FAILURE MECHANISMS OF COMPLEX SYSTEMS

A Dissertation
Presented to
The Academic Faculty

by

Shahnewaz Siddique

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Aerospace Engineering

Georgia Institute of Technology
May 2014

FAILURE MECHANISMS OF COMPLEX SYSTEMS

Approved by:

Dr. Eric M. Feron, Committee Chair
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Vitali V. Volovoi, Advisor
Independent Consultant

Dr. Dewey H. Hodges
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Daniel P. Schrage
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Jeff S. Shamma
School of Electrical and Computer Engineering
Georgia Institute of Technology

Date Approved: 26 February 2014

For my father, mother, Nabila, Musa and Maryam

ACKNOWLEDGEMENTS

The School of Aerospace Engineering provided me an excellent environment to develop this thesis. I am extremely grateful for the opportunity. First, I would like to thank my advisor Prof. Vitali Volovoi for guiding me through this research work. I would like to thank Prof. Eric Feron, Prof. Dewey Hodges, Prof. Daniel Schrage and Prof. Jeff Shamma, for agreeing to be part of the thesis committee. I would like to thank Prof. Jechiel Jagoda for providing the funding that allowed this work to be completed. I would also like to thank my colleagues and peers at Georgia Tech: Anirudh Bhat, An Vu, Dr. Giorgio Calanni and others who I interacted with on a regular basis.

We spent a few formative years at Stanford University. Undoubtedly the academic environment and student life at Stanford has greatly shaped my thought process and approach to life. At Stanford, at various times, I came into contact with faculty and students who contributed significantly to my development. Here I would like to specifically thank: Prof. Andrew Fire, Prof. Oliver Fringer, Dr. Chinmay Patel, Prof. Nick Mckeown and Prof. Fu-Kuo Chang. My interaction with them enriched me greatly.

During my time at Ample Communications in Fremont, California, I had the opportunity to work with some exceptional people. Here I would like to thank my manager Kumar Bhattaram and my co-workers Jimmy Chan, Chakravarthy Kosaraju and Edward Yang. I had a excellent experience interacting and learning from them. I found them all to be kind and compassionate people.

Last but not least, I must thank my family for the difficulty they endured as I undertook this journey. First, I would like to thank my mother and father for always providing unconditional love and support. I would like to thank my wife, Nabila, for

always remaining patient. In the coming years I would like to make up for, as much as possible, the difficulties I caused.

Georgia Institute of Technology and Atlanta has been an excellent experience for me and family. We will cherish forever our very kind and generous friends and the uncountable beautiful memories.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

SUMMARY

Understanding the behavior of complex, large-scale, interconnected systems in a rigorous and structured manner is one of the most pressing scientific and technological challenges of current times. These systems include, among many others, transportation and communications systems, smart grids and power grids, financial markets etc. Failures of these systems can have potentially enormous social, environmental and financial costs.

In this work, we investigate the failure mechanisms of load-sharing complex systems. The systems are composed of multiple nodes or components whose failures are determined based on the interaction of their respective strengths and loads (or capacity and demand respectively) as well as the ability of a component to share its load with its neighbors when needed. Each component possesses a specific strength (capacity) and can be in one of three states: failed, degraded or functioning normally. The states are determined based on the load (demand) on the component.

We focus on two distinct mechanisms to model the interaction between components strengths and loads. The first, a Loss of Strength (LOS) model and the second, a Customer Service (CS) model. We implement both models on lattice and scale-free graph network topologies. The failure mechanisms of these two models demonstrate temporal scaling phenomena, phase transitions and multiple distinct failure modes excited by extremal dynamics. We find that the resiliency of these models is sensitive to the underlying network topology. For critical ranges of parameters the models demonstrate power law and exponential failure patterns.

We find that the failure mechanisms of these models have parallels to failure

mechanisms of critical infrastructure systems such as congestion in transportation networks, cascading failure in electrical power grids, creep-rupture in composite structures, and draw-downs in financial markets. Based on the different variants of failure, strategies for mitigating and postponing failure in these critical infrastructure systems can be formulated.

CHAPTER I

INTRODUCTION

1.1 Motivation

Understanding the behavior of complex, large-scale, interconnected systems in a rigorous and structured manner is one of the most pressing scientific and technological challenges of current times. While individual components of a system may be well understood and characterized, there is relatively little understanding of neither the long run time-behavior of large-scale interconnected systems nor the nature of cascading failures in complex systems. In modern times, these are of immense concern due to the enormous economic, infrastructure and social costs of complex system failures.

In this work, an attempt is made to understand the failure mechanisms of complex interconnected systems. Of particular interest is characterizing the extremal dynamics, mechanisms of cascading failure and the degree of damage and time-to-failure distributions of complex systems. Also of interest is identifying techniques for mitigating, postponing, and reversing the onset of failure.

A significant body of research has accumulated in recent years in the study of complex networks. These are networks whose structure is large-scale and dynamically evolving in time [16]. Unlike the traditional research cited in [16], the focus here is from the perspective of system reliability [84, 11, 65, 72]. The particular emphasis is on the study of time-to-failure and degree of damage distributions, rather than steady-state patterns and mean time-to-failure characteristics. A system reliability approach enables mapping of the time-to-failure distributions to the traditional failure-time parametric distributions studied in statistical reliability theory [58, 14, 15]. The utility in this approach is the predictive capability acquired in predicting the various

failure modes and pertinent distributions of complex systems. Assuming that the underlying physical mechanisms of failure are correctly identified, selecting a parametric distribution effectively provides a means of representing the underlying failure dynamics of the system and thus significantly reduces the need for experimental data.

1.2 Literature review

The study of the structure and dynamics of complex systems [73, 37, 16] has attracted a great deal of interest in recent years. These are systems that are composed of a large number of interconnected components. These systems include but are not limited to: social networks, chemical and biological systems such as protein, neural and genetic networks, engineered systems such as the power grid, Internet, transportation and communication networks etc. The structural analysis of complex systems is mainly concerned with characterizing the topology of the underlying network, i.e. the wiring of the components that compose the system. On one hand, characterizing the topology of the network allows researchers to identify the structural properties of the network and develop realistic node connectivity models for real networks. On the other hand, studying the dynamics of complex systems allows one to understand the collective behavior of the system as it evolves in time. Study of the dynamics of complex systems relies on capturing the dynamics of individual components and the dynamical interaction of these components with each other.

The robustness of complex systems to failure in both static and dynamic contexts has received significant attention in the literature. Failures in terms of static node or component removals have been studied in [1, 32, 33]. Dynamic failure scenarios have also been studied in a variety of contexts: capacity constraints and propagating failures due to shared load [68, 69], cascading failures in power grids [55, 31], jamming transition in air transportation networks [56], and congestion in communication networks [36, 40].

From a more theoretical or conceptual perspective of the dynamical evolution of complex systems, *Self-organized criticality* (SOC) has been put forth as one explanation for the complexity in nature [5]. SOC describes the dynamics of complex systems which have a "critical-point" as an attractor [7, 6, 75]. SOC is typically observed in non-equilibrium dynamical systems with spatial degrees of freedom which naturally evolve into a self-organized critical-point. These systems possess a key state parameter, such as density or temperature, which characterizes the underlying generic and random system. A specific value of density or temperature is associated with the system at the critical-point. The characteristic signatures of systems exhibiting SOC are: self-similarity, phase transitions, power laws, punctuated equilibria, universality classes and other signatures. Power laws have been discovered in information/communication networks [78, 76, 77, 99, 18], biological systems [51, 50], and social networks [35, 46, 3, 39, 102]. It has been observed that many real networks display power law shaped degree distributions, $P(k) \sim Ak^{-\gamma}$ [9, 10]. The widespread observation of power laws has been interpreted as evidence for SOC and edge of chaos (EOC) [59] phenomena.

However, other authors have provided alternate explanations to how complexity arises in nature that are at variance with SOC. For example, Carlson and Doyle [20, 21] have proposed *Highly Optimized Tolerance* (HOT) as an alternative to SOC. HOT systems are optimized, through natural selection or engineering design, to provide robust performance at designed-for-uncertainties. In their work, Carlson and Doyle illustrate several essential characteristics of complexity in nature that are absent in the SOC framework but present in HOT [20]. For example, like SOC, HOT also produces power-laws. However, in SOC the power-laws are associated with the critical density, where as in HOT, power-laws are present for values higher than the critical density. The mechanisms for producing power-law distributions in HOT are different from the mechanisms for criticality in SOC.

Carlson and Doyle [20, 21] cite the Internet as one example in which power laws are ubiquitous. It is tempting to view the Internet as a collection of independent agents who adaptively self-organize into a complex state. However, the internal details of the Internet are highly structured with sophisticated control protocols that are extremely robust to the main sources of uncertainty. The different layers of the Internet, which sit one on top of the other, create the illusion of apparent simplicity, which is exactly the opposite of SOC and EOC. Furthermore, the power-laws of the Internet are **independent** of density (which in this case is the congestion level) which can vary enormously over the network. This suggests that criticality may not be relevant.

Unlike SOC/EOC, where external forces only initiate events and the mechanism which gives rise to complexity is self-contained, HOT takes into account the fact that engineering designs are developed and biological systems evolve in a manner which rewards successful strategies subject to a specific form of external stimulus. Using HOT states, which are essentially added design configurations to percolation and sand-pile models, the authors retain maximum yield well beyond the critical point.

To capture the structural properties of complex systems, a natural approach is to model system connectivity using graph theory [17, 104]. In this case, the nodes of the graph represent the components of the system and edges represent the wiring of the system and the interaction of the components with each other. While graph theory constitutes a natural framework to characterize the structural topology of complex networks, a simpler and nearly canonical form of a complex system is represented by Cellular Automata (CA) [106]. Cellular automata consists only of nodes with links implied implicitly. In cellular automata of complex systems, each node of the network represents an agent that can be in any one of a finite number of states. At each time step, the state of the agent is computed based on its previous state and the state of its neighbors. Cellular Automata is a relatively new simulation approach that simulates physical processes that are discrete both in time and in space [24]. Broadly, CA can

be categorized into two types: deterministic, using deterministic evolution rules and probabilistic, using probabilistic evolution rules. Cellular Automata has been used to simulate statistical physical phenomena such as forest fires, urban traffic jams and flows, growth of dendrite [60], durability of concrete in aggressive environments [13], multi-pit corrosion [82], wind damage in forest planning [107], rock failures [2], creep rupture [25], and meso-scale damage evolution in structures [64, 108]. Cellular automata has also been used in connection to *self-organized criticality* behavior to model landslides, forest fires, and earthquakes [97]. In general, cellular automata is mostly concerned with steady-state patterns and averages properties such as mean time-to-failure and expected transient time.

A model employing cellular automata techniques, the *Universal Failure Model* (UFM), is proposed in [101] to model the failure dynamics of complex systems. The UFM consists of a large number of components performing a common function. Each component possesses a specific strength or load capacity and can be in one of three states: failed, damaged or functioning normally. The states are determined based on the demand load on the component. If the demand exceeds the capacity or strength of the component than the component fails and the components load is redistributed among its neighbors. Utilizing the UFM, specific shapes of distributions for time-to-failure and cumulative damage (thus making possible the study of cascading failure mechanisms in complex systems) can be studied, thus enabling a mapping to the traditional failure-time parametric distributions such as: Log-normal, Weibull, Exponential, Gamma, Birnbaum-Saunders distributions etc.

1.3 Objectives

The objective of this work is to use the UFM to model failure dynamics of complex systems. In particular, the object is to identify distinct patterns of failure propagation as functions of the input parameters.

First, we will investigate the UFM in a lattice configuration. Later we will relax our assumption on the lattice network topology and investigate the failure mechanism of the UFM in a scale-free graph network topology. We will explore different variants of the UFM; and for reasons to be described in detail in the next chapter, we will refer to the different flavors of the UFM as the LOS (Loss of Strength) or CS (Customer Service) model.

Second, we will investigate congestion phenomena in both lattice and scale-free network setting from the perspective of communication and transportation systems. We will investigate congestion phenomena in the context of a Customer Service (CS) model. We will describe in detail the CS model in the next chapter.

With these objective in mind the outline of this thesis is described next.

1.4 Outline

The thesis is organized in the following chapters:

- In chapter II, we introduce the LOS and CS models. We study the failure mechanisms of both the LOS and CS models on lattice and scale-free graph network topologies.
- In chapter III, we provide a mathematical treatment of the LOS and CS models.
- In chapter IV, we provide additional discussion of the properties of the LOS and CS models that were not covered in chapters II and III.
- In chapter V, we extend the LOS and CS models to study practical applications. We use the LOS model in creep-rupture studies and compare it to the model proposed by Mahesh and Phoenix [62]. We use the CS model in air traffic congestion management studies and relate it to the aggregate flow models of Sridhar et. al. [93]. We conclude by employing the CS model to interpret

metropolitan road traffic congestion using metro Atlanta traffic congestion as an example.

- In chapter VI, we investigate complex systems from a multi-agent system perspective. To model the dynamics of individual agents in the system we use the Dubins vehicle model [38]. This provides a realistic framework to study congestion phenomena for air transportation systems.

