02	1.3072E-01	2.0510E-01
90	0.0000E+00	1.3332E-02	3.6334E-02	6.5914E-02
91	0.0000E+00	1.5234E-02	4.1157E-02	7.3995E-02
92	0.0000E+00	1.7268E-02	4.6206E-02	8.2274E-02
93	0.0000E+00	1.9426E-02	5.1529E-02	9.0931E-02
94	0.0000E+00	2.1712E-02	5.7078E-02	9.9803E-02
95	0.0000E+00	2.4124E-02	6.2847E-02	1.0887E-01
96	0.0000E+00	2.6663E-02	6.8827E-02	1.1812E-01
97	0.0000E+00	2.9330E-02	7.5009E-02	1.2750E-01
98	0.0000E+00	3.2123E-02	8.1382E-02	1.3700E-01
99	0.0000E+00	3.5043E-02	8.7938E-02	1.4660E-01
100	0.0000E+00	3.8090E-02	9.4667E-02	1.5626E-01
101	0.0000E+00	4.1265E-02	1.0156E-01	1.6597E-01
102	0.0000E+00	4.4566E-02	1.0861E-01	1.7569E-01
103	0.0000E+00	4.7994E-02	1.1580E-01	1.8541E-01
104	0.0000E+00	5.1549E-02	1.2312E-01	1.9510E-01
105	0.0000E+00	5.5231E-02	1.3058E-01	2.0475E-01

$C_{if}(T), i=2$ (continued)				
106	0.0000E+00	5.9040E-02	1.3815E-01	2.1433E-01
107	0.0000E+00	6.2976E-02	1.4582E-01	2.2381E-01
108	0.0000E+00	6.7039E-02	1.5359E-01	2.3319E-01
109	0.0000E+00	7.1238E-02	1.6152E-01	2.4257E-01
110	0.0000E+00	1.7268E-02	4.6206E-02	8.2274E-02
111	0.0000E+00	1.9426E-02	5.1529E-02	9.0931E-02
112	0.0000E+00	2.1712E-02	5.7078E-02	9.9803E-02
113	0.0000E+00	2.4124E-02	6.2847E-02	1.0887E-01
114	0.0000E+00	2.6663E-02	6.8827E-02	1.1812E-01
115	0.0000E+00	2.9330E-02	7.5009E-02	1.2750E-01
116	0.0000E+00	3.2123E-02	8.1382E-02	1.3700E-01
117	0.0000E+00	3.5043E-02	8.7938E-02	1.4660E-01
118	0.0000E+00	3.8090E-02	9.4667E-02	1.5626E-01
119	0.0000E+00	4.1265E-02	1.0156E-01	1.6597E-01
120	0.0000E+00	4.4566E-02	1.0861E-01	1.7569E-01
121	0.0000E+00	4.7994E-02	1.1580E-01	1.8541E-01
122	0.0000E+00	5.1549E-02	1.2312E-01	1.9510E-01
123	0.0000E+00	5.5231E-02	1.3058E-01	2.0475E-01
124	0.0000E+00	5.9040E-02	1.3815E-01	2.1433E-01
125	0.0000E+00	6.2976E-02	1.4582E-01	2.2381E-01
126	0.0000E+00	6.7039E-02	1.5359E-01	2.3319E-01
127	0.0000E+00	7.1229E-02	1.6146E-01	2.4244E-01
128	0.0000E+00	7.5546E-02	1.6940E-01	2.5154E-01
129	0.0000E+00	8.0000E-02	1.7747E-01	2.6062E-01

*E.d C<sub>ij</sub>(T), i=3*

Tank ID	i = 3 f = 1	i = 3 f = 2	i = 3 f = 3	i = 3 f = 4
0	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
1	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
2	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
3	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
4	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
5	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
6	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
7	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
8	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
9	0.0000E+00	0.0000E+00	3.1458E-06	1.2993E-05
10	0.0000E+00	0.0000E+00	1.2583E-05	5.1226E-05
11	0.0000E+00	0.0000E+00	3.1470E-05	1.2715E-04
12	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
13	0.0000E+00	0.0000E+00	3.1446E-06	1.2983E-05
14	0.0000E+00	0.0000E+00	1.2583E-05	5.0907E-05
15	0.0000E+00	0.0000E+00	3.1458E-05	1.2673E-04
16	0.0000E+00	0.0000E+00	6.2916E-05	2.5134E-04
17	0.0000E+00	0.0000E+00	1.1010E-04	4.3644E-04
18	0.0000E+00	0.0000E+00	1.7617E-04	6.9277E-04
19	0.0000E+00	0.0000E+00	2.6435E-04	1.0316E-03
20	0.0000E+00	0.0000E+00	1.2583E-05	5.0908E-05
21	0.0000E+00	0.0000E+00	3.1446E-05	1.2664E-04
22	0.0000E+00	0.0000E+00	6.2917E-05	2.5028E-04
23	0.0000E+00	0.0000E+00	1.1010E-04	4.3537E-04
24	0.0000E+00	0.0000E+00	1.7617E-04	6.9108E-04
25	0.0000E+00	0.0000E+00	2.6425E-04	1.0285E-03
26	0.0000E+00	0.0000E+00	3.7750E-04	1.4574E-03
27	0.0000E+00	0.0000E+00	5.1906E-04	1.9876E-03
28	0.0000E+00	0.0000E+00	6.9208E-04	2.6281E-03
29	0.0000E+00	0.0000E+00	9.0005E-04	3.3903E-03
30	0.0000E+00	0.0000E+00	6.2917E-05	2.5028E-04
31	0.0000E+00	0.0000E+00	1.1006E-04	4.3503E-04
32	0.0000E+00	0.0000E+00	1.7617E-04	6.8885E-04
33	0.0000E+00	0.0000E+00	2.6425E-04	1.0262E-03

$C_{if}(T), i=3$ (continued)				
34	0.0000E+00	0.0000E+00	3.7750E-04	1.4544E-03
35	0.0000E+00	0.0000E+00	5.1906E-04	1.9837E-03
36	0.0000E+00	0.0000E+00	6.9208E-04	2.6232E-03
37	0.0000E+00	0.0000E+00	8.9970E-04	3.3817E-03
38	0.0000E+00	0.0000E+00	1.1451E-03	4.2674E-03
39	0.0000E+00	0.0000E+00	1.4313E-03	5.2883E-03
40	0.0000E+00	0.0000E+00	1.7617E-03	6.4518E-03
41	0.0000E+00	0.0000E+00	2.1400E-03	7.7711E-03
42	0.0000E+00	0.0000E+00	1.7617E-04	6.8886E-04
43	0.0000E+00	0.0000E+00	2.6415E-04	1.0254E-03
44	0.0000E+00	0.0000E+00	3.7750E-04	1.4505E-03
45	0.0000E+00	0.0000E+00	5.1906E-04	1.9798E-03
46	0.0000E+00	0.0000E+00	6.9208E-04	2.6183E-03
47	0.0000E+00	0.0000E+00	8.9970E-04	3.3757E-03
48	0.0000E+00	0.0000E+00	1.1451E-03	4.2603E-03
49	0.0000E+00	0.0000E+00	1.4313E-03	5.2799E-03
50	0.0000E+00	0.0000E+00	1.7617E-03	6.4420E-03
51	0.0000E+00	0.0000E+00	2.1392E-03	7.7537E-03
52	0.0000E+00	0.0000E+00	2.5670E-03	9.2215E-03
53	0.0000E+00	0.0000E+00	3.0483E-03	1.0852E-02
54	0.0000E+00	0.0000E+00	3.5862E-03	1.2650E-02
55	0.0000E+00	0.0000E+00	4.1856E-03	1.4633E-02
56	0.0000E+00	0.0000E+00	3.7750E-04	1.4506E-03
57	0.0000E+00	0.0000E+00	5.1886E-04	1.9783E-03
58	0.0000E+00	0.0000E+00	6.9208E-04	2.6125E-03
59	0.0000E+00	0.0000E+00	8.9970E-04	3.3699E-03
60	0.0000E+00	0.0000E+00	1.1451E-03	4.2532E-03
61	0.0000E+00	0.0000E+00	1.4313E-03	5.2715E-03
62	0.0000E+00	0.0000E+00	1.7617E-03	6.4323E-03
63	0.0000E+00	0.0000E+00	2.1392E-03	7.7424E-03
64	0.0000E+00	0.0000E+00	2.5670E-03	9.2086E-03
65	0.0000E+00	0.0000E+00	3.0483E-03	1.0837E-02
66	0.0000E+00	0.0000E+00	3.5862E-03	1.2633E-02
67	0.0000E+00	0.0000E+00	4.1839E-03	1.4603E-02
68	0.0000E+00	0.0000E+00	4.8446E-03	1.6751E-02
69	0.0000E+00	0.0000E+00	5.5713E-03	1.9081E-02

$C_{if}(T), i=3$ (continued)				
70	0.0000E+00	0.0000E+00	6.3671E-03	2.1598E-02
71	0.0000E+00	0.0000E+00	7.2382E-03	2.4325E-02
72	0.0000E+00	0.0000E+00	6.9209E-04	2.6125E-03
73	0.0000E+00	0.0000E+00	8.9936E-04	3.3673E-03
74	0.0000E+00	0.0000E+00	1.1451E-03	4.2449E-03
75	0.0000E+00	0.0000E+00	1.4313E-03	5.2632E-03
76	0.0000E+00	0.0000E+00	1.7617E-03	6.4225E-03
77	0.0000E+00	0.0000E+00	2.1392E-03	7.7311E-03
78	0.0000E+00	0.0000E+00	2.5670E-03	9.1957E-03
79	0.0000E+00	0.0000E+00	3.0483E-03	1.0822E-02
80	0.0000E+00	0.0000E+00	3.5862E-03	1.2617E-02
81	0.0000E+00	0.0000E+00	4.1839E-03	1.4585E-02
82	0.0000E+00	0.0000E+00	4.8446E-03	1.6730E-02
83	0.0000E+00	0.0000E+00	5.5713E-03	1.9059E-02
84	0.0000E+00	0.0000E+00	6.3671E-03	2.1574E-02
85	0.0000E+00	0.0000E+00	7.2354E-03	2.4279E-02
86	0.0000E+00	0.0000E+00	8.1791E-03	2.7177E-02
87	0.0000E+00	0.0000E+00	9.2015E-03	3.0272E-02
88	0.0000E+00	0.0000E+00	1.0306E-02	3.3564E-02
89	0.0000E+00	0.0000E+00	1.1499E-02	3.7088E-02
90	0.0000E+00	0.0000E+00	1.1451E-03	4.2450E-03
91	0.0000E+00	0.0000E+00	1.4308E-03	5.2591E-03
92	0.0000E+00	0.0000E+00	1.7617E-03	6.4114E-03
93	0.0000E+00	0.0000E+00	2.1392E-03	7.7199E-03
94	0.0000E+00	0.0000E+00	2.5670E-03	9.1828E-03
95	0.0000E+00	0.0000E+00	3.0483E-03	1.0808E-02
96	0.0000E+00	0.0000E+00	3.5862E-03	1.2600E-02
97	0.0000E+00	0.0000E+00	4.1839E-03	1.4566E-02
98	0.0000E+00	0.0000E+00	4.8446E-03	1.6710E-02
99	0.0000E+00	0.0000E+00	5.5713E-03	1.9036E-02
100	0.0000E+00	0.0000E+00	6.3671E-03	2.1549E-02
101	0.0000E+00	0.0000E+00	7.2354E-03	2.4252E-02
102	0.0000E+00	0.0000E+00	8.1791E-03	2.7148E-02
103	0.0000E+00	0.0000E+00	9.2015E-03	3.0240E-02
104	0.0000E+00	0.0000E+00	1.0306E-02	3.3530E-02
105	0.0000E+00	0.0000E+00	1.1495E-02	3.7021E-02