- In chapter VII, we conclude with a summary of our results, contributions and possible directions for future work.

CHAPTER II

FAILURE MECHANISMS OF LOAD-SHARING COMPLEX SYSTEMS

In this chapter we discuss the failure mechanisms of load-sharing complex systems. Most of the material in this chapter has been published in Physical Review E in [88]. Here we reproduce that material.

2.1 Introduction

In the last decade a significant body of research has accumulated in the study of complex systems, their structure and dynamics [73, 16]. Static robustness in terms of node removals have been explored in [1, 32, 33]. But most real networks undergo dynamic failures where the failure of a single or multiple nodes might trigger cascades of failure through the network. Dynamical redistribution of flow have been considered in different real world networks: power grids [55, 31], air transportation networks [56], and communication networks [36].

Many physical systems fail as their capacity or strength degrades over time under constant load or load increases over time as strength remains fixed. For example, loss of strength phenomena is observed in stress-rupture or creep rupture [62], tire wear, and level of fluid in a hydraulic system [58]; whereas load buildup is commonly considered in fiber-bundle models on complex systems [66, 54]. Failure occurs when component load is greater than its strength. Component failure due to overloading is a serious threat in networks: a single component failure and its subsequent load redistribution can trigger cascades of failures through the network, ultimately bringing down the entire system [66, 55].

On the other hand, many communication and transportation systems exhibit congestion phenomena as data or customer traffic density increases beyond certain thresholds. Congestion or jamming phenomena for critical values of traffic density has been demonstrated in models of transportation [71, 87, 70] and communication networks [74, 89, 98]. Traffic flow models for air transportation systems have been explored in [56, 86, 93]. Congestion in one part of the network has the effect of rerouting traffic to other parts of the network resulting in slowing down or clogging traffic in the entire system.

In this work we explore two different models of interaction between component strength and load to understand the failure mechanisms of complex systems. The first one is a Loss of Strength (LOS) model where components lose strength over time following prescribed rules. The second one is a Customer Service (CS) model where component demand is modeled through customer or data traffic arrival rate. For both models we investigate the strength-load interaction both at and below critical loading levels [66, 67]. At critical loading levels and above the entire system abruptly collapses, which we refer to as the critical state.

First, we describe the general system setup. We implement both models on two different network topologies: individual components organized in a $n \times n$ lattice or a scale-free (SF) network of n^2 components following a power law degree distribution $P(k) \sim Ak^{-\gamma}$, with exponents $2 < \gamma < 3$. The SF network is constructed using the Barabási-Albert (BA) algorithm [9, 10]. The BA SF model is a growth and preferential attachment algorithm where at each iteration step a new node is attached to "m" existing nodes in the network, where m is a constant input parameter. At the end of the iteration steps, a scale-free network of average degree $\langle k \rangle = 2m$ is obtained. We generate BA SF networks for $m \in (2, 4, 6)$ which results in average degree $\langle k \rangle \in (4, 8, 12)$. These choices of $\langle k \rangle$ cover a range of communication, biological and social networks in the real world [16]. In both network topologies each component can be

in one of three possible modes: fully operational, stressed, and failed. We denote by (i, j) the location of a component in the lattice. For the scale-free configuration we number the components from 1 to n^2. Next we describe the LOS model in the lattice configuration.

2.2 The LOS model

For the LOS model on a lattice topology each component is initialized with a specific strength S_{ij}. Component loads L_{ij} are initialized with the same value and during the simulation are not exogenously varied. If $L_{ij} \leq \eta S_{ij}$, where $\eta \in (0, 1)$ is a parameter to control strength degradation threshold, then the component is fully operational and strength does not degrade. If $\eta S_{ij} < L_{ij} \leq S_{ij}$ then the component at (i, j) is considered stressed and loss of strength takes place over time. We consider deterministic loss of strength for components [58]. The components strength degradation follows the relationship $S_{ij}^{t} = -\alpha t + S_{ij}^{t'}$ where α is the strength degradation rate parameter, t is the time and $S_{ij}^{t'}$ denotes the strength at time $t = t'$ when component LOS commences. If $L_{ij} > S_{ij}$ then the component fails and the load is redistributed equally to the components immediate neighbors in the system. Once a component fails it is removed from the network.

The objective of the S_{ij} and L_{ij} initializations is to capture the interaction dynamics of component strength and load. The simulations work in the following way, first all component loads are set to a specific value $L_{ij} = L$ where $L \in [0.5..4]$. During a simulation L is not exogenously varied. For each load setting L, 30,000 Monte-Carlo simulations are carried out and S_{ij} is reinitialized for each simulation. To generate a mix of strong and weak components, S_{ij} is initialized from a real uniform distribution $\mathcal{U}[6, 10]$. To initiate LOS dynamics, for each simulation 4-5 components are initialized to a stressed mode $L_{ij} > \eta S_{ij}$ where deterministic loss of strength takes place. As L is steadily increased, we arrive at critical ranges of L where interactions

between components L_{ij} and S_{ij} trigger LOS dynamics and load redistributions for increasingly greater number of components. This allows us to capture the failure mechanisms of the system. In the simulations $t = k\Delta t$ with $\Delta t = 0.1$, $k = 1..500$, $n = 12$, $\eta = 0.7$ and $\alpha = 0.2$. The components at the boundary are initialized to very high strength to prevent failure. Since boundary components do not fail we do not need to deal with their load redistributions.

In the SF network case each component has neighbors following a power-law degree distribution. The simulation initializations for LOS SF network model is the same as the lattice configuration except with $n^2 = 100$ and for each simulation we generate a BA SF network. Also by construction all components for the LOS model on a SF network have neighbors so special handling of boundary components is not necessary.

2.3 The CS model

Next we describe the CS model on a lattice configuration. For the CS model we have taken a "Eulerian" [57] point of view for component flow dynamics as opposed to the standard "Lagrangian" point of view of our references. Component demand is modeled as a customer or data arrival rate λ. Although traffic in real communication networks is non-Poissonian [89, 79], as a first step we follow [86, 74] and model customer demand as a Poisson process with rate λ. The rate λ is the same for all components and does not vary during a simulation. Thus the system is in effect subjected to a uniformly distributed globally varying load. Component capacity is modeled through a fixed customer departure rate γ_{ij}. In addition, each component possesses an associated queue q_{ij} for extra storage capacity. At a given time step if $\lambda \leq \gamma_{ij}$ then the component is fully operational. If $\lambda > \gamma_{ij}$ then the excess demand $(\lambda - \gamma_{ij})$ is redistributed to the fully operational neighbors of the component. Excess demand is transferred sequentially to the neighbors with the largest spare capacity $(\gamma_{ij} - \lambda) > 0$ where (i, j) denotes the location of the neighbors. If component demand redistribution

is successful then the component remains fully operational. The component excess demand redistribution might be partially or completely unsuccessful. In that event, the remaining excess demand is placed in the queue q_{ij} for processing in the next time step. If the remaining excess demand is placed in q_{ij} successfully then the component at (i, j) is considered stressed. If the remaining excess demand overwhelms q_{ij} then the component is considered congested (failed) as it is not able to service the traffic demand. Once a component is congested it is taken off the grid.

Similar to the LOS model, we capture the interaction between component capacity and demand for the CS model through critical ranges of λ that trigger demand redistribution and congestion. In our simulations, component capacities γ_{ij} are initialized by sampling from a integer uniform distribution $\mathcal{U}[8, 12]$ to generate a mix of strong and weak components. We run 30,000 Monte-Carlo simulations for each integer value of $\lambda \in [5, 11]$. For each simulation the system is initialized with new capacities γ_{ij}. Each simulation is run for $t = k\Delta t = 500$ time steps where $\Delta t = 1$. The queue size is set to $q = 6$ and $n = 12$. For our ranges of λ the queue essentially provides components additional time to prevent failure. Boundary components have queue's set to large values to prevent component failure and avoid load redistribution. For the CS model on a SF network we have $n^2 = 100$ and due to circular boundary conditions special handling of boundary components is not necessary.

Before presenting our results we note a important difference between the CS and LOS models. In the LOS model, a component's load redistribution is the final step before it fails: once LOS dynamics is initiated the component will fail and an attempt will be made to redistribute its load. In the LOS model a component can undergo, at most, one load redistribution. In the CS model a component is essentially renewed through successful excess demand redistribution. The component fails only if the load redistribution is unsuccessful and the associated component queue q_{ij} is overwhelmed. In the CS model a component can complete multiple excess demand redistributions

and remain fully operational.

Another important point is regarding failed component load that is not successfully redistributed. In both models we shed the load and consider it lost. This is common in the context of internet traffic where packets are routinely discarded when routers are congested [49]. Similarly, power grid substations have mechanisms which take them offline during capacity/demand imbalances [1].

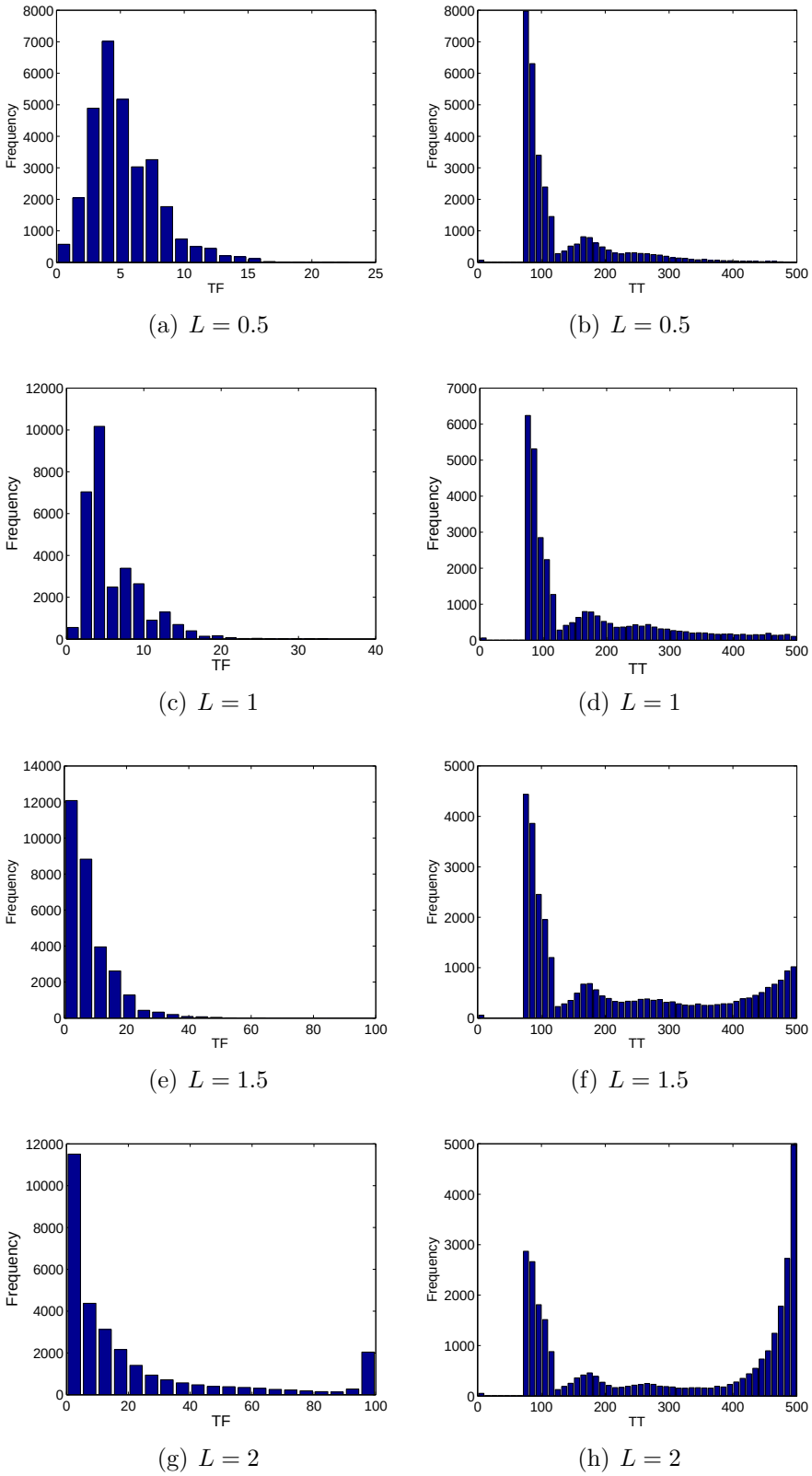
2.4 Results

Next, we define two quantities measured in the simulations. During each simulation components fail as the system evolves in time. We denote by 'TF' (Terminal Failure) the number of component failures at the end of a simulation. TF is a measure of the degree of system failure. We denote by 'TT' (Terminal Time) the time step when the penultimate component failure occurred. TT can be interpreted as the time the system achieves a pseudo steady-state.

The TF and TT distributions for different values of load initialization for the LOS model on a lattice configuration are shown in Figs. 1 and 2. At loading $L = 0.5$ the system is far from critical. At these loading levels component failures are mainly due to components that were initialized to commence LOS and their subsequent load redistributions to weak components. At these loading levels the systems are resilient to chains of cascading failure triggered by load redistribution, this fact is indicated by Fig. 1(a).

As the initialization load is increased, transition load conditions can be identified for $L = 2, 2.5$. The bi-modal distributions in Fig. 1(g), 2(a) resemble bathtub like curves that are commonly observed in reliability of complex systems [65]. In Fig. 2(a) half the simulations represent systems with all components failing. The other half

[1] M.J. Wald, R. Perez-Pena, N. Banerjee, *The New York Times*, August 16, 2003



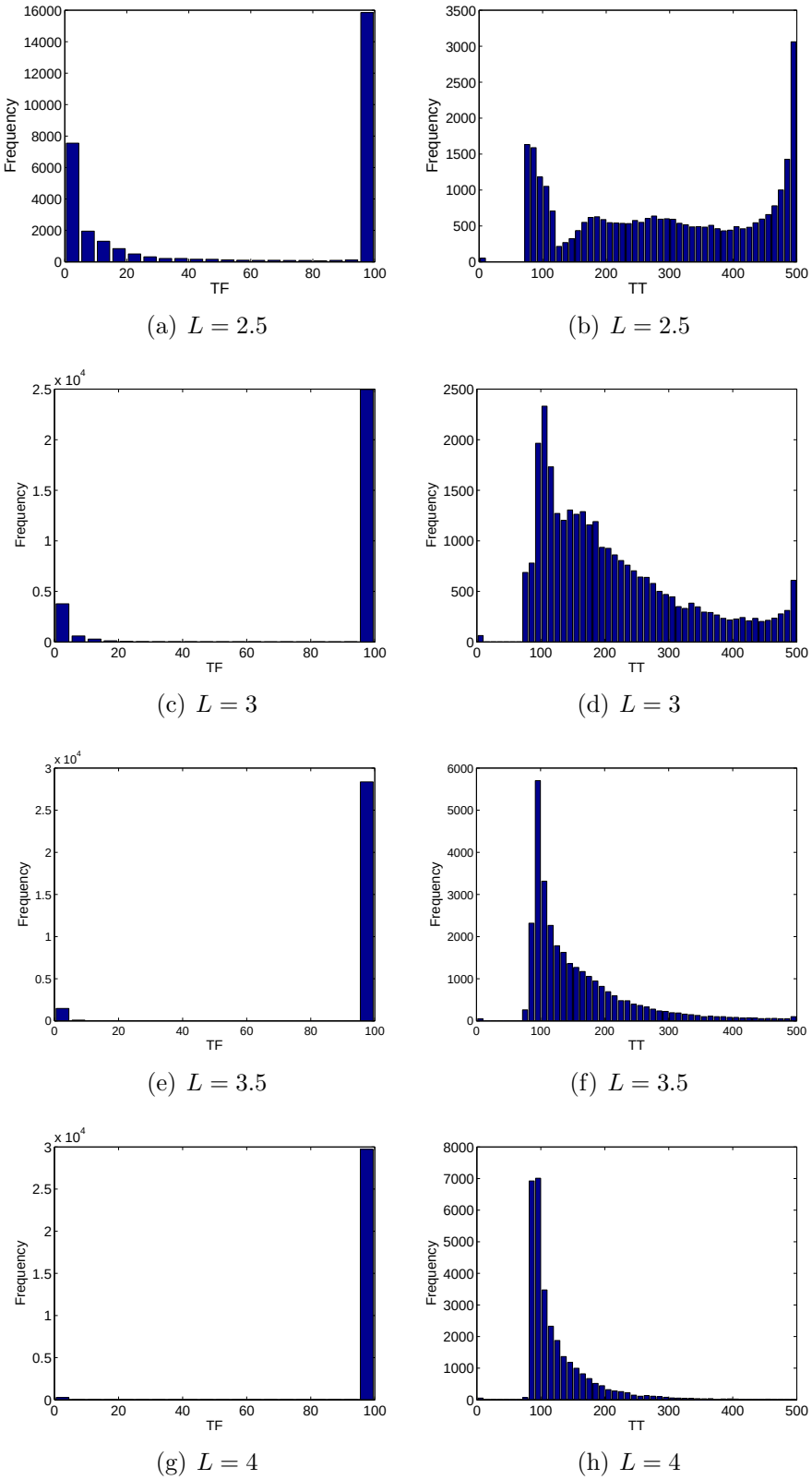
(a) $L = 0.5$ (b) $L = 0.5$

(c) $L = 1$ (d) $L = 1$

(e) $L = 1.5$ (f) $L = 1.5$

(g) $L = 2$ (h) $L = 2$

Figure 1: LOS model distributions. TF distributions (left column); TT distributions (right column).



(a) $L = 2.5$ (b) $L = 2.5$

(c) $L = 3$ (d) $L = 3$

(e) $L = 3.5$ (f) $L = 3.5$

(g) $L = 4$ (h) $L = 4$

Figure 2: LOS model distributions. TF distributions (left column); TT distributions (right column).

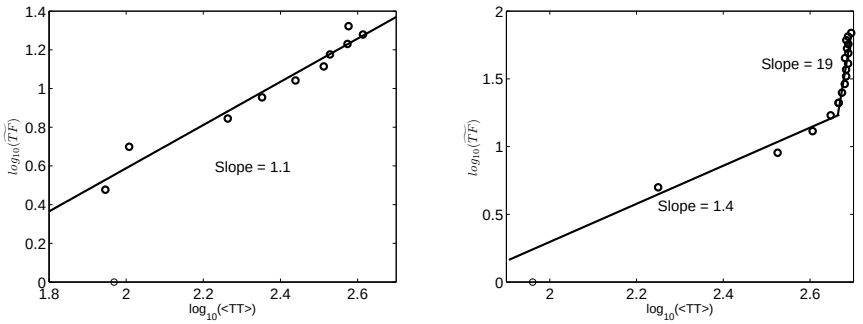
represent systems constituting partial component failures. This implies for approximately 50% of the simulations the system strength topology is such that cascading chains of load redistributions are triggered which eventually bring down the entire system. For the other 50% of simulations the system strength topology is strong enough to withstand the load redistributions thus avoiding a cascading chain of failures. Transition load settings are similar to 'tipping points' or 'critical thresholds' [94, 63]. In our simulation framework, at tipping-points systems may or may not, depending on system strength topology, descend into catastrophic failure.

2.4.1 Scaling phenomena

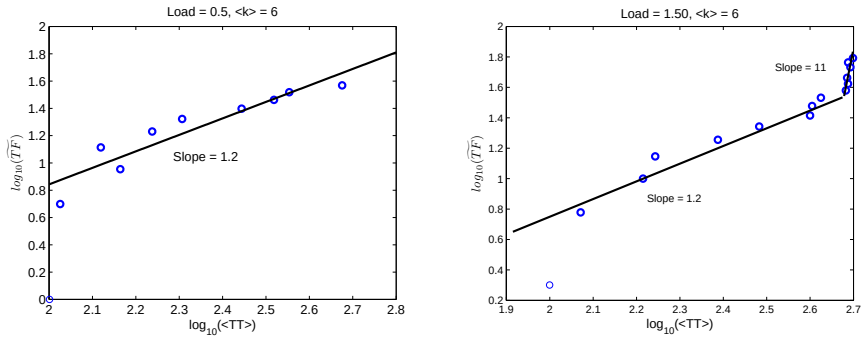
Recalling that TF represents the degree of system failure for a given simulation, let $\widetilde{TF}$ denote "smoothed" versions of TF. $\widetilde{TF}$ exhibits temporal scaling phenomena for the LOS model for load values lower than the transition load on both the lattice and SF networks of $\langle k \rangle = 12$. $\widetilde{TF}$ is constructed in the following way. Referring to the TF and TT distributions in Figs. 1 and 2, each point in the TF distribution has a associated point in the TT distribution. For a particular system loading L, we bin the TF distribution in groups of 4 in ascending order and denote them by $\widetilde{TF}_i$ with value set to the minimum TF value in the bin [2]. For each bin we find the corresponding values in the TT distribution and compute their mean, denoted $\langle TT \rangle_i$. We illustrate the temporal scaling phenomena of $\widetilde{TF}$ in Fig. 3 for different values of loading for the LOS model in both lattice and SF network configurations. In Fig. 3, $\widetilde{TF}$ versus $\langle TT \rangle$ is plotted in a log-log scale. Each circle in the figures represents the mean of a TT distribution conditioned on a $\widetilde{TF}_i$.

From these figures the following scaling relation is established for loading values far below the critical load,

[2]The results still hold if we set the value to the mean or max of the bin. The objective of binning is to smooth or denoise the data



(a) Lattice network $L = 0.5$ (b) Lattice network $L = 1.5$



(c) Scale-Free network $L = 0.5$, $\langle k \rangle = 12$ (d) Scale-Free network $L = 1.5$, $\langle k \rangle = 12$

Figure 3: $\widetilde{TF}$ temporal scaling for the LOS model, lattice configuration (top row) and SF network (bottom row). The plots are on a base 10 log-log scale.

$$\langle TT \rangle = \kappa \widetilde{TF}^{\tau} \tag{1}$$

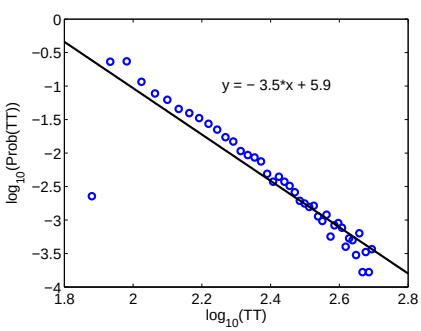
Table 1 tabulates the numerical values for κ and τ for different values of load. At low values of load, Figures 3(a) and 3(c), the logarithm of $\widetilde{TF}$ scales linearly versus the logarithm of $\langle TT \rangle$. As the initial load setting is increased, a breakpoint develops and the $\widetilde{TF}$'s separate into two different log-log linear scales, as illustrated in Figs. 3(b), 3(d). The slopes of the figures indicate the second group of $\widetilde{TF}$'s have faster transition dynamics to $\langle TT \rangle$ compared to the first $\widetilde{TF}$ group. Table 1 also tabulates the break point $\widetilde{TF}$ when the switch to faster transition dynamics occurs and the residual error of the data fit.

2.4.2 Critical behavior

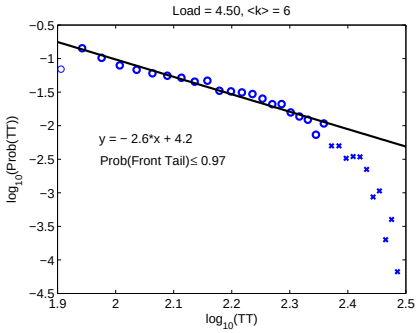
All systems in the LOS lattice configuration undergo complete failure at the critical load $L = 4$, as seen in Fig. 2(g) and $\widetilde{TF}$ is characterized by a first-order phase transition into the critical state [Fig. 5(a)]. We define the critical load as the load required for complete system failure, TF = 99 or 100, with probability greater than 0.95. For the LOS model on SF networks the critical load is slightly higher at $L = 4.5$ for $\langle k \rangle = 12$, $L = 5$ for $\langle k \rangle = 8$ and $L = 5.5$ for $\langle k \rangle = 4$. Here we recall that component strengths are initialized from a real uniform distribution $\mathcal{U}[6, 10]$ with $\eta = 0.7$, meaning that for $L > 6 \times 0.7 = 4.2$ a considerable number of components will be initialized in the stressed mode $L_{ij} > \eta S_{ij}$. To induce system failure in LOS SF networks, with decreasing $\langle k \rangle$ more and more components need to be initialized to commence failure dynamics. The implication is that the LOS model is increasingly resilient to system failure with decreasing average network connectivity $\langle k \rangle$. This result is in agreement with [1] which demonstrate that scale-free networks are more resilient to random errors or failure [3] compared to other network topologies.