$C_{if}(T), i=3$ (continued)				
106	0.0000E+00	0.0000E+00	1.2772E-02	4.0714E-02
107	0.0000E+00	0.0000E+00	1.4140E-02	4.4609E-02
108	0.0000E+00	0.0000E+00	1.5603E-02	4.8708E-02
109	0.0000E+00	0.0000E+00	1.7170E-02	5.3053E-02
110	0.0000E+00	0.0000E+00	1.7617E-03	6.4114E-03
111	0.0000E+00	0.0000E+00	2.1392E-03	7.7199E-03
112	0.0000E+00	0.0000E+00	2.5670E-03	9.1828E-03
113	0.0000E+00	0.0000E+00	3.0483E-03	1.0808E-02
114	0.0000E+00	0.0000E+00	3.5862E-03	1.2600E-02
115	0.0000E+00	0.0000E+00	4.1839E-03	1.4566E-02
116	0.0000E+00	0.0000E+00	4.8446E-03	1.6710E-02
117	0.0000E+00	0.0000E+00	5.5713E-03	1.9036E-02
118	0.0000E+00	0.0000E+00	6.3671E-03	2.1549E-02
119	0.0000E+00	0.0000E+00	7.2354E-03	2.4252E-02
120	0.0000E+00	0.0000E+00	8.1791E-03	2.7148E-02
121	0.0000E+00	0.0000E+00	9.2015E-03	3.0240E-02
122	0.0000E+00	0.0000E+00	1.0306E-02	3.3530E-02
123	0.0000E+00	0.0000E+00	1.1495E-02	3.7021E-02
124	0.0000E+00	0.0000E+00	1.2772E-02	4.0714E-02
125	0.0000E+00	0.0000E+00	1.4140E-02	4.4609E-02
126	0.0000E+00	0.0000E+00	1.5603E-02	4.8708E-02
127	0.0000E+00	0.0000E+00	1.7164E-02	5.3011E-02
128	0.0000E+00	0.0000E+00	1.8825E-02	5.7517E-02
129	0.0000E+00	0.0000E+00	2.0597E-02	6.2276E-02

*E.e C<sub>ij</sub>(T), i=4*

Tank	i = 4	i = 4	i = 4	i = 4
ID	f = 1	f = 2	f = 3	f = 4
0	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
1	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
2	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
3	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
4	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
5	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
6	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
7	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
8	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
9	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
10	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
11	0.0000E+00	0.0000E+00	0.0000E+00	1.0658E-07
12	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
13	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
14	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
15	0.0000E+00	0.0000E+00	0.0000E+00	1.0650E-07
16	0.0000E+00	0.0000E+00	0.0000E+00	5.3250E-07
17	0.0000E+00	0.0000E+00	0.0000E+00	1.5975E-06
18	0.0000E+00	0.0000E+00	0.0000E+00	3.7274E-06
19	0.0000E+00	0.0000E+00	0.0000E+00	7.4608E-06
20	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
21	0.0000E+00	0.0000E+00	0.0000E+00	1.0642E-07
22	0.0000E+00	0.0000E+00	0.0000E+00	5.3250E-07
23	0.0000E+00	0.0000E+00	0.0000E+00	1.5975E-06
24	0.0000E+00	0.0000E+00	0.0000E+00	3.7275E-06
25	0.0000E+00	0.0000E+00	0.0000E+00	7.4550E-06
26	0.0000E+00	0.0000E+00	0.0000E+00	1.3419E-05
27	0.0000E+00	0.0000E+00	0.0000E+00	2.2365E-05
28	0.0000E+00	0.0000E+00	0.0000E+00	3.5144E-05
29	0.0000E+00	0.0000E+00	0.0000E+00	5.2759E-05
30	0.0000E+00	0.0000E+00	0.0000E+00	5.3251E-07
31	0.0000E+00	0.0000E+00	0.0000E+00	1.5963E-06
32	0.0000E+00	0.0000E+00	0.0000E+00	3.7275E-06
33	0.0000E+00	0.0000E+00	0.0000E+00	7.4550E-06

$C_{if}(T), i=4$ (continued)				
34	0.0000E+00	0.0000E+00	0.0000E+00	1.3419E-05
35	0.0000E+00	0.0000E+00	0.0000E+00	2.2365E-05
36	0.0000E+00	0.0000E+00	0.0000E+00	3.5145E-05
37	0.0000E+00	0.0000E+00	0.0000E+00	5.2717E-05
38	0.0000E+00	0.0000E+00	0.0000E+00	7.6147E-05
39	0.0000E+00	0.0000E+00	0.0000E+00	1.0661E-04
40	0.0000E+00	0.0000E+00	0.0000E+00	1.4537E-04
41	0.0000E+00	0.0000E+00	0.0000E+00	1.9398E-04
42	0.0000E+00	0.0000E+00	0.0000E+00	3.7276E-06
43	0.0000E+00	0.0000E+00	0.0000E+00	7.4492E-06
44	0.0000E+00	0.0000E+00	0.0000E+00	1.3419E-05
45	0.0000E+00	0.0000E+00	0.0000E+00	2.2365E-05
46	0.0000E+00	0.0000E+00	0.0000E+00	3.5145E-05
47	0.0000E+00	0.0000E+00	0.0000E+00	5.2717E-05
48	0.0000E+00	0.0000E+00	0.0000E+00	7.6147E-05
49	0.0000E+00	0.0000E+00	0.0000E+00	1.0661E-04
50	0.0000E+00	0.0000E+00	0.0000E+00	1.4537E-04
51	0.0000E+00	0.0000E+00	0.0000E+00	1.9383E-04
52	0.0000E+00	0.0000E+00	0.0000E+00	2.5347E-04
53	0.0000E+00	0.0000E+00	0.0000E+00	3.2589E-04
54	0.0000E+00	0.0000E+00	0.0000E+00	4.1279E-04
55	0.0000E+00	0.0000E+00	0.0000E+00	5.1640E-04
56	0.0000E+00	0.0000E+00	0.0000E+00	1.3419E-05
57	0.0000E+00	0.0000E+00	0.0000E+00	2.2348E-05
58	0.0000E+00	0.0000E+00	0.0000E+00	3.5145E-05
59	0.0000E+00	0.0000E+00	0.0000E+00	5.2717E-05
60	0.0000E+00	0.0000E+00	0.0000E+00	7.6147E-05
61	0.0000E+00	0.0000E+00	0.0000E+00	1.0661E-04
62	0.0000E+00	0.0000E+00	0.0000E+00	1.4537E-04
63	0.0000E+00	0.0000E+00	0.0000E+00	1.9383E-04
64	0.0000E+00	0.0000E+00	0.0000E+00	2.5347E-04
65	0.0000E+00	0.0000E+00	0.0000E+00	3.2589E-04
66	0.0000E+00	0.0000E+00	0.0000E+00	4.1279E-04
67	0.0000E+00	0.0000E+00	0.0000E+00	5.1599E-04
68	0.0000E+00	0.0000E+00	0.0000E+00	6.3740E-04
69	0.0000E+00	0.0000E+00	0.0000E+00	7.7904E-04

$C_{if}(T), i=4$ (continued)				
70	0.0000E+00	0.0000E+00	0.0000E+00	9.4304E-04
71	0.0000E+00	0.0000E+00	0.0000E+00	1.1326E-03
72	0.0000E+00	0.0000E+00	0.0000E+00	3.5146E-05
73	0.0000E+00	0.0000E+00	0.0000E+00	5.2677E-05
74	0.0000E+00	0.0000E+00	0.0000E+00	7.6148E-05
75	0.0000E+00	0.0000E+00	0.0000E+00	1.0661E-04
76	0.0000E+00	0.0000E+00	0.0000E+00	1.4537E-04
77	0.0000E+00	0.0000E+00	0.0000E+00	1.9383E-04
78	0.0000E+00	0.0000E+00	0.0000E+00	2.5347E-04
79	0.0000E+00	0.0000E+00	0.0000E+00	3.2589E-04
80	0.0000E+00	0.0000E+00	0.0000E+00	4.1279E-04
81	0.0000E+00	0.0000E+00	0.0000E+00	5.1599E-04
82	0.0000E+00	0.0000E+00	0.0000E+00	6.3740E-04
83	0.0000E+00	0.0000E+00	0.0000E+00	7.7904E-04
84	0.0000E+00	0.0000E+00	0.0000E+00	9.4305E-04
85	0.0000E+00	0.0000E+00	0.0000E+00	1.1317E-03
86	0.0000E+00	0.0000E+00	0.0000E+00	1.3472E-03
87	0.0000E+00	0.0000E+00	0.0000E+00	1.5922E-03
88	0.0000E+00	0.0000E+00	0.0000E+00	1.8690E-03
89	0.0000E+00	0.0000E+00	0.0000E+00	2.1823E-03
90	0.0000E+00	0.0000E+00	0.0000E+00	7.6149E-05
91	0.0000E+00	0.0000E+00	0.0000E+00	1.0652E-04
92	0.0000E+00	0.0000E+00	0.0000E+00	1.4537E-04
93	0.0000E+00	0.0000E+00	0.0000E+00	1.9383E-04
94	0.0000E+00	0.0000E+00	0.0000E+00	2.5347E-04
95	0.0000E+00	0.0000E+00	0.0000E+00	3.2589E-04
96	0.0000E+00	0.0000E+00	0.0000E+00	4.1279E-04
97	0.0000E+00	0.0000E+00	0.0000E+00	5.1599E-04
98	0.0000E+00	0.0000E+00	0.0000E+00	6.3740E-04
99	0.0000E+00	0.0000E+00	0.0000E+00	7.7904E-04
100	0.0000E+00	0.0000E+00	0.0000E+00	9.4305E-04
101	0.0000E+00	0.0000E+00	0.0000E+00	1.1317E-03
102	0.0000E+00	0.0000E+00	0.0000E+00	1.3472E-03
103	0.0000E+00	0.0000E+00	0.0000E+00	1.5922E-03
104	0.0000E+00	0.0000E+00	0.0000E+00	1.8691E-03
105	0.0000E+00	0.0000E+00	0.0000E+00	2.1806E-03

$C_{if}(T), i=4$ (continued)				
106	0.0000E+00	0.0000E+00	0.0000E+00	2.5295E-03
107	0.0000E+00	0.0000E+00	0.0000E+00	2.9186E-03
108	0.0000E+00	0.0000E+00	0.0000E+00	3.3510E-03
109	0.0000E+00	0.0000E+00	0.0000E+00	3.8327E-03
110	0.0000E+00	0.0000E+00	0.0000E+00	1.4537E-04
111	0.0000E+00	0.0000E+00	0.0000E+00	1.9383E-04
112	0.0000E+00	0.0000E+00	0.0000E+00	2.5347E-04
113	0.0000E+00	0.0000E+00	0.0000E+00	3.2589E-04
114	0.0000E+00	0.0000E+00	0.0000E+00	4.1279E-04
115	0.0000E+00	0.0000E+00	0.0000E+00	5.1599E-04
116	0.0000E+00	0.0000E+00	0.0000E+00	6.3740E-04
117	0.0000E+00	0.0000E+00	0.0000E+00	7.7904E-04
118	0.0000E+00	0.0000E+00	0.0000E+00	9.4305E-04
119	0.0000E+00	0.0000E+00	0.0000E+00	1.1317E-03
120	0.0000E+00	0.0000E+00	0.0000E+00	1.3472E-03
121	0.0000E+00	0.0000E+00	0.0000E+00	1.5922E-03
122	0.0000E+00	0.0000E+00	0.0000E+00	1.8691E-03
123	0.0000E+00	0.0000E+00	0.0000E+00	2.1806E-03
124	0.0000E+00	0.0000E+00	0.0000E+00	2.5295E-03
125	0.0000E+00	0.0000E+00	0.0000E+00	2.9186E-03
126	0.0000E+00	0.0000E+00	0.0000E+00	3.3510E-03
127	0.0000E+00	0.0000E+00	0.0000E+00	3.8297E-03
128	0.0000E+00	0.0000E+00	0.0000E+00	4.3579E-03
129	0.0000E+00	0.0000E+00	0.0000E+00	4.9429E-03