The TT distribution fit for the LOS model at the critical loads is shown in Fig. 4. The LOS model on a lattice configuration for $L = 4$ is shown in Fig. 4(a). In the lattice configuration the LOS model fits a power law distribution $Prob(TT) = \alpha TT^{-\beta}$ with $\beta = 3.5$ and $\alpha = 10^{5.9}$. The LOS model on SF networks with $L = 4.5$, $\langle k \rangle = 12$ is shown in Fig 4(b). With probability 0.97 the model fits a power law distribution $Prob(TT) = \alpha TT^{-\beta}$ with $\beta = 2.6$ and $\alpha = 10^{4.2}$. The LOS model on SF networks with $L = 5$, $\langle k \rangle = 8$ is shown in Fig 4(c) and $L = 5.5$, $\langle k \rangle = 4$ is shown in Fig 4(d). As can be seen from the figures, at the critical load as average degree $\langle k \rangle$ decreases the TT distributions loose their power-law scaling. Implying, at the critical load, the

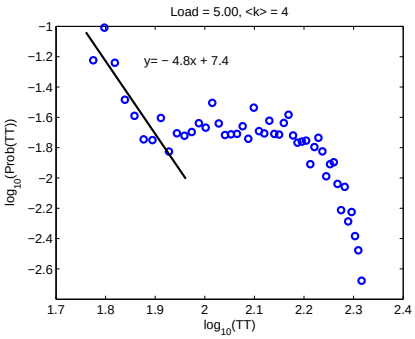
[3] In the literature there is a distinction between errors and attacks. Attacks target specific nodes where as random nodes are susceptible to errors. In this work we commence LOS dynamics for random components



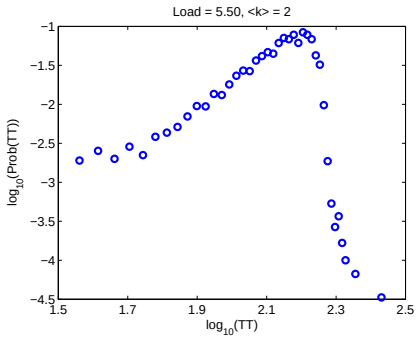
(a) $L = 4$ LOS Lattice model, Fit to $P(TT) = \alpha TT^{-\beta}$



(b) $L = 4.5$, $\langle k \rangle = 12$ LOS SF model, Fit to $P(TT) = \alpha TT^{-\beta}$



(c) $L = 5$, $\langle k \rangle = 8$ LOS SF model



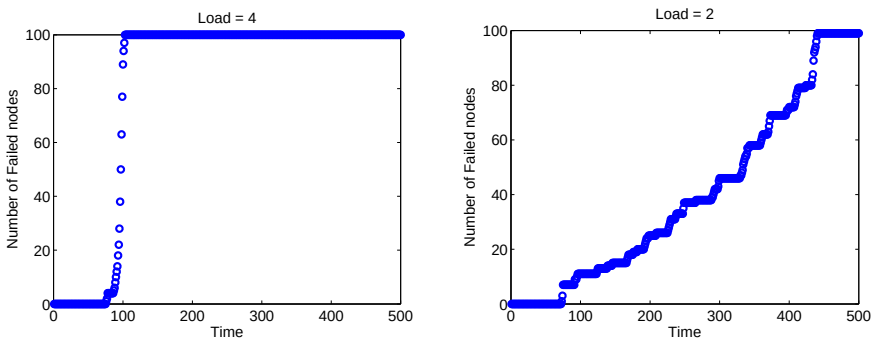
(d) $L = 5.5$, $\langle k \rangle = 4$ LOS SF model

Figure 4: LOS model TT distributions at the critical load. Note the breakdown in power law scaling with decreasing average network connectivity.

LOS model loses power-law scale invariance in system failure time distribution with decreasing average network connectivity $\langle k \rangle$.

2.4.3 Phase diagram

The phase diagram of the LOS lattice model is shown in Fig. 5. The LOS lattice model demonstrates phase diagrams similar to both first-order and second-order phase transitions. At the critical load $L = 4$, cascades of load redistributions induce massive failure causing all systems to fail as shown in Fig. 2(g). The corresponding first-order phase diagram is shown in Fig. 5(a). The loading at $L = 4$ is such that cascading load redistributions induce failure with minimal LOS dynamics. At transition loadings



(a) First order transition at $L = 4$ (b) Second order transition at $L = 2$

Figure 5: Phase transition of the LOS lattice model

$L = 2$, systems undergoing complete failure [refer to Figs. 1(g), 2(a)] exhibit second-order phase transitions as shown in Fig. 5(b). For second-order phase transitions, the transition to complete system failure is a gradual process involving repetition of LOS dynamics and load redistributions cascading from one component to the next.

Here we note that first and second-order phase transitions for traffic congestion in complex networks were also reported in [36, 40]. In [36], the authors show that by increasing the probability of node congestion (from $\bar{\eta} = 0.05$ to $\bar{\eta} = 0.95$, where $\bar{\eta}$ is a parameter to control node congestion probability) the traffic flow phase diagram switches from second-order to first-order. On the other hand in [40], the first or second-order phase transitions depend on the particular traffic routing protocol utilized (shortest-path routing versus traffic-aware routing). In comparison for the LOS model, by increasing the load from $L = 2$ to $L = 4$, the component failure phase diagram switches from second-order to first-order.

2.4.4 Failure modes and extremal behavior

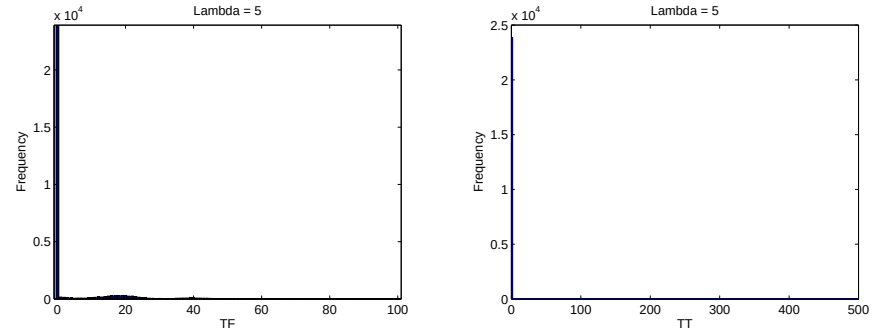
The TF and TT distributions for the CS model on a lattice configuration are shown in Fig. 6. The CS model on a SF network demonstrates qualitatively similar distributions. Loadings $\lambda = 6, 7$ correspond to transition loadings for these systems. The multi-modal nature of the TF distributions in Figs. 6(c), 6(e) indicate that multiple

failure modes are present in the distributions. Multi-modal distributions have been observed in nature in the eruption of geysers [4] and sizes of ants [103].

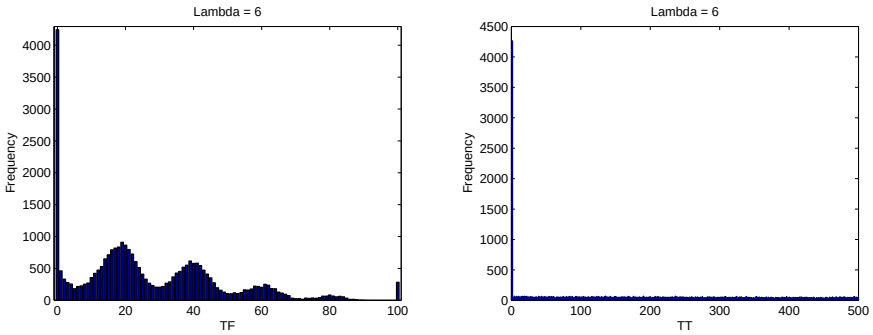
To isolate and identify the different failure modes, we filter the TF distributions based on the capacities of the failed components. Recall that in simulating the CS model we initialize component capacities from a integer uniform distribution $\mathcal{U}[8, 12]$. In Fig. 7(a) we color code the TF distribution for $\lambda = 6$ [Fig. 6(c)] based on the capacities of the failed components. From Fig. 7(a) the composition of the different failure modes becomes clear. The TF distribution for $\lambda = 6$ is composed of a failure mode where only components of capacity= 8 fail, a second failure mode where only components of capacity= $(8, 9)$ fail, a third failure mode where only components of capacity= $(8, 9, 10)$ fail and so on. Similarly we can filter the TF distribution for $\lambda = 7$.

It is also of interest to understand the dynamics that are exciting the multiple failure modes for transitions loadings $\lambda = 6, 7$. Motivated by Extreme Value Theory [47, 44] one explanation lies in the demand dynamics. Although the average demand on the system is $\lambda = 6$ or 7; the CS model is sensitive to the extremal behavior of the demand dynamics. Extremal events have been modeled in areas as diverse as finance [41] to earthquake characterization [83]. In Fig. 7(b) we plot the extremal behavior of the demand dynamics as a function of TF for $\lambda = 6$. The figure is constructed in the following way: in Fig. 7(a) for each TF $\in [1, 100]$, we first determine the maximum demand seen by each of the systems in their associated window $[0, TT]$. For each TF $\in [1, 100]$ we then compute and plot, the mean maximum demand (shown in blue), the maximum maximum demand (shown in red) and the minimum maximum demand (shown in green).

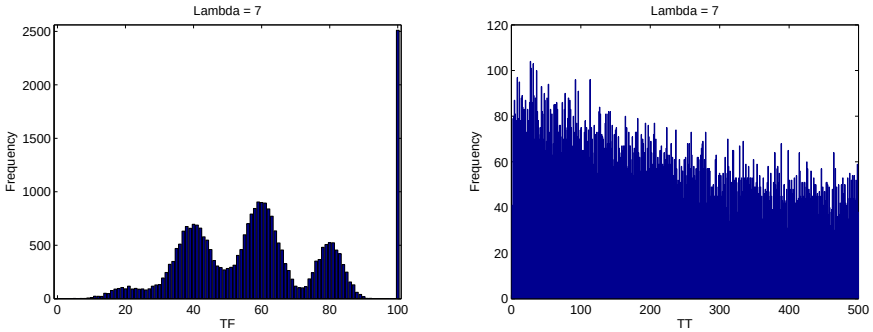
In Fig. 7(b) we can clearly observe the staircase like growth trend of mean maximum demand as a function of TF and the step function growth of the associated min/max bounds of maximum demand. It is our opinion the extremal behavior of



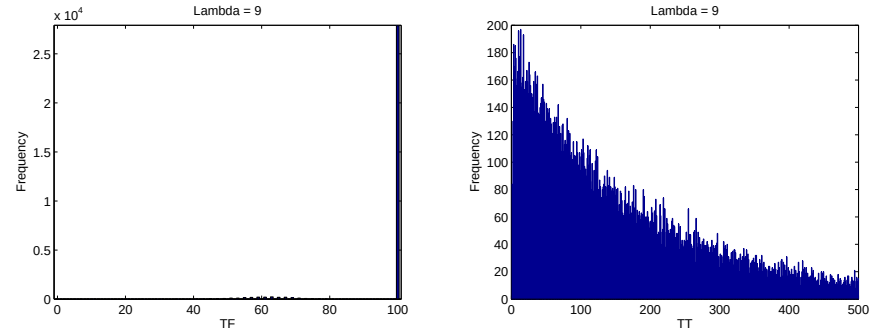
(a) $\lambda = 5$ (b) $\lambda = 5$



(c) $\lambda = 6$ (d) $\lambda = 6$

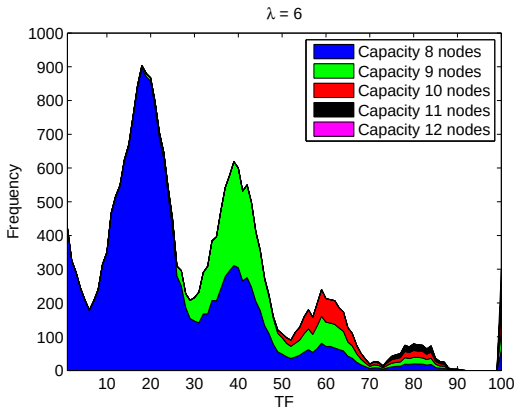


(e) $\lambda = 7$ (f) $\lambda = 7$

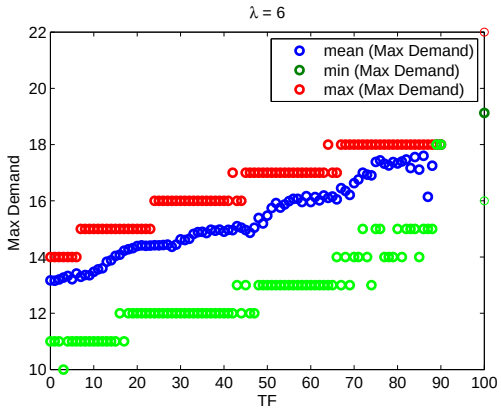


(g) $\lambda = 9$ (h) $\lambda = 9$

Figure 6: CS model distributions. TF distributions (left column); TT distributions (right column).

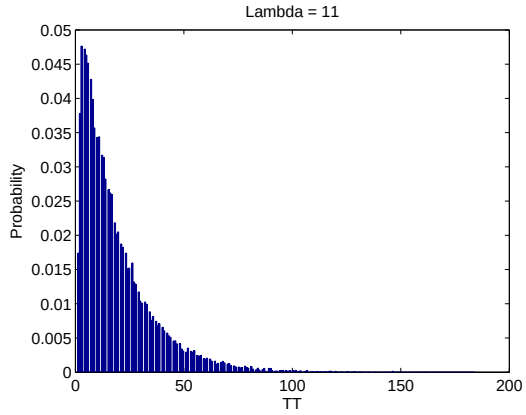


(a) Multiple failures modes

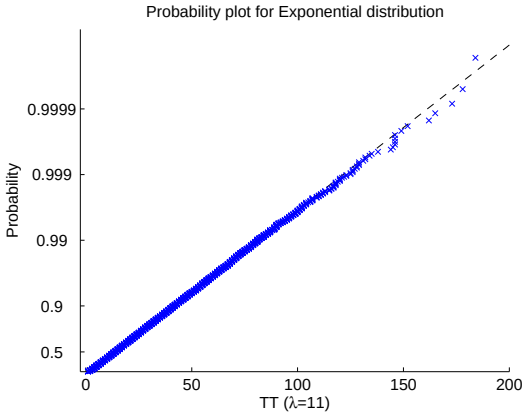


(b) Maximum Demand as a function of TF

Figure 7: Extremal behavior of the CS model for $\lambda = 6$ (Color Online)



(a) $\lambda = 11$, TT distribution



(b) $\lambda = 11$, CS model Fit to $P(TT) = Exp(\mu)$

Figure 8: CS model TT distribution fit at the critical load

the demand dynamics [4] in conjunction with the structure of the component capacity topology is responsible for exciting the multiple distinct failure modes observed in Fig. 7(a).

For example in Fig. 7 consider the interval TF $\in [10, 25]$; mean maximum demand in this interval roughly corresponds to around 14 with components of capacity= 8 failing. Noting that the queue size is 6, we can understand why components of capacity= 8 are being overwhelmed by the mean maximum demand $(8 + 6 = 14)$ in this interval. However, in addition to the specific sequence and number of extremal demands, relatively stronger neighborhood capacity topologies are partly responsible for the left side of the bell shape and relatively weaker neighborhood capacity topologies are partly responsible for the right side of the bell shape in the interval TF $\in [10, 25]$. For a specific level and sequence of extremal demand, a relatively stronger neighborhood capacity topology provides components a greater opportunity to survive through load sharing. We could construct similar arguments for the other bell curve like waves in Fig. 7 such as the interval TF $\in [30, 50]$ where components of capacity= $(8, 9)$ are failing and mean maximum demand is approximately 15.

The result in Fig. 7 is similar in spirit to results in [61], where the authors show using shell models of turbulence that large fluid velocity fluctuations propagating from shell to shell cause multiscaling in the shell velocity variation distributions. In other words the velocity variation distribution is composed of two separate regions, the first due to "small" but "usual" velocity fluctuations and the second due to "large" but "rare" velocity fluctuations. Comparing to our results in Fig. 7, we can see the extremal demand dynamics exciting different scales of failure in the TF distribution.

The TT distribution fit for the CS lattice model at the critical load $\lambda = 11$ is shown in Fig. 8. At the critical load the CS model fits a exponential distribution

[4] The number, size and sequence of extreme demand constitute the extremal behavior of the demand dynamics.

$Prob(TT) = \frac{1}{\mu} e^{-TT/\mu}$ with parameter $\mu = 19.06$ [Fig. 8(b)]. The CS model on SF networks demonstrates similar results. In communication and transportation applications M/M/1 queues have arrivals according to Poisson processes and service time distributions are exponential [12]. Although individual components in the CS model resemble M/D/1 queues, at critical demand rates the load sharing capability of the CS model is rendered redundant and the structure of the component capacity topology causes the system to demonstrate exponential distribution failure times. Here we also note that exponential and sub-exponential distributions have been widely reported in financial applications such as draw-downs of the stock market, major currencies and major financial indices [90, 52]. The relationship between the extremal dynamics of the CS model and market draw-downs presents an interesting subject for future investigation.

2.5 Accuracy of Monte Carlo simulations

We simulated LOS and CS models for each parameter setting of L and λ on both lattice and scale-free network configurations for 30 000 cases. Simulations were conducted on two Intel Pentium 64-bit desktops. For each setting, simulations were conducted multiple times. Regardless of the desktop used the simulation results were always identical. Since the results were identical over multiple runs and over different desktops, we concluded 30 000 cases were a sufficient numbers of runs.

Also we would like to point out, the shapes of the distribution would not change if the number of runs were increased. Numerical precision of the distributions would increase from increased number of simulations but qualitatively the shapes would remain the same. Ultimately, we are analyzing the behavior of the systems.

2.6 Discussion

To summarize, we have used the concept of component strength and load interaction to investigate the failure mechanisms of complex systems utilizing two different

strength/load interaction models. The LOS model explores strength/load interaction through a loss of strength mechanism. The CS model explores capacity/demand interaction through customer or data arrival/departure rate mechanism. At low levels of loading which correspond to lower network utilization, the failure mechanisms in the LOS model follow predictable trends [Eq. 1 and Fig. 3]. The failures in the systems can be managed and network resources are sufficiently allocated. The system is resilient to cascading failure triggered by load redistribution.

At transition loadings or 'tipping-points', both models demonstrate increasingly unpredictable behavior with system volatility and increasing disorder. Systems may or may not descend into catastrophic failure and extremal dynamics excite multiple failure modes in systems. The results imply that at these loadings the system resources (characterized by the system strength topology) need to be allocated appropriately to avoid catastrophic failure.

For critical loads system failure is reached through phase transitions. At criticality, depending on the strength/load interaction mechanism, systems demonstrate power law or exponential temporal failure patterns.

Table 1: Experimental values for the temporal system failure scaling phenomena of Equation. 1

	L	κ_1	τ_1	Residual Norm	Breakpoint (TF)	κ_2	τ_2	Residual Norm
LOS Model	0.5	28.5	0.9	0.16	-	-	-	-
on a	1	46.4	0.66	0.16	-	-	-	-
Lattice network	1.5	61	0.7	0.13	25	428	0.05	0.28
	L	κ_1	τ_1	Residual Norm	Breakpoint (TF)	κ_2	τ_2	Residual Norm
LOS Model	0.5	21.54	0.83	0.29	-	-	-	-
on a	1	24.24	0.77	0.23	-	-	-	-
Scale-Free network	1.5	21.54	0.83	0.13	33	432.87	0.09	0.11
$\langle k \rangle = 12$	2	41.24	0.76	0.09	29	342.93	0.14	0.04

CHAPTER III

MATHEMATICAL MODEL DEVELOPMENT

In the previous chapter, we provided a simulation based analysis of the failure mechanisms of the LOS and CS models. In this chapter, we provide a mathematical treatment of the LOS and CS models. First, we analyze the cascading failure mechanism of the LOS model. Next, we provide a Markov chain formulation of the CS model. We conclude by comparing the results of the Markov chain formulation with the results obtained from the CS model simulations.

3.1 Cascading failure property of the LOS mechanism

We define cascading failure as the failure process where a components failure and subsequent load redistribution initiates the failure process for it's neighbor; the neighbor's failure and subsequent load redistribution initiates the failure process for the neighbor's neighbor; and the cycle keeps repeating in this manner.

It is of interest to us to mathematically define the conditions determining the presence or absence of cascading failure for components possessing LOS dynamics. We will first investigate these conditions for components in a 1d or chain configuration and then discuss the extension of our results to lattice and other network configurations.

Consider a 1d or chain arrangement of components where each component can undergo the LOS mechanism as shown in Fig. 9(a). Assume initially the LOS mechanism is initiated for only one component and the subsequent load redistribution triggers the neighboring component to commence LOS. Repetition of this process results in a progression of cascading failures over time. If a component's load redistribution fails to initiate a neighbor's LOS mechanism then the cascading failure process terminates.

We consider two cases, the first where a component at the boundary of the chain

initiates cascading failure as shown in Fig.9(a). The second is when a component in the interior of the chain initiates cascading failure as shown in Fig.10(a).

3.1.1 1d cascading failure initiated by a boundary component

Consider Fig. 9, the initial conditions on the components are $L_1 \geq \mu S_1$ and $L_i < \mu S_i$ for $i \in [2, 7]$. Therefore component 1 commences LOS, fails at a future point in time and redistributes its load to component 2. If the load redistributions triggers LOS for component 2 then it too will fail some time later. Subsequently if the load redistribution from component 2 triggers LOS for component 3 than component 3 will also fail. If this process keeps repeating a chain of cascading failure results. If at any time a load redistribution does not trigger LOS for it's neighbor then the cascading failure process stops and all nodes downstream survive.

Mathematically we can express the cascading failure conditions for the 1d system in Fig. 9 as a set of equations shown in Eq. 2. In this equation set, component load due to redistribution is accounted for in the summation $\sum$.

$$\text{Initial Conditions: } L_1 \geq \eta S_1 \text{ and } L_i < \eta S_i \text{ for } i \in [2, 7] \tag{2a}$$

$$\text{Node 2: Load redistribution from } L_1,\ L_2 + \sum_{i=1}^{1} L_i > \eta S_2 \tag{2b}$$

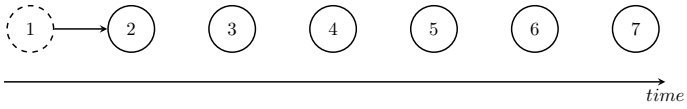
$$\text{Node 3: Load redistribution from } L_2,\ L_3 + \sum_{i=1}^{2} L_i > \eta S_3 \tag{2c}$$

$$\text{Node 4: Load redistribution from } L_3,\ L_4 + \sum_{i=1}^{3} L_i > \eta S_4 \tag{2d}$$

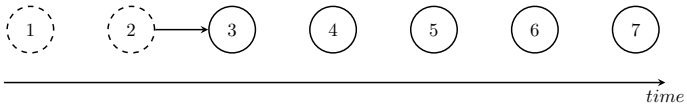
$$\text{Node 5: Load redistribution from } L_4,\ L_5 + \sum_{i=1}^{4} L_i > \eta S_5 \tag{2e}$$

$$\text{Node 6: Load redistribution from } L_5,\ L_6 + \sum_{i=1}^{5} L_i > \eta S_6 \tag{2f}$$

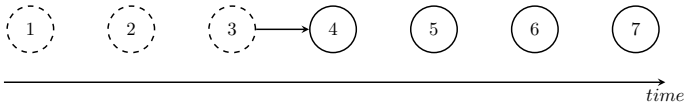
$$\text{Node 7: Load redistribution from } L_6,\ L_7 + \sum_{i=1}^{6} L_i > \eta S_7 \tag{2g}$$



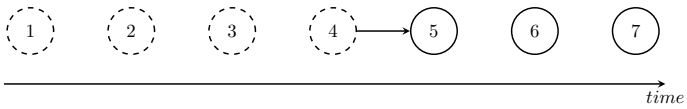
(a) Cascading failure initiated at node 1



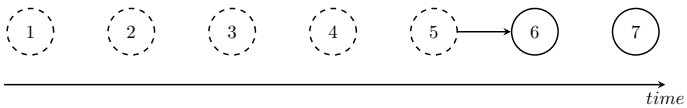
(b) Node 1 fails. Node 2 commences LOS.



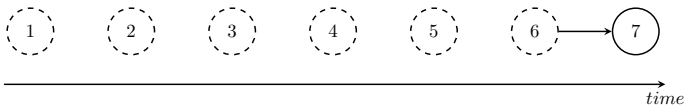
(c) Node 2 fails. Node 3 commences LOS.



(d) Node 3 fails. Node 4 commences LOS.



(e) Node 4 fails. Node 5 commences LOS.



(f) Node 5 fails. Node 6 commences LOS.

(g) Node 6 fails. Node 7 commences LOS.

Figure 9: Cascading failure initiated by a boundary component. Solid nodes indicate fully operational components. Dashed nodes indicate LOS or Failure. The plots show the progression of failure in time.

Eq. 2a defines the initial configuration. Eq. 2b defines the condition for component 2 to fail after component 1 has failed. Eq. 2c defines the condition for component 3 to fail after component 2 has failed and so on.

In equation set 2 if any one of the equations failed to hold then components downstream of the penultimate load redistribution would not fail. For example, if Eq. (2d) fails to hold than the cascading failure process terminates and components 4 through 7 would survive.

If the inequalities in Eq. 2 hold sequentially then there is a important observation regarding the load that is redistributed. In Eq. 2 as cascading failure progresses from component to component, the load due to redistribution, the load terms in the summation $\sum$, increase sequentially. For example the load due to redistribution in Eq. 2d is greater than the load due to redistribution in Eq. 2c which in turn is greater than the load due to redistribution in Eq. 2b. An appropriate analogy here would be to tidal waves or tsunami's in nature. As tidal waves approach the shore their height keeps steadily increasing, similarly as cascading failure progresses the load due to redistribution keeps steadily increasing.

From these set of equations we observe that, provided knowledge of the distribution of component strength and load, we can a-priori calculate whether cascading failure will be present or not if any of the components undergo LOS dynamics. We can also a-priori calculate the number of components that will fail.

To summarize, for failure initiated at a boundary component of a 1d n-component system, we can write the cascading failure conditions compactly as,

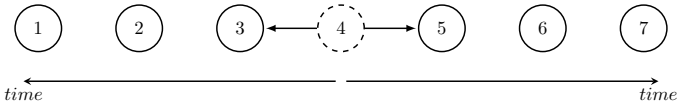
$$\text{Initial Conditions: } L_1 \geq \eta S_1 \text{ and } L_i < \eta S_i \text{ for } i \in [2, n] \tag{3a}$$

$$\text{Node k: Load redistribution from } L_{k-1},\ L_k + \sum_{i=1}^{k-1} L_i \geq \eta S_k \text{ for } k \in [2, n] \tag{3b}$$

If Eq. 3b holds sequentially with increasing k then the n-component system will undergo cascading failure.