## Appendix F – Combinatorial Algebra Results, $C_i(T)$

Tank	i=0	i=1	i=2	i=3	i=4
0	9.8577E-01	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
1	9.8577E-01	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
2	1.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
3	9.9988E-01	1.1611E-04	0.0000E+00	0.0000E+00	0.0000E+00
4	9.9977E-01	2.3203E-04	0.0000E+00	0.0000E+00	0.0000E+00
5	9.9965E-01	3.4789E-04	6.4566E-08	0.0000E+00	0.0000E+00
6	9.9977E-01	2.3203E-04	0.0000E+00	0.0000E+00	0.0000E+00
7	9.9965E-01	3.4788E-04	6.4548E-08	0.0000E+00	0.0000E+00
8	9.9954E-01	4.6355E-04	1.9350E-07	0.0000E+00	0.0000E+00
9	9.9942E-01	5.7914E-04	3.8688E-07	2.8828E-11	0.0000E+00
10	9.9930E-01	6.9461E-04	6.4453E-07	1.1516E-10	0.0000E+00
11	9.9919E-01	8.0996E-04	9.6653E-07	2.8783E-10	2.1219E-14
12	9.9954E-01	4.6355E-04	1.9350E-07	0.0000E+00	0.0000E+00
13	9.9942E-01	5.7914E-04	3.8683E-07	2.8816E-11	0.0000E+00
14	9.9930E-01	6.9455E-04	6.4442E-07	1.1510E-10	0.0000E+00
15	9.9919E-01	8.0989E-04	9.6628E-07	2.8765E-10	2.1202E-14
16	9.9907E-01	9.2510E-04	1.3522E-06	5.7487E-10	1.0601E-13
17	9.9896E-01	1.0402E-03	1.8022E-06	1.0053E-09	3.1803E-13
18	9.9884E-01	1.1551E-03	2.3162E-06	1.6074E-09	7.4206E-13
19	9.9873E-01	1.2700E-03	2.8944E-06	2.4105E-09	1.4853E-12
20	9.9930E-01	6.9455E-04	6.4442E-07	1.1510E-10	0.0000E+00
21	9.9919E-01	8.0988E-04	9.6615E-07	2.8753E-10	2.1186E-14
22	9.9907E-01	9.2503E-04	1.3521E-06	5.7466E-10	1.0601E-13
23	9.9896E-01	1.0401E-03	1.8021E-06	1.0051E-09	3.1803E-13
24	9.9884E-01	1.1551E-03	2.3160E-06	1.6071E-09	7.4207E-13
25	9.9873E-01	1.2699E-03	2.8938E-06	2.4090E-09	1.4841E-12
26	9.9861E-01	1.3846E-03	3.5354E-06	3.4391E-09	2.6714E-12
27	9.9850E-01	1.4992E-03	4.2407E-06	4.7255E-09	4.4524E-12
28	9.9838E-01	1.6136E-03	5.0095E-06	6.2963E-09	6.9965E-12
29	9.9827E-01	1.7279E-03	5.8428E-06	8.1829E-09	1.0503E-11
30	9.9907E-01	9.2503E-04	1.3521E-06	5.7466E-10	1.0601E-13
31	9.9896E-01	1.0401E-03	1.8018E-06	1.0047E-09	3.1778E-13
32	9.9884E-01	1.1550E-03	2.3158E-06	1.6067E-09	7.4208E-13
33	9.9873E-01	1.2698E-03	2.8936E-06	2.4086E-09	1.4841E-12

$C_i(T)$ (continued)					
34	9.9861E-01	1.3845E-03	3.5351E-06	3.4385E-09	2.6714E-12
35	9.9850E-01	1.4991E-03	4.2404E-06	4.7248E-09	4.4524E-12
36	9.9838E-01	1.6135E-03	5.0092E-06	6.2954E-09	6.9966E-12
37	9.9827E-01	1.7278E-03	5.8416E-06	8.1783E-09	1.0495E-11
38	9.9815E-01	1.8420E-03	6.7375E-06	1.0401E-08	1.5159E-11
39	9.9804E-01	1.9561E-03	7.6967E-06	1.2993E-08	2.1223E-11
40	9.9792E-01	2.0700E-03	8.7191E-06	1.5980E-08	2.8940E-11
41	9.9781E-01	2.1838E-03	9.8061E-06	1.9398E-08	3.8618E-11
42	9.9884E-01	1.1550E-03	2.3158E-06	1.6067E-09	7.4209E-13
43	9.9873E-01	1.2698E-03	2.8932E-06	2.4076E-09	1.4830E-12
44	9.9861E-01	1.3845E-03	3.5348E-06	3.4378E-09	2.6715E-12
45	9.9850E-01	1.4990E-03	4.2401E-06	4.7240E-09	4.4524E-12
46	9.9838E-01	1.6135E-03	5.0089E-06	6.2944E-09	6.9966E-12
47	9.9827E-01	1.7278E-03	5.8413E-06	8.1771E-09	1.0495E-11
48	9.9815E-01	1.8420E-03	6.7371E-06	1.0400E-08	1.5159E-11
49	9.9804E-01	1.9560E-03	7.6963E-06	1.2991E-08	2.1223E-11
50	9.9792E-01	2.0700E-03	8.7187E-06	1.5978E-08	2.8941E-11
51	9.9781E-01	2.1838E-03	9.8043E-06	1.9388E-08	3.8588E-11
52	9.9769E-01	2.2974E-03	1.0953E-05	2.3249E-08	5.0461E-11
53	9.9758E-01	2.4110E-03	1.2165E-05	2.7588E-08	6.4878E-11
54	9.9746E-01	2.5244E-03	1.3439E-05	3.2433E-08	8.2178E-11
55	9.9735E-01	2.6377E-03	1.4779E-05	3.7828E-08	1.0280E-10
56	9.9861E-01	1.3845E-03	3.5348E-06	3.4378E-09	2.6715E-12
57	9.9850E-01	1.4990E-03	4.2395E-06	4.7220E-09	4.4490E-12
58	9.9838E-01	1.6134E-03	5.0086E-06	6.2932E-09	6.9967E-12
59	9.9827E-01	1.7277E-03	5.8409E-06	8.1759E-09	1.0495E-11
60	9.9815E-01	1.8419E-03	6.7367E-06	1.0399E-08	1.5159E-11
61	9.9804E-01	1.9560E-03	7.6959E-06	1.2989E-08	2.1223E-11
62	9.9792E-01	2.0699E-03	8.7183E-06	1.5976E-08	2.8941E-11
63	9.9781E-01	2.1837E-03	9.8039E-06	1.9386E-08	3.8588E-11
64	9.9769E-01	2.2974E-03	1.0953E-05	2.3246E-08	5.0461E-11
65	9.9758E-01	2.4109E-03	1.2164E-05	2.7585E-08	6.4878E-11
66	9.9746E-01	2.5243E-03	1.3439E-05	3.2430E-08	8.2179E-11
67	9.9735E-01	2.6376E-03	1.4776E-05	3.7808E-08	1.0272E-10
68	9.9723E-01	2.7508E-03	1.6177E-05	4.3747E-08	1.2689E-10
69	9.9712E-01	2.8638E-03	1.7640E-05	5.0272E-08	1.5509E-10

$C_i(T)$ (continued)					
70	9.9700E-01	2.9767E-03	1.9165E-05	5.7412E-08	1.8774E-10
71	9.9689E-01	3.0895E-03	2.0756E-05	6.5221E-08	2.2547E-10
72	9.9838E-01	1.6134E-03	5.0086E-06	6.2933E-09	6.9968E-12
73	9.9827E-01	1.7277E-03	5.8401E-06	8.1725E-09	1.0487E-11
74	9.9815E-01	1.8418E-03	6.7363E-06	1.0397E-08	1.5160E-11
75	9.9804E-01	1.9559E-03	7.6955E-06	1.2988E-08	2.1223E-11
76	9.9792E-01	2.0698E-03	8.7178E-06	1.5974E-08	2.8941E-11
77	9.9781E-01	2.1836E-03	9.8034E-06	1.9383E-08	3.8588E-11
78	9.9769E-01	2.2973E-03	1.0952E-05	2.3244E-08	5.0461E-11
79	9.9758E-01	2.4109E-03	1.2164E-05	2.7583E-08	6.4878E-11
80	9.9746E-01	2.5243E-03	1.3438E-05	3.2427E-08	8.2179E-11
81	9.9735E-01	2.6376E-03	1.4776E-05	3.7805E-08	1.0272E-10
82	9.9723E-01	2.7507E-03	1.6176E-05	4.3743E-08	1.2689E-10
83	9.9712E-01	2.8638E-03	1.7639E-05	5.0268E-08	1.5509E-10
84	9.9700E-01	2.9767E-03	1.9165E-05	5.7408E-08	1.8774E-10
85	9.9689E-01	3.0895E-03	2.0753E-05	6.5189E-08	2.2529E-10
86	9.9678E-01	3.2021E-03	2.2403E-05	7.3638E-08	2.6820E-10
87	9.9666E-01	3.3146E-03	2.4116E-05	8.2783E-08	3.1697E-10
88	9.9655E-01	3.4271E-03	2.5891E-05	9.2649E-08	3.7209E-10
89	9.9643E-01	3.5394E-03	2.7732E-05	1.0331E-07	4.3445E-10
90	9.9815E-01	1.8418E-03	6.7363E-06	1.0397E-08	1.5160E-11
91	9.9804E-01	1.9559E-03	7.6944E-06	1.2982E-08	2.1207E-11
92	9.9792E-01	2.0698E-03	8.7174E-06	1.5972E-08	2.8941E-11
93	9.9781E-01	2.1836E-03	9.8030E-06	1.9381E-08	3.8588E-11
94	9.9769E-01	2.2972E-03	1.0952E-05	2.3241E-08	5.0461E-11
95	9.9758E-01	2.4108E-03	1.2163E-05	2.7580E-08	6.4878E-11
96	9.9746E-01	2.5242E-03	1.3438E-05	3.2424E-08	8.2179E-11
97	9.9735E-01	2.6375E-03	1.4775E-05	3.7801E-08	1.0272E-10
98	9.9723E-01	2.7507E-03	1.6176E-05	4.3739E-08	1.2689E-10
99	9.9712E-01	2.8637E-03	1.7638E-05	5.0263E-08	1.5509E-10
100	9.9700E-01	2.9766E-03	1.9164E-05	5.7403E-08	1.8774E-10
101	9.9689E-01	3.0894E-03	2.0752E-05	6.5183E-08	2.2529E-10
102	9.9678E-01	3.2021E-03	2.2402E-05	7.3632E-08	2.6820E-10
103	9.9666E-01	3.3146E-03	2.4115E-05	8.2777E-08	3.1697E-10
104	9.9655E-01	3.4270E-03	2.5890E-05	9.2642E-08	3.7209E-10
105	9.9643E-01	3.5393E-03	2.7728E-05	1.0326E-07	4.3411E-10



$C_i(T)$ (continued)					
106	9.9632E-01	3.6514E-03	2.9627E-05	1.1465E-07	5.0357E-10
107	9.9620E-01	3.7634E-03	3.1588E-05	1.2684E-07	5.8104E-10
108	9.9609E-01	3.8753E-03	3.3612E-05	1.3985E-07	6.6711E-10
109	9.9598E-01	3.9871E-03	3.5702E-05	1.5379E-07	7.6302E-10
110	9.9792E-01	2.0698E-03	8.7174E-06	1.5972E-08	2.8941E-11
111	9.9781E-01	2.1836E-03	9.8030E-06	1.9381E-08	3.8588E-11
112	9.9769E-01	2.2972E-03	1.0952E-05	2.3241E-08	5.0461E-11
113	9.9758E-01	2.4108E-03	1.2163E-05	2.7580E-08	6.4878E-11
114	9.9746E-01	2.5242E-03	1.3438E-05	3.2424E-08	8.2179E-11
115	9.9735E-01	2.6375E-03	1.4775E-05	3.7801E-08	1.0272E-10
116	9.9723E-01	2.7507E-03	1.6176E-05	4.3739E-08	1.2689E-10
117	9.9712E-01	2.8637E-03	1.7638E-05	5.0263E-08	1.5509E-10
118	9.9700E-01	2.9766E-03	1.9164E-05	5.7403E-08	1.8774E-10
119	9.9689E-01	3.0894E-03	2.0752E-05	6.5183E-08	2.2529E-10
120	9.9678E-01	3.2021E-03	2.2402E-05	7.3632E-08	2.6820E-10
121	9.9666E-01	3.3146E-03	2.4115E-05	8.2777E-08	3.1697E-10
122	9.9655E-01	3.4270E-03	2.5890E-05	9.2642E-08	3.7209E-10
123	9.9643E-01	3.5393E-03	2.7728E-05	1.0326E-07	4.3411E-10
124	9.9632E-01	3.6514E-03	2.9627E-05	1.1465E-07	5.0357E-10
125	9.9620E-01	3.7634E-03	3.1588E-05	1.2684E-07	5.8104E-10
126	9.9609E-01	3.8753E-03	3.3612E-05	1.3985E-07	6.6712E-10
127	9.9598E-01	3.9871E-03	3.5697E-05	1.5373E-07	7.6242E-10
128	9.9586E-01	4.0988E-03	3.7844E-05	1.6848E-07	8.6757E-10
129	9.9575E-01	4.2103E-03	4.0058E-05	1.8421E-07	9.8403E-10