3.1.2 1d cascading failure initiated by a interior component

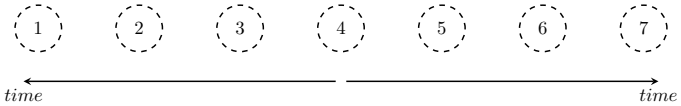
In a similar fashion we can derive the cascading failure conditions for the component configuration in Fig. 10. Here the initial conditions are given by $L_4 \geq \eta S_4$ and $L_i < \eta S_i$ for $i \in [1, 7]$ and $i \neq 4$. In this case, cascading failure might progress in 2 directions, to the left and right of component 4. After component 4 fails, its load is redistributed to components 3 and 5 as shown in Fig. 10(b). If components 3 and 5 commence LOS and fail then their load is redistributed to components 2 and 6 as shown in Fig. 10(c). In Fig. 10 cascading failure is progressing in 2 directions.



(a) Cascading failure initiated at node 4.

(b) Node 4 fails. Nodes 3 and 5 commence LOS.

(c) Nodes 3, 5 fail. Nodes 2 and 6 commence LOS.



(d) Nodes 2, 6 fails. Node 1 and 7 commence LOS.

Figure 10: Cascading failure initiated by a interior component. Solid nodes indicate fully operational components. Dashed nodes indicate LOS or Failure. The plots show the progression of failure in time.

For cascading failure on the right of component 4, the set of equations are given by Eq. 4. In this equation set, component load due to redistribution is accounted for

by the terms in the brackets ( ).

$$\text{Initial Conditions: } L_4 \geq \eta S_4 \text{ and } L_i < \eta S_i \text{ for } i \in [5,7] \tag{4a}$$

$$\text{Node 5: Load redistribution from } L_4,\ L_5 + (\frac{L_4}{2}) > \eta S_5 \tag{4b}$$

$$\text{Node 6: Load redistribution from } L_5,\ L_6 + (\sum_{i=5}^{5} L_i + \frac{L_4}{2}) > \eta S_6 \tag{4c}$$

$$\text{Node 7: Load redistribution from } L_6,\ L_7 + (\sum_{i=5}^{6} L_i + \frac{L_4}{2}) > \eta S_7 \tag{4d}$$

For cascading failure on the left of component 4, the set of equations are given by Eq. 5.

$$\text{Initial Conditions: } L_4 \geq \eta S_4 \text{ and } L_i < \eta S_i \text{ for } i \in [1,3] \tag{5a}$$

$$\text{Node 3: Load redistribution from } L_4,\ L_3 + (\frac{L_4}{2}) > \eta S_3 \tag{5b}$$

$$\text{Node 2: Load redistribution from } L_3,\ L_2 + (\sum_{i=3}^{3} L_i + \frac{L_4}{2}) > \eta S_2 \tag{5c}$$

$$\text{Node 1: Load redistribution from } L_2,\ L_1 + (\sum_{i=2}^{3} L_i + \frac{L_4}{2}) > \eta S_1 \tag{5d}$$

From the above equations we can derive the mathematical conditions for failure progression for a 1d n-component system with failure initiated at a interior component p,

Initial Conditions:

$$L_p \geq \eta S_p \text{ for } p \in (1,n) \text{ and } L_i < \eta S_i \text{ for } i \in [1,n] \text{ and } i \neq p \tag{6a}$$

Cascading failure conditions to the right of component p:

$$\text{Node } p+1\text{:Load redistribution from } L_p,\ L_{p+1} + (\frac{L_p}{2}) \geq \eta S_{p+1} \tag{6b}$$

$$\text{Node } k\text{: Load redistribution from } L_{k-1},\ L_k + (\sum_{i=p+1}^{k-1} L_i + \frac{L_p}{2}) \geq \eta S_k \text{ for } k \in [p+2, n] \tag{6c}$$

Cascading failure conditions to the left of component p:

$$\text{Node } p-1\text{:Load redistribution from } L_p,\ L_{p-1} + (\frac{L_p}{2}) \geq \eta S_{p-1} \tag{6d}$$

$$\text{Node } k\text{: Load redistribution from } L_{k+1},\ L_k + (\sum_{i=k+1}^{p-1} L_i + \frac{L_p}{2}) \geq \eta S_k \text{ for } k \in [1, p-2] \tag{6e}$$

3.1.3 Cascading failure in 2d or 3d

From the mathematical equations for cascading failure in 1d, it is clear that the equations for cascading failure in 2d or 3d can become quite complicated. For example, in 1d cascading failure might progress in at most 2 directions. However for a 2d lattice configuration, for each component cascading failure might progress in 8 possible directions. It is clear that the number of paths the cascading failure process might take for a lattice configuration of components will quickly become countably infinite. The same argument applies for components arrangement in a graph network.

However, the mathematical equations for 1d cascading failure from the previous sections provide us adequate insight on the time progression of cascading failure in systems and provide us a guideline on how to initialize 2d or 3d systems to achieve desired failure directions and patterns. For example, suppose it is desired to have failure progression in a particular direction in a 2d system. In this case the system could be initialized with strengths S_{ij} following a "canyon" topology and the loads L_{ij} initialized accordingly. The result would be that failure would follow the route of the canyon. This is similar to a flowing river in a canyon, which only follows the route dictated by the topology of the canyon.

From the above discussion it is clear, we can initialize the 2d or 3d system as appropriate to achieve the desired directions and patterns of failure.

3.2 Markov formulation of the CS model

For networks where components undergo failure through the CS mechanism we can model the CS mechanism through Discrete Time Markov Chains.

Consider a network where component demand is the same for all component. Let D be a discrete random variable whose probability mass function (pmf) is given by $p_D(i) = P(D = i)$, where $i \in (0, 1, 2, 3, ...)$. For ease of notation for the rest of the discussion we write $D = i$ as D_i. Let C denote the capacity of a particular component and C_{Lj} the capacity of the jth neighbor where $j \in [1, n]$ and n is the number of neighbors of the component. Also let k denote the length of the queue and q the queue occupancy where $q \leq k$.

For ease of derivation let us consider a component which has 2 neighbors as depicted in Fig. 11. In the figure, four cases are pointed out. Let us assume for all cases that queues are unoccupied.

- Case 1) In figure 11(a), the demand on the component is less than or equal it's capacity $D_i \leq C$ hence the component does not need to share its load with it's neighbors.

- Case 2) In figure 11(b), the demand is greater than the component capacity $D > C$ but less than the capacity of neighbor's $D \leq C_{L1} < C_{L2}$.

- Case 3) In figure 11(c), the demand is greater than the component's capacity and also the neighbor 1's capacity $D_i > C > C_{L1}$ but less than the capacity of neighbor 2 implying load sharing with neighbor 2 is possible but not neighbor 1. For case 2 and 3, we would like to compute the probability that the neighbor's possess sufficient excess capacity and/or the queue k has sufficient vacancy to

accommodate the excess demand $D_i - C$.

- Case 4) Finally in figure 11(d), the demand is greater than capacity of the components and both neighbor's, in this case we would like to know the probability that there is enough space in the queue to meet the excess demand.

3.2.1 CS mechanism Markov chain model

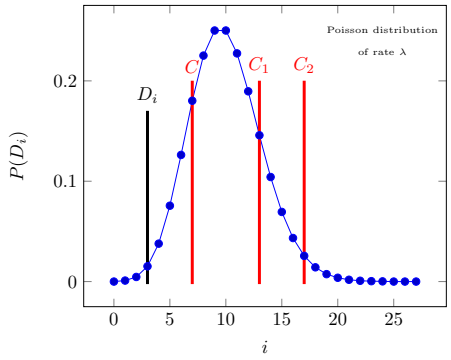
From the above discussion we can define 4 states that a component can reside in. Let C_o define the state that the component is able to meet it's demand without resorting to load sharing with neighbor's or utilizing the queue. Let C_L denote the state that the demand is greater than component capacity but the excess demand $D_i - C$ is completely satisfied through load sharing with neighbor's. Let B_k denote the state where excess demand is met through a combination of load sharing and utilizing the queue or purely through utilizing the queue. Finally, let F denote the state where the excess demand is not met through load sharing and the queue. Note, F is an absorbing state and component remains in F once it enters F. Figure 12 depicts the Markov chain model for the CS mechanism.

3.2.1.1 Transition probabilities for state C_o

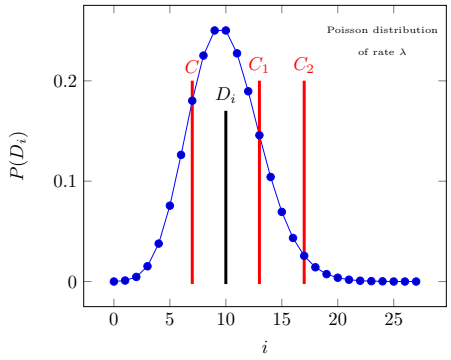
Next, we derive the state transition probabilities. We do this for the general case of n neighbors. Let S denote the set of the neighbor's out of n that are available for load sharing. Let p,determined using equations 7, 8, denote the total number of neighbor's that are available for load sharing.

$$p = \sum_{\omega=1}^{n} I(\omega) \tag{7}$$

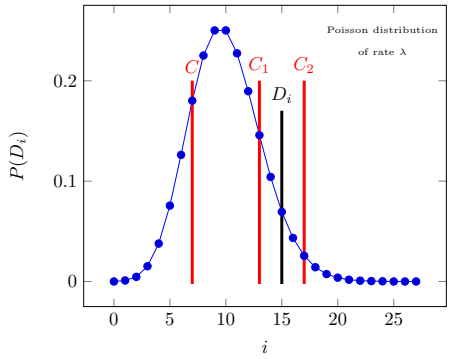
$$I(\omega) = \begin{cases} 1 & : \omega \in S \\ 0 & : \omega \notin S \end{cases} \tag{8}$$



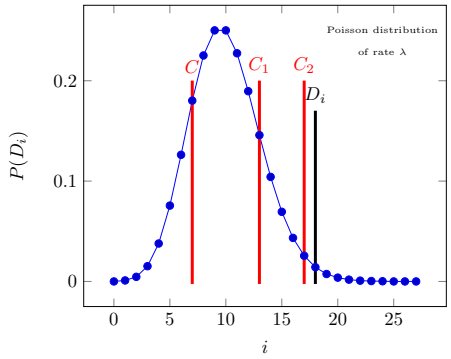
(a) D < C



(b) C < D < CL



(c) D > CL



(d) D > CL

Figure 11: Distribution plot for different D, C, CL

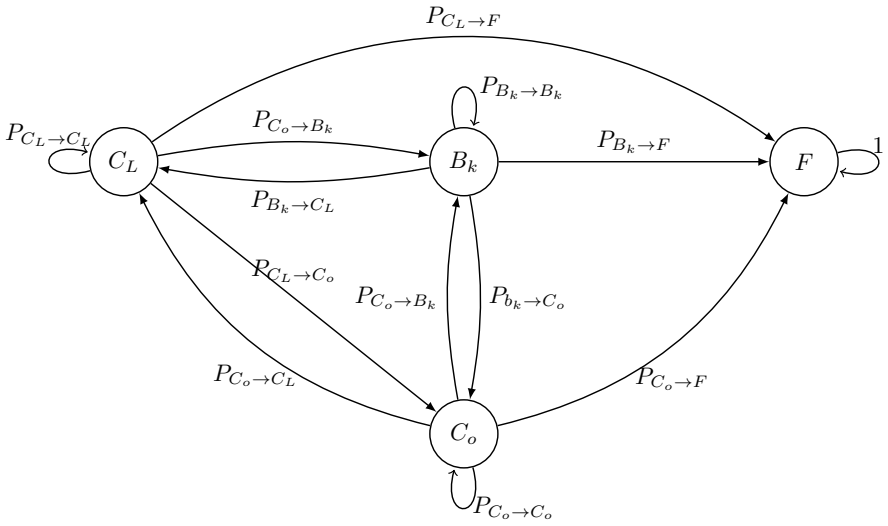


Figure 12: DTMC

It follows from fig 11(a), if the component is in state C_o then it will remain in state C_o with the transition probability:

$$P_{C_o \to C_o} = P(D < C) = \sum_{r=0}^{C} p_D(r) \tag{9}$$

If $D_i > C$ than the component will attempt to share the excess demand with available neighbors. Assume there are neighbors available with excess capacity who can fully satisfy the excess demand, this situation is depicted in figure 11(b). In that case the transition probability from C_o to C_L is given by equations 10, 11 [1]. Note that Eq. 11 is essentially a maximization over the random variable D for the given constraints.

[1]We would like to thank Prof. Panagiotis Tsiotras for his comments on mathematical notation for this problem.

$$P_{C_o \to C_L} = P(D - C \leq (\sum_{\omega=1,\omega\in S}^{n} C_\omega - pD)) = \sum_{r=C+1}^{g} p_D(r) \tag{10}$$

where the parameter g is the solution to the equation:

$$g = \sup\{y : y - C \leq (\sum_{\omega=1,\omega\in S}^{n} C_\omega - py)\} \tag{11}$$

s.t.

$$C < y < C_{L\omega} \mid_{\omega\in S}$$

If the component is able to meet excess demand with load sharing then it transitions to state C_L. There might be situations where the excess demand is partially met through load sharing and the remainder is successfully placed in the queue. In that case the component transitions to state B_k. Figures 11(c) and 11(d) depict a possible situation where this might arise. The transition probability from state C_o to B_k is given by the equations 12, 13.

$$P_{C_o \to B_k} = P(D - C - \sum_{\omega=1,\omega\in S}^{n} (C_\omega - pD) \leq k) = \sum_{r=g+1}^{f} p_D(r) \tag{12}$$

where the parameter f is determined from equation:

$$f = \sup\{y : y - C - (\sum_{\omega=1,\omega\in S}^{n} C_\omega - py) \leq k\} \tag{13}$$

s.t.

$$C < y < C_{L\omega} \mid_{\omega\in S}$$

Finally there might be situations where load sharing is not possible and the queue does not have adequate capacity to meet the excess demand. In that case the component transitions to state F. Figure 11(d) depicts a possible case when this might occur. The transition probability from state C_o to state F is given by equation 14.

$$P_{C_o \to F} = P(D - C - (\sum_{\omega=1, \omega \in S}^{n} C_\omega - pD) - k > 0) = \sum_{r=f+1}^{\lfloor i \rfloor} p_D(r) \tag{14}$$

Here $\lfloor i \rfloor$ denotes the floor function.