# Appendix G – DTMC Spreadsheet

## G.a Case-0

	Fault			Pkt Len								
	Fault 1	99		368	Bytes							
	Fault 2	99		2944	Bits							
	Fault 3	99		BER	1.0E-06							
	Fault 4	99										
	Fault 5	99		Max Hops	38							
	Fault 6	99		MER	0.0029							
MER	MDR	Hops	Pr(0)	Pr(1)	Pr(2)	Pr(3)	1-SUM of Probs	Hops (h)	n(h)	Q(h)		
0.0029	0.9971	0	0.00	0.00	1.00	0.00	0.00E+00	0	0			
0.0029	0.9971	1	8.6417E-06	2.9310E-03	2.9310E-03	0.994129	0.00E+00	1	1	8.6417E-06		
0.0029	0.9971	2	8.6670E-06	8.5909E-06	5.8620E-03	0.994121	0.00E+00	2	1	8.6670E-06		
0.0029	0.9971	3	8.7177E-06	1.7182E-05	2.9482E-03	0.997026	0.00E+00	3	2	8.7177E-06		
0.0029	0.9971	4	8.7431E-06	8.6412E-06	2.9568E-03	0.997026	0.00E+00	4	2	8.7431E-06		
0.0029	0.9971	5	8.7687E-06	8.6663E-06	2.9482E-03	0.997034	0.00E+00	5	2	8.7687E-06		
0.0029	0.9971	6	8.7942E-06	8.6414E-06	2.9483E-03	0.997034	0.00E+00	6	2	8.7942E-06		
0.0029	0.9971	7	8.8196E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	7	3	8.8196E-06		
0.0029	0.9971	8	8.8451E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	8	3	8.8451E-06		
0.0029	0.9971	9	8.8706E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	9	3	8.8706E-06		
0.0029	0.9971	10	8.8961E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	10	3	8.8961E-06		
0.0029	0.9971	11	8.9216E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	11	4	8.9216E-06		
0.0029	0.9971	12	8.9470E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	12	4	8.9470E-06		
0.0029	0.9971	13	8.9725E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	13	4	8.9725E-06		
0.0029	0.9971	14	8.9980E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	14	4	8.9980E-06		
0.0029	0.9971	15	9.0235E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	15	5	9.0235E-06		
0.0029	0.9971	16	9.0489E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	16	5	9.0489E-06		
0.0029	0.9971	17	9.0744E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	17	5	9.0744E-06		
0.0029	0.9971	18	9.0999E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	18	5	9.0999E-06		
0.0029	0.9971	19	9.1254E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	19	6	9.1254E-06		
0.0029	0.9971	20	9.1509E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	20	6	9.1509E-06		
0.0029	0.9971	21	9.1763E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	21	5	9.1763E-06		
0.0029	0.9971	22	9.2018E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	22	5	9.2018E-06		
0.0029	0.9971	23	9.2273E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	23	5	9.2273E-06		
0.0029	0.9971	24	9.2528E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	24	5	9.2528E-06		
0.0029	0.9971	25	9.2782E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	25	4	9.2782E-06		
0.0029	0.9971	26	9.3037E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	26	4	9.3037E-06		
0.0029	0.9971	27	9.3292E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	27	4	9.3292E-06		
0.0029	0.9971	28	9.3547E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	28	4	9.3547E-06		
0.0029	0.9971	29	9.3802E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	29	3	9.3802E-06		
0.0029	0.9971	30	9.4056E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	30	3	9.4056E-06		
0.0029	0.9971	31	9.4311E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	31	3	9.4311E-06		
0.0029	0.9971	32	9.4566E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	32	3	9.4566E-06		
0.0029	0.9971	33	9.4821E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	33	2	9.4821E-06		
0.0029	0.9971	34	9.5075E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	34	2	9.5075E-06		
0.0029	0.9971	35	9.5330E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	35	2	9.5330E-06		
0.0029	0.9971	36	9.5585E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	36	2	9.5585E-06		
0.0029	0.9971	37	9.5840E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	37	1	9.5840E-06		
0.0029	0.9971	38	9.6095E-06	8.6414E-06	2.9482E-03	0.997034	0.00E+00	38	1	9.6095E-06		
0.0029	0.9971	39	9.6349E-06	8.6414E-06	2.9482E-03	0.997033	0.00E+00	39	0	9.6349E-06		
0.0029	0.9971	40	9.6349E-06	0.0000E+00	8.6414E-06	0.000000	1.00E+00	40	0	9.6349E-06		
									128			







## Appendix H – DTMC Results, $q_i(d(T))$

### H.a Case-0 DTMC Results, $q_i(d(T))$

Tank	Hops	Depth	0 Faults	1 Fault	2 Faults	3 Faults	4 Faults
0	0	0	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
1	0	1	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
2	1	2	8.6417E-06	2.9397E-03	2.9397E-03	2.9397E-03	2.9397E-03
3	2	3	8.6670E-06	2.9397E-03	2.9397E-03	2.9397E-03	2.9397E-03
4	3	4	8.7177E-06	2.9483E-03	5.8707E-03	5.8707E-03	5.8707E-03
5	4	5	8.7431E-06	2.9483E-03	5.8707E-03	5.8707E-03	5.8707E-03
6	3	4	8.7177E-06	2.9483E-03	5.8707E-03	5.8707E-03	5.8707E-03
7	4	5	8.7431E-06	2.9483E-03	5.8707E-03	5.8707E-03	5.8707E-03
8	5	6	8.7687E-06	2.9484E-03	5.8793E-03	8.7931E-03	8.7931E-03
9	6	7	8.7942E-06	2.9484E-03	5.8793E-03	8.7931E-03	8.7931E-03
10	7	8	8.8196E-06	2.9484E-03	5.8794E-03	8.8017E-03	1.1707E-02
11	8	9	8.8451E-06	2.9484E-03	5.8794E-03	8.8017E-03	1.1707E-02
12	5	6	8.7687E-06	2.9484E-03	5.8793E-03	8.7931E-03	8.7931E-03
13	6	7	8.7942E-06	2.9484E-03	5.8793E-03	8.7931E-03	8.7931E-03
14	7	8	8.8196E-06	2.9484E-03	5.8794E-03	8.8017E-03	1.1707E-02
15	8	9	8.8451E-06	2.9484E-03	5.8794E-03	8.8017E-03	1.1707E-02
16	9	10	8.8706E-06	2.9485E-03	5.8794E-03	8.8018E-03	1.1715E-02
17	10	11	8.8961E-06	2.9485E-03	5.8794E-03	8.8018E-03	1.1716E-02
18	11	12	8.9216E-06	2.9485E-03	5.8795E-03	8.8018E-03	1.1716E-02
19	12	13	8.9470E-06	2.9485E-03	5.8795E-03	8.8018E-03	1.1716E-02
20	7	8	8.8196E-06	2.9484E-03	5.8794E-03	8.8017E-03	1.1707E-02
21	8	9	8.8451E-06	2.9484E-03	5.8794E-03	8.8017E-03	1.1707E-02
22	9	10	8.8706E-06	2.9485E-03	5.8794E-03	8.8018E-03	1.1715E-02
23	10	11	8.8961E-06	2.9485E-03	5.8794E-03	8.8018E-03	1.1716E-02
24	11	12	8.9216E-06	2.9485E-03	5.8795E-03	8.8018E-03	1.1716E-02
25	12	13	8.9470E-06	2.9485E-03	5.8795E-03	8.8018E-03	1.1716E-02
26	13	14	8.9725E-06	2.9486E-03	5.8795E-03	8.8019E-03	1.1716E-02
27	14	15	8.9980E-06	2.9486E-03	5.8795E-03	8.8019E-03	1.1716E-02
28	15	16	9.0235E-06	2.9486E-03	5.8796E-03	8.8019E-03	1.1716E-02
29	16	17	9.0489E-06	2.9486E-03	5.8796E-03	8.8019E-03	1.1716E-02
30	9	10	8.8706E-06	2.9485E-03	5.8794E-03	8.8018E-03	1.1715E-02
31	10	11	8.8961E-06	2.9485E-03	5.8794E-03	8.8018E-03	1.1716E-02

<i>Case0 - <math>q_i(d(T))</math> (continued)</i>							
32	11	12	8.9216E-06	2.9485E-03	5.8795E-03	8.8018E-03	1.1716E-02
33	12	13	8.9470E-06	2.9485E-03	5.8795E-03	8.8018E-03	1.1716E-02
34	13	14	8.9725E-06	2.9486E-03	5.8795E-03	8.8019E-03	1.1716E-02
35	14	15	8.9980E-06	2.9486E-03	5.8795E-03	8.8019E-03	1.1716E-02
36	15	16	9.0235E-06	2.9486E-03	5.8796E-03	8.8019E-03	1.1716E-02
37	16	17	9.0489E-06	2.9486E-03	5.8796E-03	8.8019E-03	1.1716E-02
38	17	18	9.0744E-06	2.9487E-03	5.8796E-03	8.8020E-03	1.1716E-02
39	18	19	9.0999E-06	2.9487E-03	5.8796E-03	8.8020E-03	1.1716E-02
40	19	20	9.1254E-06	2.9487E-03	5.8797E-03	8.8020E-03	1.1716E-02
41	20	21	9.1509E-06	2.9487E-03	5.8797E-03	8.8020E-03	1.1716E-02
42	11	12	8.9216E-06	2.9485E-03	5.8795E-03	8.8018E-03	1.1716E-02
43	12	13	8.9470E-06	2.9485E-03	5.8795E-03	8.8018E-03	1.1716E-02
44	13	14	8.9725E-06	2.9486E-03	5.8795E-03	8.8019E-03	1.1716E-02
45	14	15	8.9980E-06	2.9486E-03	5.8795E-03	8.8019E-03	1.1716E-02
46	15	16	9.0235E-06	2.9486E-03	5.8796E-03	8.8019E-03	1.1716E-02
47	16	17	9.0489E-06	2.9486E-03	5.8796E-03	8.8019E-03	1.1716E-02
48	17	18	9.0744E-06	2.9487E-03	5.8796E-03	8.8020E-03	1.1716E-02
49	18	19	9.0999E-06	2.9487E-03	5.8796E-03	8.8020E-03	1.1716E-02
50	19	20	9.1254E-06	2.9487E-03	5.8797E-03	8.8020E-03	1.1716E-02
51	20	21	9.1509E-06	2.9487E-03	5.8797E-03	8.8020E-03	1.1716E-02
52	21	22	9.1763E-06	2.9488E-03	5.8797E-03	8.8021E-03	1.1716E-02
53	22	23	9.2018E-06	2.9488E-03	5.8797E-03	8.8021E-03	1.1716E-02
54	23	24	9.2273E-06	2.9488E-03	5.8798E-03	8.8021E-03	1.1716E-02
55	24	25	9.2528E-06	2.9488E-03	5.8798E-03	8.8021E-03	1.1716E-02
56	13	14	8.9725E-06	2.9486E-03	5.8795E-03	8.8019E-03	1.1716E-02
57	14	15	8.9980E-06	2.9486E-03	5.8795E-03	8.8019E-03	1.1716E-02
58	15	16	9.0235E-06	2.9486E-03	5.8796E-03	8.8019E-03	1.1716E-02
59	16	17	9.0489E-06	2.9486E-03	5.8796E-03	8.8019E-03	1.1716E-02
60	17	18	9.0744E-06	2.9487E-03	5.8796E-03	8.8020E-03	1.1716E-02
61	18	19	9.0999E-06	2.9487E-03	5.8796E-03	8.8020E-03	1.1716E-02
62	19	20	9.1254E-06	2.9487E-03	5.8797E-03	8.8020E-03	1.1716E-02
63	20	21	9.1509E-06	2.9487E-03	5.8797E-03	8.8020E-03	1.1716E-02
64	21	22	9.1763E-06	2.9488E-03	5.8797E-03	8.8021E-03	1.1716E-02
65	22	23	9.2018E-06	2.9488E-03	5.8797E-03	8.8021E-03	1.1716E-02
66	23	24	9.2273E-06	2.9488E-03	5.8798E-03	8.8021E-03	1.1716E-02
67	24	25	9.2528E-06	2.9488E-03	5.8798E-03	8.8021E-03	1.1716E-02

<i>Case0 - <math>q_i(d(T))</math> (continued)</i>							
68	25	26	9.2782E-06	2.9489E-03	5.8798E-03	8.8022E-03	1.1716E-02
69	26	27	9.3037E-06	2.9489E-03	5.8799E-03	8.8022E-03	1.1716E-02
70	27	28	9.3292E-06	2.9489E-03	5.8799E-03	8.8022E-03	1.1716E-02
71	28	29	9.3547E-06	2.9489E-03	5.8799E-03	8.8022E-03	1.1716E-02
72	15	16	9.0235E-06	2.9486E-03	5.8796E-03	8.8019E-03	1.1716E-02
73	16	17	9.0489E-06	2.9486E-03	5.8796E-03	8.8019E-03	1.1716E-02
74	17	18	9.0744E-06	2.9487E-03	5.8796E-03	8.8020E-03	1.1716E-02
75	18	19	9.0999E-06	2.9487E-03	5.8796E-03	8.8020E-03	1.1716E-02
76	19	20	9.1254E-06	2.9487E-03	5.8797E-03	8.8020E-03	1.1716E-02
77	20	21	9.1509E-06	2.9487E-03	5.8797E-03	8.8020E-03	1.1716E-02
78	21	22	9.1763E-06	2.9488E-03	5.8797E-03	8.8021E-03	1.1716E-02
79	22	23	9.2018E-06	2.9488E-03	5.8797E-03	8.8021E-03	1.1716E-02
80	23	24	9.2273E-06	2.9488E-03	5.8798E-03	8.8021E-03	1.1716E-02
81	24	25	9.2528E-06	2.9488E-03	5.8798E-03	8.8021E-03	1.1716E-02
82	25	26	9.2782E-06	2.9489E-03	5.8798E-03	8.8022E-03	1.1716E-02
83	26	27	9.3037E-06	2.9489E-03	5.8799E-03	8.8022E-03	1.1716E-02
84	27	28	9.3292E-06	2.9489E-03	5.8799E-03	8.8022E-03	1.1716E-02
85	28	29	9.3547E-06	2.9489E-03	5.8799E-03	8.8022E-03	1.1716E-02
86	29	30	9.3802E-06	2.9490E-03	5.8799E-03	8.8023E-03	1.1716E-02
87	30	31	9.4056E-06	2.9490E-03	5.8800E-03	8.8023E-03	1.1716E-02
88	31	32	9.4311E-06	2.9490E-03	5.8800E-03	8.8023E-03	1.1716E-02
89	32	33	9.4566E-06	2.9491E-03	5.8800E-03	8.8023E-03	1.1716E-02
90	17	18	9.0744E-06	2.9487E-03	5.8796E-03	8.8020E-03	1.1716E-02
91	18	19	9.0999E-06	2.9487E-03	5.8796E-03	8.8020E-03	1.1716E-02
92	19	20	9.1254E-06	2.9487E-03	5.8797E-03	8.8020E-03	1.1716E-02
93	20	21	9.1509E-06	2.9487E-03	5.8797E-03	8.8020E-03	1.1716E-02
94	21	22	9.1763E-06	2.9488E-03	5.8797E-03	8.8021E-03	1.1716E-02
95	22	23	9.2018E-06	2.9488E-03	5.8797E-03	8.8021E-03	1.1716E-02
96	23	24	9.2273E-06	2.9488E-03	5.8798E-03	8.8021E-03	1.1716E-02
97	24	25	9.2528E-06	2.9488E-03	5.8798E-03	8.8021E-03	1.1716E-02
98	25	26	9.2782E-06	2.9489E-03	5.8798E-03	8.8022E-03	1.1716E-02
99	26	27	9.3037E-06	2.9489E-03	5.8799E-03	8.8022E-03	1.1716E-02
100	27	28	9.3292E-06	2.9489E-03	5.8799E-03	8.8022E-03	1.1716E-02
101	28	29	9.3547E-06	2.9489E-03	5.8799E-03	8.8022E-03	1.1716E-02
102	29	30	9.3802E-06	2.9490E-03	5.8799E-03	8.8023E-03	1.1716E-02
103	30	31	9.4056E-06	2.9490E-03	5.8800E-03	8.8023E-03	1.1716E-02



<i>Case0 - <math>q_i(d(T))</math> (continued)</i>							
104	31	32	9.4311E-06	2.9490E-03	5.8800E-03	8.8023E-03	1.1716E-02
105	32	33	9.4566E-06	2.9491E-03	5.8800E-03	8.8023E-03	1.1716E-02
106	33	34	9.4821E-06	2.9491E-03	5.8800E-03	8.8024E-03	1.1716E-02
107	34	35	9.5075E-06	2.9491E-03	5.8801E-03	8.8024E-03	1.1716E-02
108	35	36	9.5330E-06	2.9491E-03	5.8801E-03	8.8024E-03	1.1716E-02
109	36	37	9.5585E-06	2.9492E-03	5.8801E-03	8.8024E-03	1.1716E-02
110	19	20	9.1254E-06	2.9487E-03	5.8797E-03	8.8020E-03	1.1716E-02
111	20	21	9.1509E-06	2.9487E-03	5.8797E-03	8.8020E-03	1.1716E-02
112	21	22	9.1763E-06	2.9488E-03	5.8797E-03	8.8021E-03	1.1716E-02
113	22	23	9.2018E-06	2.9488E-03	5.8797E-03	8.8021E-03	1.1716E-02
114	23	24	9.2273E-06	2.9488E-03	5.8798E-03	8.8021E-03	1.1716E-02
115	24	25	9.2528E-06	2.9488E-03	5.8798E-03	8.8021E-03	1.1716E-02
116	25	26	9.2782E-06	2.9489E-03	5.8798E-03	8.8022E-03	1.1716E-02
117	26	27	9.3037E-06	2.9489E-03	5.8799E-03	8.8022E-03	1.1716E-02
118	27	28	9.3292E-06	2.9489E-03	5.8799E-03	8.8022E-03	1.1716E-02
119	28	29	9.3547E-06	2.9489E-03	5.8799E-03	8.8022E-03	1.1716E-02
120	29	30	9.3802E-06	2.9490E-03	5.8799E-03	8.8023E-03	1.1716E-02
121	30	31	9.4056E-06	2.9490E-03	5.8800E-03	8.8023E-03	1.1716E-02
122	31	32	9.4311E-06	2.9490E-03	5.8800E-03	8.8023E-03	1.1716E-02
123	32	33	9.4566E-06	2.9491E-03	5.8800E-03	8.8023E-03	1.1716E-02
124	33	34	9.4821E-06	2.9491E-03	5.8800E-03	8.8024E-03	1.1716E-02
125	34	35	9.5075E-06	2.9491E-03	5.8801E-03	8.8024E-03	1.1716E-02
126	35	36	9.5330E-06	2.9491E-03	5.8801E-03	8.8024E-03	1.1716E-02
127	36	37	9.5585E-06	2.9492E-03	5.8801E-03	8.8024E-03	1.1716E-02
128	37	38	9.5840E-06	2.9492E-03	5.8801E-03	8.8025E-03	1.1716E-02
129	38	39	9.6095E-06	2.9492E-03	5.8802E-03	8.8025E-03	1.1716E-02

### H.b Macro DTMC Results, $q_i(d(T))$

Tank	Hops	Depth	0 Faults	1 Fault	2 Faults	3 Faults	4 Faults
0	0	0	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
1	0	1	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
2	1	2	7.5395E-11	9.4857E-06	9.4857E-06	9.4857E-06	9.4857E-06
3	2	3	7.5836E-11	9.4857E-06	9.4857E-06	9.4857E-06	9.4857E-06
4	3	4	7.6721E-11	9.5391E-06	3.7826E-05	3.7826E-05	3.7826E-05
5	4	5	7.7168E-11	9.5393E-06	3.7826E-05	3.7826E-05	3.7826E-05
6	3	4	7.6721E-11	9.5391E-06	3.7826E-05	3.7826E-05	3.7826E-05
7	4	5	7.7168E-11	9.5393E-06	3.7826E-05	3.7826E-05	3.7826E-05
8	5	6	7.7618E-11	9.5396E-06	3.7932E-05	8.4846E-05	8.4846E-05
9	6	7	7.8068E-11	9.5398E-06	3.7932E-05	8.4846E-05	8.4846E-05
10	7	8	7.8519E-11	9.5399E-06	3.7933E-05	8.5005E-05	1.5037E-04
11	8	9	7.8971E-11	9.5401E-06	3.7933E-05	8.5005E-05	1.5037E-04
12	5	6	7.7618E-11	9.5396E-06	3.7932E-05	8.4846E-05	8.4846E-05
13	6	7	7.8068E-11	9.5398E-06	3.7932E-05	8.4846E-05	8.4846E-05
14	7	8	7.8519E-11	9.5399E-06	3.7933E-05	8.5005E-05	1.5037E-04
15	8	9	7.8971E-11	9.5401E-06	3.7933E-05	8.5005E-05	1.5037E-04
16	9	10	7.9425E-11	9.5402E-06	3.7934E-05	8.5006E-05	1.5058E-04
17	10	11	7.9880E-11	9.5404E-06	3.7934E-05	8.5007E-05	1.5058E-04
18	11	12	8.0336E-11	9.5405E-06	3.7934E-05	8.5007E-05	1.5059E-04
19	12	13	8.0794E-11	9.5407E-06	3.7935E-05	8.5008E-05	1.5059E-04
20	7	8	7.8519E-11	9.5399E-06	3.7933E-05	8.5005E-05	1.5037E-04
21	8	9	7.8971E-11	9.5401E-06	3.7933E-05	8.5005E-05	1.5037E-04
22	9	10	7.9425E-11	9.5402E-06	3.7934E-05	8.5006E-05	1.5058E-04
23	10	11	7.9880E-11	9.5404E-06	3.7934E-05	8.5007E-05	1.5058E-04
24	11	12	8.0336E-11	9.5405E-06	3.7934E-05	8.5007E-05	1.5059E-04
25	12	13	8.0794E-11	9.5407E-06	3.7935E-05	8.5008E-05	1.5059E-04
26	13	14	8.1253E-11	9.5409E-06	3.7935E-05	8.5008E-05	1.5059E-04
27	14	15	8.1713E-11	9.5410E-06	3.7935E-05	8.5009E-05	1.5059E-04
28	15	16	8.2174E-11	9.5412E-06	3.7936E-05	8.5009E-05	1.5059E-04
29	16	17	8.2637E-11	9.5413E-06	3.7936E-05	8.5009E-05	1.5059E-04
30	9	10	7.9425E-11	9.5402E-06	3.7934E-05	8.5006E-05	1.5058E-04
31	10	11	7.9880E-11	9.5404E-06	3.7934E-05	8.5007E-05	1.5058E-04
32	11	12	8.0336E-11	9.5405E-06	3.7934E-05	8.5007E-05	1.5059E-04