3.2.1.2 Transition probabilities for state C_L

If the component is in state C_L the transition probabilities are given by the equation set 15.

$$\begin{aligned} P_{C_L \to C_o} &= P_{C_o \to C_o} \\ P_{C_L \to C_L} &= P_{C_o \to C_L} \\ P_{C_L \to B_k} &= P_{C_o \to B_k} \\ P_{C_L \to F} &= P_{C_o \to F} \end{aligned} \tag{15}$$

3.2.1.3 Transition probabilities for state B_k

If the component is in state B_k then in computing the transition probabilities we need to take into account the queue occupancy q ($q \leq k$). We assume the component capacity $C > k$, the queue capacity. The transition probability from B_k to C_o is given by equation 16.

$$P_{B_k \to C_o} = P(D < C - q) = \sum_{r=0}^{C} p_D(r) \tag{16}$$

The transition probability from B_k to C_L is given by Eq. 17.

$$P_{B_k \to C_L} = P(D - (C - q) \leq (\sum_{\omega=1,\omega\in S}^{n} C_\omega - pD)) = \sum_{r=C+1}^{g} p_D(r) \tag{17}$$

where the parameter g is the solution to the equation:

$$g = \sup\{y : y - (C - q) \leq (\sum_{\omega=1,\omega\in S}^{n} C_\omega - py)\} \tag{18}$$

s.t.

$$C < y < C_{L\omega} \mid_{\omega\in S}$$

The transition probability from B_k to B_k is given by Eq. 19.

$$P_{B_k \to B_k} = P(D - (C - q) - \sum_{\omega=1,\omega\in S}^{n} (C_\omega - pD) \leq k) = \sum_{r=g+1}^{f} p_D(r) \tag{19}$$

where the parameter f is determined from equation:

$$f = \sup\{y : y - (C - q) - (\sum_{\omega=1,\omega\in S}^{n} C_\omega - py) \leq k\} \tag{20}$$

s.t.

$$C < y < C_{L\omega} \mid_{\omega\in S}$$

The transition probability from B_k to F is given by Eq. 21.

$$P_{B_k \to F} = P(D - (C - q) - (\sum_{\omega=1,\omega\in S}^{n} C_\omega - pD) - k > 0) = \sum_{r=f+1}^{\lfloor i \rfloor} p_D(r) \tag{21}$$

Here $\lfloor i \rfloor$ denotes the floor function.

Although we derived the transition probabilities for queue occupancy q, we can derive the transition probabilities for each slot that the queue is occupied. So for a queue of size k, we can derive $4 * k$ transition probabilities. In this way we can track the queue occupancy through its occupancy state. For example if the queue size is $k = 3$, we would have states B_1, B_2, B_3 to track its occupancy. This is the preferred approach when numerically solving the equations for the transition probabilities.

3.2.1.4 Transition probabilities for state F

State F is an absorbing state. If it is reached than the components remain in state F with probability 1.

Using these transition probabilities we can construct the discrete time Markov chain shown in Fig 12.

3.2.2 Absorbing Markov Chains

It is clear from the description of the CS model Markov chain in the previous section that at high enough demand rates that the Markov chain is an absorbing Markov chain. In the next sections, we will work out some examples using the equations developed in the previous section. But before that we will present the equations of absorbing Markov chains that we will use subsequently. These equations are taken from the book "Finite Markov Chains" by Kemeny and Snell and specifically from chapter 3, "Absorbing Markov Chains" published by Springer-Verlag [53].

Let P denote the transition matrix of an absorbing Markov chain with t transient states and r absorbing states. Then matrix P can be partitioned as,

$$P = \begin{pmatrix} Q & R \\ M & I_r \end{pmatrix} \tag{22}$$

where Q is a $t \times t$ matrix, R is a $t \times r$ matrix,M is a $r \times t$ zero matrix and I_r is a $r \times r$ identity matrix.

With P partitioned in the above manner we can calculate the fundamental matrix N using the following equation,

$$N = \sum_{k=0}^{\infty} Q^k = (I_t - Q)^{-1} \tag{23}$$

where I_t is a $t \times t$ identity matrix.

We can then calculate the expected number of steps before being absorbed when starting in transient state i. This is given by the ith entry of the vector,

$$t = N\mathbf{1} \tag{24}$$

where $\mathbf{1}$ is column vector of all 1s of length t.

Many other properties of absorbing Markov chains can be computed such as the variance on number of visits, variance on number of steps, transient probabilities, absorbing probabilities etc. But for our purposes the equations above suffice for now.

3.2.3 Examples

We now demonstrate the CS mechanism Markov chain formulation using 2 simple examples. The examples provide insight on how the CS mechanism is operating.

3.2.3.1 Example 1

Consider a system composed of 3 components arranged in a ring configuration as shown in Fig. 13. We label the components as 1, 2 and 3. In this configuration each node has two neighbors for load sharing. We will derive the transition probability matrix for component 1. Assume component 1 has capacity $C = 5$, component 2 has capacity $C_{L1} = 10$ and component 2 has capacity $C_{L2} = 12$. The queue size is $k = 3$. The demand D is modeled as a discrete uniform random variable with pmf $p_D(i) = \frac{1}{14}$ with support $i \in \{1, 2, ..14\}$.

For load sharing purposes we assume component 1 has higher priority then component 2 and 3 and component 2 has higher priority then component 3. Also we assume in deriving the transition probabilities that the queue's in component 2 and 3 are not occupied.

Using equations 9 through 14 we derive the following transition probabilities for state C_o.

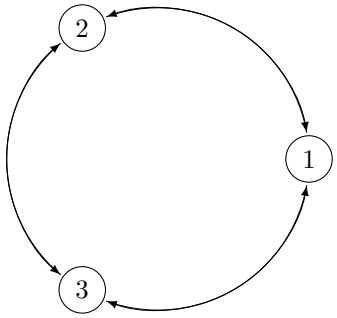


Figure 13: Ring of load sharing components

$$
\begin{aligned}
P_{C_o \to C_o} &= 5/14 \\
P_{C_o \to C_L} &= 3/14 \\
P_{C_o \to B_1} &= 1/14 \\
P_{C_o \to B_2} &= 0 \\
P_{C_o \to B_3} &= 1/14 \\
P_{C_o \to F} &= 4/14
\end{aligned}
$$

Using equations 15 we derive the following transition probabilities for state C_L.

$$
\begin{aligned}
P_{C_L \to C_o} &= 5/14 \\
P_{C_L \to C_L} &= 3/14 \\
P_{C_L \to B_1} &= 1/14 \\
P_{C_L \to B_2} &= 0 \\
P_{C_L \to B_3} &= 1/14 \\
P_{C_L \to F} &= 4/14
\end{aligned}
$$

Using equations 16 through 21 with $q = 1$ we derive the following transition probabilities for state B_1.

$$
\begin{aligned}
P_{B_1 \to C_o} &= 4/14 \\
P_{B_1 \to C_L} &= 4/14 \\
P_{B_1 \to B_1} &= 0 \\
P_{B_1 \to B_2} &= 1/14 \\
P_{B_1 \to B_3} &= 0 \\
P_{B_1 \to F} &= 5/14
\end{aligned}
$$

Using equations 16 through 21 with $q = 2$ we derive the following transition probabilities for state B_2.

$$
\begin{aligned}
P_{B_2 \to C_o} &= 3/14 \\
P_{B_2 \to C_L} &= 5/14 \\
P_{B_2 \to B_1} &= 0 \\
P_{B_2 \to B_2} &= 0 \\
P_{B_2 \to B_3} &= 1/14 \\
P_{B_2 \to F} &= 5/14
\end{aligned}
$$

Using equations 16 through 21 with $q = 3$ we derive the following transition probabilities for state B_3.

$$
\begin{aligned}
P_{B_3 \to C_o} &= 2/14 \\
P_{B_3 \to C_L} &= 6/14 \\
P_{B_3 \to B_1} &= 0 \\
P_{B_3 \to B_2} &= 0 \\
P_{B_3 \to B_3} &= 1/14 \\
P_{B_3 \to F} &= 5/14
\end{aligned}
$$

The state transition matrix P is given by.

$$
P = \begin{pmatrix}
0.3571 & 0.2143 & 0.0714 & 0 & 0.0714 & 0.2857 \\
0.3571 & 0.2143 & 0.0714 & 0 & 0.0714 & 0.2857 \\
0.2857 & 0.2857 & 0 & 0.0714 & 0 & 0.3571 \\
0.2143 & 0.3571 & 0 & 0 & 0.0714 & 0.3571 \\
0.1429 & 0.4286 & 0 & 0 & 0.0714 & 0.3571 \\
0 & 0 & 0 & 0 & 0 & 1.0000
\end{pmatrix} \tag{25}
$$

Since we have an absorbing Markov chain, we can calculate the fundamental matrix, denoted by N.

$$
N = \begin{pmatrix}
2.1439 & 0.7916 & 0.2097 & 0.0150 & 0.2270 \\
1.1439 & 1.7916 & 0.2097 & 0.0150 & 0.2270 \\
1.0058 & 0.8007 & 1.1290 & 0.0806 & 0.1452 \\
0.9292 & 0.8772 & 0.1290 & 1.0092 & 0.2166 \\
0.8578 & 0.9487 & 0.1290 & 0.0092 & 1.2166
\end{pmatrix} \tag{26}
$$

We can also calculate the expected number of steps before being absorbed when starting in transient state i. This is given by the vector,

$$t = \begin{pmatrix} 3.3871 \\ 3.3871 \\ 3.1613 \\ 3.1613 \\ 3.1613 \end{pmatrix} \tag{27}$$

3.2.3.2 Example 2

Consider a system composed of 3 components arranged in a ring configuration as shown in Fig. 13. We label the components as 1, 2 and 3. In this configuration each node has two neighbors for load sharing. Assume component 1 has capacity $C = 8$, component 2 has capacity $C_{L1} = 10$ and component 3 has capacity $C_{L2} = 12$. The queue size is $k = 6$. The demand D is modeled as a Poisson random variable with pmf $p_D(i) = \frac{\lambda^i}{i!}e^{-\lambda}$ with support $i \in \{0, 1, 2, ...\}$ with $\lambda = 9$. It is clear that component 1 is the weakest member of the ring network and we expect it to fail first.

For load sharing purposes we assume component 1 has higher priority then component 2 and 3 and component 2 has higher priority then component 3.