<i>Macro - <math>q_i(d(T))</math> (continued)</i>							
33	12	13	8.0794E-11	9.5407E-06	3.7935E-05	8.5008E-05	1.5059E-04
34	13	14	8.1253E-11	9.5409E-06	3.7935E-05	8.5008E-05	1.5059E-04
35	14	15	8.1713E-11	9.5410E-06	3.7935E-05	8.5009E-05	1.5059E-04
36	15	16	8.2174E-11	9.5412E-06	3.7936E-05	8.5009E-05	1.5059E-04
37	16	17	8.2637E-11	9.5413E-06	3.7936E-05	8.5009E-05	1.5059E-04
38	17	18	8.3101E-11	9.5415E-06	3.7936E-05	8.5010E-05	1.5059E-04
39	18	19	8.3567E-11	9.5416E-06	3.7937E-05	8.5010E-05	1.5059E-04
40	19	20	8.4033E-11	9.5418E-06	3.7937E-05	8.5011E-05	1.5059E-04
41	20	21	8.4501E-11	9.5420E-06	3.7937E-05	8.5011E-05	1.5059E-04
42	11	12	8.0336E-11	9.5405E-06	3.7934E-05	8.5007E-05	1.5059E-04
43	12	13	8.0794E-11	9.5407E-06	3.7935E-05	8.5008E-05	1.5059E-04
44	13	14	8.1253E-11	9.5409E-06	3.7935E-05	8.5008E-05	1.5059E-04
45	14	15	8.1713E-11	9.5410E-06	3.7935E-05	8.5009E-05	1.5059E-04
46	15	16	8.2174E-11	9.5412E-06	3.7936E-05	8.5009E-05	1.5059E-04
47	16	17	8.2637E-11	9.5413E-06	3.7936E-05	8.5009E-05	1.5059E-04
48	17	18	8.3101E-11	9.5415E-06	3.7936E-05	8.5010E-05	1.5059E-04
49	18	19	8.3567E-11	9.5416E-06	3.7937E-05	8.5010E-05	1.5059E-04
50	19	20	8.4033E-11	9.5418E-06	3.7937E-05	8.5011E-05	1.5059E-04
51	20	21	8.4501E-11	9.5420E-06	3.7937E-05	8.5011E-05	1.5059E-04
52	21	22	8.4971E-11	9.5421E-06	3.7937E-05	8.5012E-05	1.5059E-04
53	22	23	8.5441E-11	9.5423E-06	3.7938E-05	8.5012E-05	1.5059E-04
54	23	24	8.5913E-11	9.5424E-06	3.7938E-05	8.5013E-05	1.5059E-04
55	24	25	8.6386E-11	9.5426E-06	3.7938E-05	8.5013E-05	1.5059E-04
56	13	14	8.1253E-11	9.5409E-06	3.7935E-05	8.5008E-05	1.5059E-04
57	14	15	8.1713E-11	9.5410E-06	3.7935E-05	8.5009E-05	1.5059E-04
58	15	16	8.2174E-11	9.5412E-06	3.7936E-05	8.5009E-05	1.5059E-04
59	16	17	8.2637E-11	9.5413E-06	3.7936E-05	8.5009E-05	1.5059E-04
60	17	18	8.3101E-11	9.5415E-06	3.7936E-05	8.5010E-05	1.5059E-04
61	18	19	8.3567E-11	9.5416E-06	3.7937E-05	8.5010E-05	1.5059E-04
62	19	20	8.4033E-11	9.5418E-06	3.7937E-05	8.5011E-05	1.5059E-04
63	20	21	8.4501E-11	9.5420E-06	3.7937E-05	8.5011E-05	1.5059E-04
64	21	22	8.4971E-11	9.5421E-06	3.7937E-05	8.5012E-05	1.5059E-04
65	22	23	8.5441E-11	9.5423E-06	3.7938E-05	8.5012E-05	1.5059E-04
66	23	24	8.5913E-11	9.5424E-06	3.7938E-05	8.5013E-05	1.5059E-04
67	24	25	8.6386E-11	9.5426E-06	3.7938E-05	8.5013E-05	1.5059E-04
68	25	26	8.6861E-11	9.5427E-06	3.7939E-05	8.5014E-05	1.5059E-04

<i>Macro - <math>q_i(d(T))</math> (continued)</i>							
69	26	27	8.7336E-11	9.5429E-06	3.7939E-05	8.5014E-05	1.5060E-04
70	27	28	8.7814E-11	9.5431E-06	3.7939E-05	8.5015E-05	1.5060E-04
71	28	29	8.8292E-11	9.5432E-06	3.7940E-05	8.5015E-05	1.5060E-04
72	15	16	8.2174E-11	9.5412E-06	3.7936E-05	8.5009E-05	1.5059E-04
73	16	17	8.2637E-11	9.5413E-06	3.7936E-05	8.5009E-05	1.5059E-04
74	17	18	8.3101E-11	9.5415E-06	3.7936E-05	8.5010E-05	1.5059E-04
75	18	19	8.3567E-11	9.5416E-06	3.7937E-05	8.5010E-05	1.5059E-04
76	19	20	8.4033E-11	9.5418E-06	3.7937E-05	8.5011E-05	1.5059E-04
77	20	21	8.4501E-11	9.5420E-06	3.7937E-05	8.5011E-05	1.5059E-04
78	21	22	8.4971E-11	9.5421E-06	3.7937E-05	8.5012E-05	1.5059E-04
79	22	23	8.5441E-11	9.5423E-06	3.7938E-05	8.5012E-05	1.5059E-04
80	23	24	8.5913E-11	9.5424E-06	3.7938E-05	8.5013E-05	1.5059E-04
81	24	25	8.6386E-11	9.5426E-06	3.7938E-05	8.5013E-05	1.5059E-04
82	25	26	8.6861E-11	9.5427E-06	3.7939E-05	8.5014E-05	1.5059E-04
83	26	27	8.7336E-11	9.5429E-06	3.7939E-05	8.5014E-05	1.5060E-04
84	27	28	8.7814E-11	9.5431E-06	3.7939E-05	8.5015E-05	1.5060E-04
85	28	29	8.8292E-11	9.5432E-06	3.7940E-05	8.5015E-05	1.5060E-04
86	29	30	8.8772E-11	9.5434E-06	3.7940E-05	8.5016E-05	1.5060E-04
87	30	31	8.9253E-11	9.5435E-06	3.7940E-05	8.5016E-05	1.5060E-04
88	31	32	8.9735E-11	9.5437E-06	3.7941E-05	8.5016E-05	1.5060E-04
89	32	33	9.0218E-11	9.5438E-06	3.7941E-05	8.5017E-05	1.5060E-04
90	17	18	8.3101E-11	9.5415E-06	3.7936E-05	8.5010E-05	1.5059E-04
91	18	19	8.3567E-11	9.5416E-06	3.7937E-05	8.5010E-05	1.5059E-04
92	19	20	8.4033E-11	9.5418E-06	3.7937E-05	8.5011E-05	1.5059E-04
93	20	21	8.4501E-11	9.5420E-06	3.7937E-05	8.5011E-05	1.5059E-04
94	21	22	8.4971E-11	9.5421E-06	3.7937E-05	8.5012E-05	1.5059E-04
95	22	23	8.5441E-11	9.5423E-06	3.7938E-05	8.5012E-05	1.5059E-04
96	23	24	8.5913E-11	9.5424E-06	3.7938E-05	8.5013E-05	1.5059E-04
97	24	25	8.6386E-11	9.5426E-06	3.7938E-05	8.5013E-05	1.5059E-04
98	25	26	8.6861E-11	9.5427E-06	3.7939E-05	8.5014E-05	1.5059E-04
99	26	27	8.7336E-11	9.5429E-06	3.7939E-05	8.5014E-05	1.5060E-04
100	27	28	8.7814E-11	9.5431E-06	3.7939E-05	8.5015E-05	1.5060E-04
101	28	29	8.8292E-11	9.5432E-06	3.7940E-05	8.5015E-05	1.5060E-04
102	29	30	8.8772E-11	9.5434E-06	3.7940E-05	8.5016E-05	1.5060E-04
103	30	31	8.9253E-11	9.5435E-06	3.7940E-05	8.5016E-05	1.5060E-04
104	31	32	8.9735E-11	9.5437E-06	3.7941E-05	8.5016E-05	1.5060E-04

<i>Macro - <math>q_i(d(T))</math> (continued)</i>							
105	32	33	9.0218E-11	9.5438E-06	3.7941E-05	8.5017E-05	1.5060E-04
106	33	34	9.0703E-11	9.5440E-06	3.7941E-05	8.5017E-05	1.5060E-04
107	34	35	9.1189E-11	9.5442E-06	3.7942E-05	8.5018E-05	1.5060E-04
108	35	36	9.1677E-11	9.5443E-06	3.7942E-05	8.5018E-05	1.5060E-04
109	36	37	9.2166E-11	9.5445E-06	3.7942E-05	8.5019E-05	1.5060E-04
110	19	20	8.4033E-11	9.5418E-06	3.7937E-05	8.5011E-05	1.5059E-04
111	20	21	8.4501E-11	9.5420E-06	3.7937E-05	8.5011E-05	1.5059E-04
112	21	22	8.4971E-11	9.5421E-06	3.7937E-05	8.5012E-05	1.5059E-04
113	22	23	8.5441E-11	9.5423E-06	3.7938E-05	8.5012E-05	1.5059E-04
114	23	24	8.5913E-11	9.5424E-06	3.7938E-05	8.5013E-05	1.5059E-04
115	24	25	8.6386E-11	9.5426E-06	3.7938E-05	8.5013E-05	1.5059E-04
116	25	26	8.6861E-11	9.5427E-06	3.7939E-05	8.5014E-05	1.5059E-04
117	26	27	8.7336E-11	9.5429E-06	3.7939E-05	8.5014E-05	1.5060E-04
118	27	28	8.7814E-11	9.5431E-06	3.7939E-05	8.5015E-05	1.5060E-04
119	28	29	8.8292E-11	9.5432E-06	3.7940E-05	8.5015E-05	1.5060E-04
120	29	30	8.8772E-11	9.5434E-06	3.7940E-05	8.5016E-05	1.5060E-04
121	30	31	8.9253E-11	9.5435E-06	3.7940E-05	8.5016E-05	1.5060E-04
122	31	32	8.9735E-11	9.5437E-06	3.7941E-05	8.5016E-05	1.5060E-04
123	32	33	9.0218E-11	9.5438E-06	3.7941E-05	8.5017E-05	1.5060E-04
124	33	34	9.0703E-11	9.5440E-06	3.7941E-05	8.5017E-05	1.5060E-04
125	34	35	9.1189E-11	9.5442E-06	3.7942E-05	8.5018E-05	1.5060E-04
126	35	36	9.1677E-11	9.5443E-06	3.7942E-05	8.5018E-05	1.5060E-04
127	36	37	9.2166E-11	9.5445E-06	3.7942E-05	8.5019E-05	1.5060E-04
128	37	38	9.2656E-11	9.5446E-06	3.7942E-05	8.5019E-05	1.5060E-04
129	38	39	9.3147E-11	9.5448E-06	3.7943E-05	8.5020E-05	1.5060E-04

### H.c Micro DTMC Results, $q_i(d(T))$

Tank	Hops	Depth	0 Faults	1 Fault	2 Faults	3 Faults	4 Faults
0	0	0	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
1	0	1	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
2	1	2	7.4679E-11	8.6417E-06	8.6417E-06	8.6417E-06	8.6417E-06
3	2	3	7.4897E-11	8.6417E-06	8.6417E-06	8.6417E-06	8.6417E-06
4	3	4	7.5335E-11	8.6417E-06	1.7283E-05	1.7283E-05	1.7283E-05
5	4	5	7.5554E-11	8.6417E-06	1.7283E-05	1.7283E-05	1.7283E-05
6	3	4	7.5335E-11	8.6417E-06	1.7283E-05	1.7283E-05	1.7283E-05
7	4	5	7.5554E-11	8.6417E-06	1.7283E-05	1.7283E-05	1.7283E-05
8	5	6	7.5774E-11	8.6417E-06	1.7283E-05	2.5925E-05	2.5925E-05
9	6	7	7.5994E-11	8.6417E-06	1.7283E-05	2.5925E-05	2.5925E-05
10	7	8	7.6213E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
11	8	9	7.6433E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
12	5	6	7.5774E-11	8.6417E-06	1.7283E-05	2.5925E-05	2.5925E-05
13	6	7	7.5994E-11	8.6417E-06	1.7283E-05	2.5925E-05	2.5925E-05
14	7	8	7.6213E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
15	8	9	7.6433E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
16	9	10	7.6652E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
17	10	11	7.6872E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
18	11	12	7.7091E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
19	12	13	7.7311E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
20	7	8	7.6213E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
21	8	9	7.6433E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
22	9	10	7.6652E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
23	10	11	7.6872E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
24	11	12	7.7091E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
25	12	13	7.7311E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
26	13	14	7.7530E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
27	14	15	7.7750E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
28	15	16	7.7969E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
29	16	17	7.8189E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
30	9	10	7.6652E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
31	10	11	7.6872E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
32	11	12	7.7091E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
33	12	13	7.7311E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
34	13	14	7.7530E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05

<i>Micro - <math>q_i(d(T))</math></i> (continued)							
35	14	15	7.7750E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
36	15	16	7.7969E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
37	16	17	7.8189E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
38	17	18	7.8409E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
39	18	19	7.8628E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
40	19	20	7.8848E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
41	20	21	7.9067E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
42	11	12	7.7091E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
43	12	13	7.7311E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
44	13	14	7.7530E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
45	14	15	7.7750E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
46	15	16	7.7969E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
47	16	17	7.8189E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
48	17	18	7.8409E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
49	18	19	7.8628E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
50	19	20	7.8848E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
51	20	21	7.9067E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
52	21	22	7.9287E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
53	22	23	7.9506E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
54	23	24	7.9726E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
55	24	25	7.9945E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
56	13	14	7.7530E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
57	14	15	7.7750E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
58	15	16	7.7969E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
59	16	17	7.8189E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
60	17	18	7.8409E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
61	18	19	7.8628E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
62	19	20	7.8848E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
63	20	21	7.9067E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
64	21	22	7.9287E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
65	22	23	7.9506E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
66	23	24	7.9726E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
67	24	25	7.9945E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
68	25	26	8.0165E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
69	26	27	8.0384E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
70	27	28	8.0604E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05

<i>Micro - <math>q_i(d(T))</math></i> (continued)							
71	28	29	8.0823E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
72	15	16	7.7969E-11	8.6417E-06	1.7283E-05	2.5925E-05	3.4566E-05
73	16	17	7.8189E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
74	17	18	7.8409E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
75	18	19	7.8628E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
76	19	20	7.8848E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
77	20	21	7.9067E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
78	21	22	7.9287E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
79	22	23	7.9506E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
80	23	24	7.9726E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
81	24	25	7.9945E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
82	25	26	8.0165E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
83	26	27	8.0384E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
84	27	28	8.0604E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
85	28	29	8.0823E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
86	29	30	8.1043E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
87	30	31	8.1262E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
88	31	32	8.1482E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
89	32	33	8.1701E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
90	17	18	7.8409E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
91	18	19	7.8628E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
92	19	20	7.8848E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
93	20	21	7.9067E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
94	21	22	7.9287E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
95	22	23	7.9506E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
96	23	24	7.9726E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
97	24	25	7.9945E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
98	25	26	8.0165E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
99	26	27	8.0384E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
100	27	28	8.0604E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
101	28	29	8.0823E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
102	29	30	8.1043E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
103	30	31	8.1262E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
104	31	32	8.1482E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
105	32	33	8.1701E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
106	33	34	8.1921E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05



<i>Micro - <math>q_i(d(T))</math></i> (continued)							
107	34	35	8.2141E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
108	35	36	8.2360E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
109	36	37	8.2580E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
110	19	20	7.8848E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
111	20	21	7.9067E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
112	21	22	7.9287E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
113	22	23	7.9506E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
114	23	24	7.9726E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
115	24	25	7.9945E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
116	25	26	8.0165E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
117	26	27	8.0384E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
118	27	28	8.0604E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
119	28	29	8.0823E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
120	29	30	8.1043E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
121	30	31	8.1262E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
122	31	32	8.1482E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
123	32	33	8.1701E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
124	33	34	8.1921E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
125	34	35	8.2141E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
126	35	36	8.2360E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
127	36	37	8.2580E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
128	37	38	8.2799E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05
129	38	39	8.3019E-11	8.6418E-06	1.7283E-05	2.5925E-05	3.4566E-05

### H.d Combo DTMC Results, $q_i(d(T))$

Tank	Hops	Depth	0 Faults	1 Fault	2 Faults	3 Faults	4 Faults
0	0	0	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
1	0	1	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
2	1	2	5.5774E-21	7.5395E-11	7.5395E-11	7.5395E-11	7.5395E-11
3	2	3	5.6101E-21	7.5395E-11	7.5395E-11	7.5395E-11	7.5395E-11
4	3	4	5.6758E-21	7.5396E-11	3.0158E-10	3.0158E-10	3.0158E-10
5	4	5	5.7089E-21	7.5396E-11	3.0158E-10	3.0158E-10	3.0158E-10
6	3	4	5.6758E-21	7.5396E-11	3.0158E-10	3.0158E-10	3.0158E-10
7	4	5	5.7089E-21	7.5396E-11	3.0158E-10	3.0158E-10	3.0158E-10
8	5	6	5.7422E-21	7.5396E-11	3.0158E-10	6.7854E-10	6.7854E-10
9	6	7	5.7756E-21	7.5396E-11	3.0158E-10	6.7854E-10	6.7854E-10
10	7	8	5.8090E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
11	8	9	5.8425E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
12	5	6	5.7422E-21	7.5396E-11	3.0158E-10	6.7854E-10	6.7854E-10
13	6	7	5.7756E-21	7.5396E-11	3.0158E-10	6.7854E-10	6.7854E-10
14	7	8	5.8090E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
15	8	9	5.8425E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
16	9	10	5.8761E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
17	10	11	5.9098E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
18	11	12	5.9436E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
19	12	13	5.9775E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
20	7	8	5.8090E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
21	8	9	5.8425E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
22	9	10	5.8761E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
23	10	11	5.9098E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
24	11	12	5.9436E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
25	12	13	5.9775E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
26	13	14	6.0115E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
27	14	15	6.0456E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
28	15	16	6.0798E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
29	16	17	6.1141E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
30	9	10	5.8761E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
31	10	11	5.9098E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
32	11	12	5.9436E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
33	12	13	5.9775E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
34	13	14	6.0115E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09

<i>Combo - <math>q_i(d(T))</math> (continued)</i>							
35	14	15	6.0456E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
36	15	16	6.0798E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
37	16	17	6.1141E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
38	17	18	6.1484E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
39	18	19	6.1829E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
40	19	20	6.2175E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
41	20	21	6.2522E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
42	11	12	5.9436E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
43	12	13	5.9775E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
44	13	14	6.0115E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
45	14	15	6.0456E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
46	15	16	6.0798E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
47	16	17	6.1141E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
48	17	18	6.1484E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
49	18	19	6.1829E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
50	19	20	6.2175E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
51	20	21	6.2522E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
52	21	22	6.2869E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
53	22	23	6.3218E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
54	23	24	6.3567E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
55	24	25	6.3918E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
56	13	14	6.0115E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
57	14	15	6.0456E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
58	15	16	6.0798E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
59	16	17	6.1141E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
60	17	18	6.1484E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
61	18	19	6.1829E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
62	19	20	6.2175E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
63	20	21	6.2522E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
64	21	22	6.2869E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
65	22	23	6.3218E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
66	23	24	6.3567E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
67	24	25	6.3918E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
68	25	26	6.4269E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
69	26	27	6.4622E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
70	27	28	6.4975E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09

<i>Combo - <math>q_i(d(T))</math> (continued)</i>							
71	28	29	6.5330E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
72	15	16	6.0798E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
73	16	17	6.1141E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
74	17	18	6.1484E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
75	18	19	6.1829E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
76	19	20	6.2175E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
77	20	21	6.2522E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
78	21	22	6.2869E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
79	22	23	6.3218E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
80	23	24	6.3567E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
81	24	25	6.3918E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
82	25	26	6.4269E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
83	26	27	6.4622E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
84	27	28	6.4975E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
85	28	29	6.5330E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
86	29	30	6.5685E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
87	30	31	6.6041E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
88	31	32	6.6399E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
89	32	33	6.6757E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
90	17	18	6.1484E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
91	18	19	6.1829E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
92	19	20	6.2175E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
93	20	21	6.2522E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
94	21	22	6.2869E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
95	22	23	6.3218E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
96	23	24	6.3567E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
97	24	25	6.3918E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
98	25	26	6.4269E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
99	26	27	6.4622E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
100	27	28	6.4975E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
101	28	29	6.5330E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
102	29	30	6.5685E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
103	30	31	6.6041E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
104	31	32	6.6399E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
105	32	33	6.6757E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
106	33	34	6.7116E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09

<i>Combo - <math>q_i(d(T))</math> (continued)</i>							
107	34	35	6.7476E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
108	35	36	6.7838E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
109	36	37	6.8200E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
110	19	20	6.2175E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
111	20	21	6.2522E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
112	21	22	6.2869E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
113	22	23	6.3218E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
114	23	24	6.3567E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
115	24	25	6.3918E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
116	25	26	6.4269E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
117	26	27	6.4622E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
118	27	28	6.4975E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
119	28	29	6.5330E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
120	29	30	6.5685E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
121	30	31	6.6041E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
122	31	32	6.6399E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
123	32	33	6.6757E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
124	33	34	6.7116E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
125	34	35	6.7476E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
126	35	36	6.7838E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
127	36	37	6.8200E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
128	37	38	6.8563E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09
129	38	39	6.8927E-21	7.5396E-11	3.0158E-10	6.7855E-10	1.2063E-09

## Appendix I – Combinatorial Algebra Results, $Q(T)$

Tank	Case-0	Micro	Macro	Combined
0	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
1	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00
2	8.6417E-06	7.4679E-11	7.5395E-11	5.5774E-21
3	9.0073E-06	1.0783E-09	1.1772E-09	8.7542E-15
4	9.3997E-06	2.0805E-09	2.2901E-09	1.7494E-14
5	9.7662E-06	3.0830E-09	3.3982E-09	2.6249E-14
6	9.3997E-06	2.0805E-09	2.2901E-09	1.7494E-14
7	9.7661E-06	3.0830E-09	3.3981E-09	2.6249E-14
8	1.0132E-05	4.0849E-09	4.5070E-09	3.5008E-14
9	1.0499E-05	5.0874E-09	5.6176E-09	4.3782E-14
10	1.0865E-05	6.0900E-09	6.7294E-09	5.2566E-14
11	1.1232E-05	7.0925E-09	7.8426E-09	6.1359E-14
12	1.0132E-05	4.0849E-09	4.5070E-09	3.5008E-14
13	1.0499E-05	5.0874E-09	5.6175E-09	4.3782E-14
14	1.0865E-05	6.0894E-09	6.7288E-09	5.2561E-14
15	1.1232E-05	7.0919E-09	7.8419E-09	6.1354E-14
16	1.1598E-05	8.0944E-09	8.9563E-09	7.0157E-14
17	1.1964E-05	9.0969E-09	1.0072E-08	7.8970E-14
18	1.2331E-05	1.0099E-08	1.1189E-08	8.7792E-14
19	1.2697E-05	1.1102E-08	1.2307E-08	9.6625E-14
20	1.0865E-05	6.0894E-09	6.7288E-09	5.2561E-14
21	1.1232E-05	7.0918E-09	7.8419E-09	6.1353E-14
22	1.1598E-05	8.0939E-09	8.9557E-09	7.0152E-14
23	1.1964E-05	9.0964E-09	1.0071E-08	7.8965E-14
24	1.2331E-05	1.0099E-08	1.1188E-08	8.7787E-14
25	1.2697E-05	1.1101E-08	1.2306E-08	9.6620E-14
26	1.3063E-05	1.2104E-08	1.3426E-08	1.0546E-13
27	1.3430E-05	1.3106E-08	1.4546E-08	1.1431E-13
28	1.3796E-05	1.4109E-08	1.5668E-08	1.2317E-13
29	1.4163E-05	1.5112E-08	1.6792E-08	1.3205E-13
30	1.1598E-05	8.0939E-09	8.9557E-09	7.0152E-14
31	1.1964E-05	9.0963E-09	1.0071E-08	7.8964E-14
32	1.2330E-05	1.0098E-08	1.1188E-08	8.7783E-14
33	1.2697E-05	1.1101E-08	1.2306E-08	9.6615E-14

$Q(T)$ (continued)				
34	1.3063E-05	1.2103E-08	1.3425E-08	1.0546E-13
35	1.3430E-05	1.3106E-08	1.4546E-08	1.1431E-13
36	1.3796E-05	1.4108E-08	1.5668E-08	1.2317E-13
37	1.4162E-05	1.5111E-08	1.6791E-08	1.3204E-13
38	1.4529E-05	1.6113E-08	1.7915E-08	1.4092E-13
39	1.4895E-05	1.7116E-08	1.9041E-08	1.4981E-13
40	1.5262E-05	1.8118E-08	2.0168E-08	1.5871E-13
41	1.5628E-05	1.9121E-08	2.1296E-08	1.6762E-13
42	1.2330E-05	1.0098E-08	1.1188E-08	8.7783E-14
43	1.2697E-05	1.1101E-08	1.2306E-08	9.6614E-14
44	1.3063E-05	1.2103E-08	1.3424E-08	1.0545E-13
45	1.3429E-05	1.3105E-08	1.4545E-08	1.1430E-13
46	1.3796E-05	1.4108E-08	1.5667E-08	1.2316E-13
47	1.4162E-05	1.5110E-08	1.6790E-08	1.3204E-13
48	1.4529E-05	1.6113E-08	1.7915E-08	1.4092E-13
49	1.4895E-05	1.7115E-08	1.9040E-08	1.4981E-13
50	1.5262E-05	1.8118E-08	2.0167E-08	1.5871E-13
51	1.5628E-05	1.9120E-08	2.1295E-08	1.6762E-13
52	1.5994E-05	2.0123E-08	2.2425E-08	1.7654E-13
53	1.6361E-05	2.1125E-08	2.3555E-08	1.8547E-13
54	1.6727E-05	2.2128E-08	2.4687E-08	1.9441E-13
55	1.7094E-05	2.3131E-08	2.5821E-08	2.0336E-13
56	1.3063E-05	1.2103E-08	1.3424E-08	1.0545E-13
57	1.3429E-05	1.3105E-08	1.4545E-08	1.1430E-13
58	1.3796E-05	1.4107E-08	1.5666E-08	1.2316E-13
59	1.4162E-05	1.5110E-08	1.6790E-08	1.3203E-13
60	1.4529E-05	1.6112E-08	1.7914E-08	1.4091E-13
61	1.4895E-05	1.7115E-08	1.9040E-08	1.4980E-13
62	1.5261E-05	1.8117E-08	2.0166E-08	1.5870E-13
63	1.5628E-05	1.9120E-08	2.1295E-08	1.6761E-13
64	1.5994E-05	2.0122E-08	2.2424E-08	1.7653E-13
65	1.6361E-05	2.1125E-08	2.3555E-08	1.8546E-13
66	1.6727E-05	2.2127E-08	2.4687E-08	1.9440E-13
67	1.7093E-05	2.3130E-08	2.5820E-08	2.0335E-13
68	1.7460E-05	2.4132E-08	2.6954E-08	2.1231E-13
69	1.7826E-05	2.5135E-08	2.8090E-08	2.2128E-13

$Q(T)$ (continued)				
70	1.8193E-05	2.6137E-08	2.9227E-08	2.3025E-13
71	1.8559E-05	2.7140E-08	3.0365E-08	2.3924E-13
72	1.3796E-05	1.4107E-08	1.5666E-08	1.2316E-13
73	1.4162E-05	1.5110E-08	1.6789E-08	1.3203E-13
74	1.4528E-05	1.6112E-08	1.7913E-08	1.4091E-13
75	1.4895E-05	1.7114E-08	1.9039E-08	1.4980E-13
76	1.5261E-05	1.8117E-08	2.0166E-08	1.5870E-13
77	1.5628E-05	1.9119E-08	2.1294E-08	1.6761E-13
78	1.5994E-05	2.0122E-08	2.2423E-08	1.7653E-13
79	1.6360E-05	2.1124E-08	2.3554E-08	1.8546E-13
80	1.6727E-05	2.2127E-08	2.4686E-08	1.9440E-13
81	1.7093E-05	2.3129E-08	2.5819E-08	2.0334E-13
82	1.7460E-05	2.4132E-08	2.6954E-08	2.1230E-13
83	1.7826E-05	2.5134E-08	2.8089E-08	2.2127E-13
84	1.8192E-05	2.6137E-08	2.9226E-08	2.3025E-13
85	1.8559E-05	2.7139E-08	3.0364E-08	2.3924E-13
86	1.8925E-05	2.8142E-08	3.1504E-08	2.4823E-13
87	1.9292E-05	2.9144E-08	3.2644E-08	2.5724E-13
88	1.9658E-05	3.0147E-08	3.3786E-08	2.6626E-13
89	2.0025E-05	3.1150E-08	3.4930E-08	2.7529E-13
90	1.4528E-05	1.6112E-08	1.7913E-08	1.4091E-13
91	1.4895E-05	1.7114E-08	1.9039E-08	1.4980E-13
92	1.5261E-05	1.8116E-08	2.0165E-08	1.5869E-13
93	1.5627E-05	1.9119E-08	2.1293E-08	1.6760E-13
94	1.5994E-05	2.0121E-08	2.2423E-08	1.7652E-13
95	1.6360E-05	2.1124E-08	2.3553E-08	1.8545E-13
96	1.6727E-05	2.2126E-08	2.4685E-08	1.9439E-13
97	1.7093E-05	2.3129E-08	2.5819E-08	2.0334E-13
98	1.7459E-05	2.4131E-08	2.6953E-08	2.1230E-13
99	1.7826E-05	2.5134E-08	2.8089E-08	2.2127E-13
100	1.8192E-05	2.6136E-08	2.9226E-08	2.3024E-13
101	1.8559E-05	2.7139E-08	3.0364E-08	2.3923E-13
102	1.8925E-05	2.8141E-08	3.1503E-08	2.4823E-13
103	1.9291E-05	2.9144E-08	3.2644E-08	2.5724E-13
104	1.9658E-05	3.0146E-08	3.3786E-08	2.6625E-13
105	2.0024E-05	3.1149E-08	3.4929E-08	2.7528E-13



$Q(T)$ (continued)				
106	2.0391E-05	3.2151E-08	3.6073E-08	2.8432E-13
107	2.0757E-05	3.3154E-08	3.7219E-08	2.9336E-13
108	2.1123E-05	3.4156E-08	3.8366E-08	3.0242E-13
109	2.1490E-05	3.5159E-08	3.9515E-08	3.1149E-13
110	1.5261E-05	1.8116E-08	2.0165E-08	1.5869E-13
111	1.5627E-05	1.9119E-08	2.1293E-08	1.6760E-13
112	1.5994E-05	2.0121E-08	2.2423E-08	1.7652E-13
113	1.6360E-05	2.1124E-08	2.3553E-08	1.8545E-13
114	1.6727E-05	2.2126E-08	2.4685E-08	1.9439E-13
115	1.7093E-05	2.3129E-08	2.5819E-08	2.0334E-13
116	1.7459E-05	2.4131E-08	2.6953E-08	2.1230E-13
117	1.7826E-05	2.5134E-08	2.8089E-08	2.2127E-13
118	1.8192E-05	2.6136E-08	2.9226E-08	2.3024E-13
119	1.8559E-05	2.7139E-08	3.0364E-08	2.3923E-13
120	1.8925E-05	2.8141E-08	3.1503E-08	2.4823E-13
121	1.9291E-05	2.9144E-08	3.2644E-08	2.5724E-13
122	1.9658E-05	3.0146E-08	3.3786E-08	2.6625E-13
123	2.0024E-05	3.1149E-08	3.4929E-08	2.7528E-13
124	2.0391E-05	3.2151E-08	3.6073E-08	2.8432E-13
125	2.0757E-05	3.3154E-08	3.7219E-08	2.9336E-13
126	2.1123E-05	3.4156E-08	3.8366E-08	3.0242E-13
127	2.1490E-05	3.5159E-08	3.9514E-08	3.1148E-13
128	2.1856E-05	3.6161E-08	4.0664E-08	3.2056E-13
129	2.2223E-05	3.7164E-08	4.1815E-08	3.2965E-13

## Appendix J – Tank Reliability Results, $R(T)$

Tank	Case-0	Micro	Macro	Combined
0	1.0000E+00	1.0000E+00	1.0000E+00	1.0000E+00
1	1.0000E+00	1.0000E+00	1.0000E+00	1.0000E+00
2	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
3	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
4	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
5	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
6	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
7	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
8	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
9	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
10	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
11	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
12	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
13	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
14	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
15	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
16	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
17	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
18	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
19	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
20	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
21	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
22	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
23	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
24	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
25	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
26	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
27	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
28	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
29	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
30	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
31	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
32	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
33	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00

<i>R(T)</i> (continued)				
34	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
35	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
36	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
37	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
38	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
39	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
40	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
41	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
42	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
43	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
44	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
45	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
46	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
47	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
48	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
49	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
50	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
51	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
52	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
53	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
54	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
55	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
56	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
57	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
58	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
59	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
60	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
61	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
62	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
63	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
64	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
65	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
66	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
67	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
68	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
69	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00

<i>R(T)</i> (continued)				
70	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
71	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
72	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
73	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
74	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
75	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
76	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
77	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
78	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
79	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
80	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
81	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
82	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
83	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
84	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
85	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
86	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
87	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
88	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
89	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
90	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
91	9.9999E-01	1.0000E+00	1.0000E+00	1.0000E+00
92	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
93	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
94	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
95	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
96	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
97	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
98	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
99	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
100	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
101	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
102	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
103	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
104	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
105	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00

<i>R(T)</i> (continued)				
106	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
107	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
108	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
109	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
110	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
111	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
112	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
113	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
114	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
115	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
116	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
117	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
118	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
119	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
120	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
121	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
122	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
123	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
124	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
125	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
126	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
127	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
128	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00
129	9.9998E-01	1.0000E+00	1.0000E+00	1.0000E+00

## Appendix K – Final Results

### K.a Sector Results

Sector	Case-0	Micro	Macro	Combined
$R_{sect} = \prod(R_T)$	9.9803E-01	1.0000E+0	1.0000E+00	1.0000E+00
$Q_{sect} = 1 - R_{sect}$	1.9749E-03	2.3832E-06	2.6597E-06	2.0932E-11
Mean Time between lost msgs (sec) = $Q_{sect}^{-1}$	5.0635E+02	4.1960E+0	3.7598E+05	4.7774E+10
Msgs lost in a year = $(31,556,926/Q_{sect}^{-1})$	6.2323E+04	7.5207E+0	8.3932E+01	6.6055E-04

### K.b System Results

System = 33 Sectors	Case-0	Micro	Macro	Combined
$R_{sys} = \prod(R_{sect})$	9.3685E-01	9.9992E-01	9.9991E-01	1.0000E+00
$Q_{sys} = 1 - R_{sys}$	6.3155E-02	7.8643E-05	8.7767E-05	6.9076E-10
Mean Time between lost msgs (sec) = $Q_{sys}^{-1}$	1.5834E+01	1.2716E+0	1.1394E+04	1.4477E+09
Msgs lost in a year = $(31,556,926/Q_{sys}^{-1})$	1.9930E+06	2.4817E+0	2.7369E+03	2.1798E-02