Using equations 9 through 21 we derive the following probability transition matrix for component 1. The states of the Markov chain are given by in sequence $C_o, C_L, B_1, B_2, B_3, B_4, B_5, B_6$ and F. Here F is the absorbing state. Also we assume in deriving the transition probabilities that the queue's in component 2 and 3 are not occupied when considering load sharing for component 1.

$$P = \begin{pmatrix} 0.4557 & 0.2503 & 0 & 0.0970 & 0 & 0.0728 & 0.0504 & 0.0324 & 0.0415 \\ 0.4557 & 0.2503 & 0 & 0.0970 & 0 & 0.0728 & 0.0504 & 0.0324 & 0.0415 \\ 0.3239 & 0.2635 & 0.1186 & 0 & 0.0970 & 0 & 0.0728 & 0.0504 & 0.0739 \\ 0.2068 & 0.3806 & 0 & 0.1186 & 0 & 0.0970 & 0 & 0.0728 & 0.1242 \\ 0.1157 & 0.4717 & 0 & 0 & 0.1186 & 0 & 0.0970 & 0 & 0.1970 \\ 0.0550 & 0.4007 & 0.1318 & 0 & 0 & 0.1186 & 0 & 0.0970 & 0.1970 \\ 0.0212 & 0.4344 & 0 & 0.1318 & 0 & 0 & 0.1186 & 0 & 0.2940 \\ 0.0062 & 0.4494 & 0 & 0 & 0.1318 & 0 & 0 & 0.1186 & 0.2940 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \tag{28}$$

Since we have an absorbing Markov chain, we can calculate the fundamental matrix, denoted by N.

$$N = \begin{pmatrix} 5.3753 & 3.5006 & 0.1269 & 1.0559 & 0.0908 & 0.8490 & 0.5278 & 0.5140 \\ 4.3753 & 4.5006 & 0.1269 & 1.0559 & 0.0908 & 0.8490 & 0.5278 & 0.5140 \\ 4.0464 & 3.3724 & 1.2409 & 0.8988 & 0.2107 & 0.7114 & 0.5496 & 0.4960 \\ 3.7352 & 3.3340 & 0.1197 & 1.9760 & 0.0906 & 0.8011 & 0.4239 & 0.5179 \\ 3.3602 & 3.1764 & 0.0938 & 0.7963 & 1.2017 & 0.6273 & 0.5136 & 0.3803 \\ 3.2341 & 3.0759 & 0.2599 & 0.7534 & 0.1034 & 1.7384 & 0.3935 & 0.5002 \\ 2.8443 & 2.8009 & 0.0835 & 0.8412 & 0.0605 & 0.5586 & 1.4707 & 0.3431 \\ 2.7712 & 2.7943 & 0.0796 & 0.6649 & 0.2266 & 0.5326 & 0.3496 & 1.4571 \end{pmatrix} \tag{29}$$

We can also calculate the expected number of steps before being absorbed in the absorbing state F when starting in any of the transient states, $C_o, C_L, B_1, B_2, B_3, B_4, B_5, B_6$. This is given by the vector,

$$t9 = \begin{pmatrix} 12.0402 \\ 12.0402 \\ 11.5262 \\ 10.9985 \\ 10.1495 \\ 10.0587 \\ 9.0028 \\ 8.8759 \end{pmatrix} \tag{30}$$

Repeating the example but this time with $\lambda = 11$, we get the expected number of steps before being absorbed in the absorbing state F as the vector,

$$t11 = \begin{pmatrix} 4.0041 \\ 4.0041 \\ 3.6897 \\ 3.4082 \\ 3.0377 \\ 2.9928 \\ 2.6104 \\ 2.5647 \end{pmatrix} \tag{31}$$

3.2.4 Comparison to simulation results

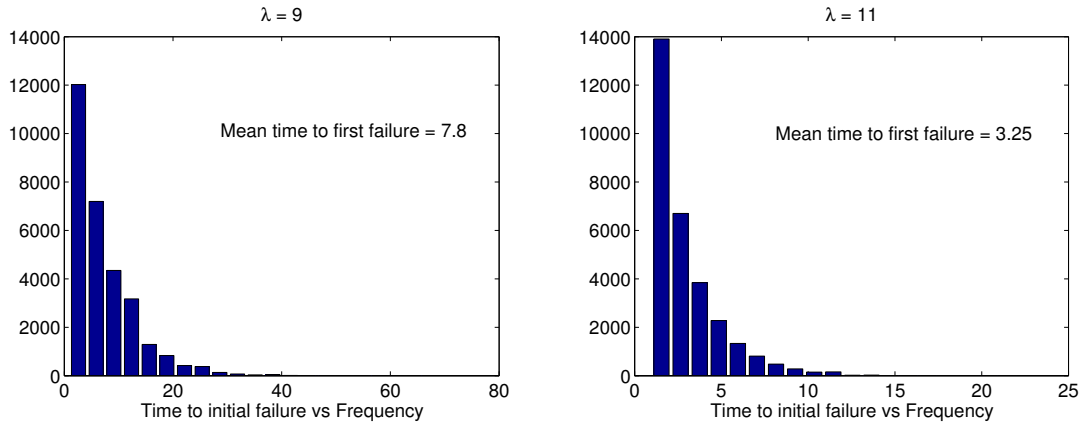
Now let us compare the theoretical results of the 3-node ring Markov chain model from Example #2 to results obtained from simulating CS model on different network configurations. For the simulations we consider four types of network configurations: a 3-node circular 1d ring, a 10-node circular 1d ring, a 144 node 2d lattice and a 100 node scale-free network. The 3-node and 10-node circular ring is the same as Fig. 13 but with 3 and 10 nodes respectively.

For the last three network configurations, the component capacities are initialized from a integer uniform distribution $\mathcal{U}[8, 12]$. For the 3-node ring configuration, component capacities are set to $8, 10, 12$ respectively. These capacity initializations are similar to the theoretical 3-node ring example. Components with capacity= 8 are the weakest links in the networks. The queue size is set to $k = 6$. The 3-node, 10-node circular ring network and the scale-free network by construction possess circular boundary conditions. For the lattice, queue size at the boundary is set very high to ensure components at the boundary do not fail. So for both the lattice and scale-free network at most 100 nodes can fail. We simulate all four network configurations for 500 time steps and carry out 30,000 Monte Carlo simulations.

In fig. 14 we show the time distributions for the first component failure for the lattice configuration for $\lambda = 9, 11$. Referring to Eq. 30, the first entry in the t9-vector is 12.04. This is the expected number of time steps before entering the absorbing state F from state C_o. C_o is the initial state of a component with queue empty. In the simulations for the lattice configuration at $\lambda = 9$, we obtained a mean time to first failure of 7.8. This is shown in fig. 14(a).

Similarly, referring to Eq. 31, the first entry in the t11-vector is 4.004. This is the expected number of time steps before entering the absorbing state from state C_o, which is the initial state of a component with buffer empty. In the simulations for lattice configuration at $\lambda = 11$, we obtained a mean time to initial failure of 3.25. This is shown in fig. 14(b).

In table 2 we tabulate for different values of λ, the mean time to first component failure for the four different network configurations obtained from simulations. We also tabulate the mean time to first component failure predicted by the 3-node ring markov chain model from Example #2. The theoretical and simulation results appear to be in good agreement. As λ increases the error between the theoretical and



(a) $\lambda = 9$ (b) $\lambda = 11$

Figure 14: Time to first component failure out of 100 components for the CS model

simulation results decreases. Also the simulations feature a greater number of components, so as expected, the mean time to first component failure in the simulations is of lower magnitude then the theoretical predictions.

In table 2 we observe that there is some difference in results between the 3-node Markov chain and 3-node simulation model. The difference in results between the 3-node Markov chain and the 3-node ring simulation model can be accounted for by the fact, that in the computing the transition probability matrix for the theoretical model we did not take into account the queue occupancy of neighbors. In calculating the transition matrix, we assumed that neighbors queue were empty. We only took into account the actual capacity of the neighbors and their difference with the demand. This approximation was chosen because the number of states in the transition matrix increases greatly if the queue occupancy of neighbors is taken into account. However, even with this approximation, the agreement between the 3-node ring Markov chain model and 3-node ring simulation model is quite good from $\lambda = 8$ to $\lambda = 13$.

Table 2: Comparison of theoretical and simulation results for mean time to first component failure

λ	Markov Chain Model	CS Model Simulations			
	3 node ring	3-node ring	10-node 1d ring	144 node 2d Lattice	100 node SF network
6	470.64	207.5	177.4	172.74	162.07
7	97.54	89.89	62.69	51.67	46.1
8	29.3	27.74	20.34	16.78	15.1
9	12.04	11.4	8.86	7.8	7.32
10	6.33	6.02	4.9	4.62	4.43
11	4.004	3.89	3.39	3.25	3.15
12	2.87	2.79	2.59	2.48	2.47
13	2.24	2.2	2.11	2.05	2.04

CHAPTER IV

LOAD SHARING, PUNCTUATED EQUILIBRIA AND FAILURE PROGRESSION CHARACTERISTICS

In this chapter we will extend our discussion of the CS and LOS extremal models to include additional properties and dynamics that were not discussed in the previous chapters. First, we will describe in more detail the load sharing properties of the CS model. Next, we will discuss SOC signatures of the CS model by drawing an analogy with the SOC signatures of the Bak-Sneppen evolution model. Finally, we will conclude the chapter with a discussion on the failure progression dynamics of the LOS model. We will show that failure progression of the LOS model is composed of different scales of failure progression. We will also show that the switch in the scales of failure progression can be reliably captured using 'change-pont' detection algorithms.

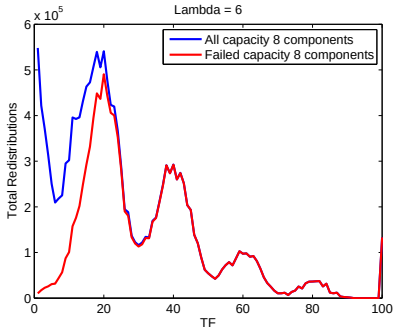
4.1 Renewal through load sharing for the CS model

Previously we have commented on the differences between the load sharing properties of the CS and LOS models. In the LOS model, a component's load redistribution is the final step before it fails. Once LOS dynamics is initiated the component will fail at a later point in time and on failure will attempt to redistribute its load to its neighbors. In the LOS model a component can undergo, at most, one load redistribution. In the CS model a component is essentially renewed through successful excess demand redistribution. The component fails only if the load redistribution is unsuccessful and the associated component queue q_{ij} is overwhelmed. In the CS model a component can complete multiple excess demand redistributions and remain fully operational. In

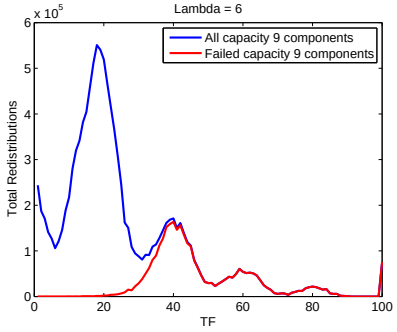
this section we will analyze the load sharing properties of the CS model for a deeper understanding of the system dynamics.

In Fig. 15 we plot the load redistributions of the CS model for $\lambda = 6$. The plots are constructed in the following way. Each component in the CS model is initialized with capacity from a uniform distribution $\mathcal{U}[8, 12]$. In a simulation, for components of a specific capacity, we compute the total number of load redistributions completed by the components collectively. We do this for all 30,000 Monte Carlo simulations. Next, for components of a specific capacity, for each TF $\in [0, 100]$ we sum the total number of load redistributions completed by the components. We do this for all components that failed and also for components that failed or survived. In Fig. 15(a) we plot the total load redistributions for components of capacity= 8 as function of TF in blue. We also plot in red,the total load redistributions for components of capacity= 8 that failed as a function of TF. We repeat this process for components of capacity= 9 in Fig. 15(b), for components of capacity= 10 in Fig. 15(c) and for components of capacity= 11 in Fig. 15(d). Similarly the load redistribution plots for $\lambda = 7$ are shown in Fig. 16.

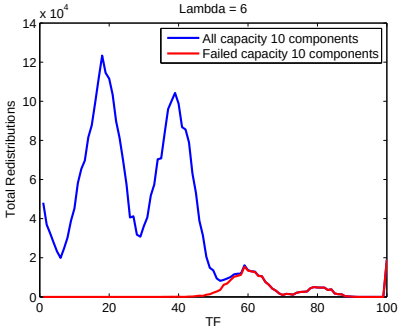
From Figs. 15 and 16 the renewal nature of the CS model load sharing becomes clear. For example, consider Fig. 15(c). In the figure we observe that up to TF=50, components of capacity= 10 complete large number of load redistributions but do not fail. From approximately TF=50 onwards we see that large number of components of capacity=10 complete load redistributions but at some point in time they fail. This is because from TF=50 onwards the extremal demand dynamics overwhelms the components. Nevertheless below TF=50, the components are able to renew themselves through load sharing and survive. Similar observations and analysis are possible for all the load sharing plots in Figs. 15 and 16, thus making clear the renewal nature of CS model load sharing.



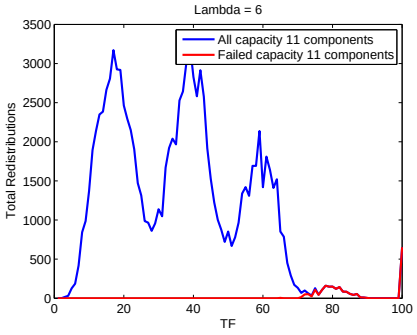
(a) Capacity = 8



(b) Capacity = 9

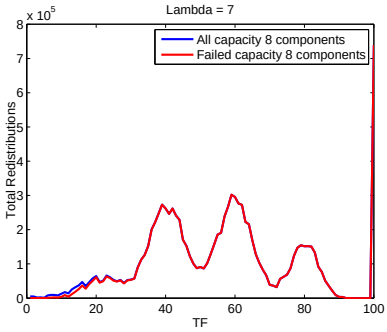


(c) Capacity = 10

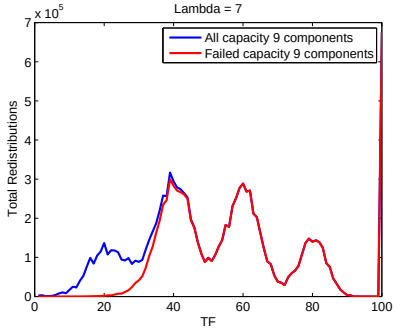


(d) Capacity = 11

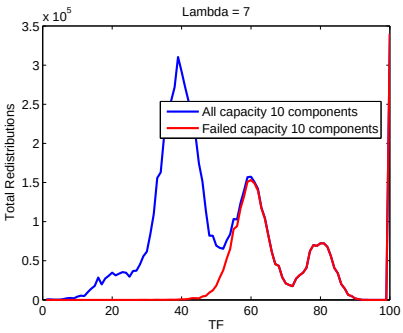
Figure 15: Load sharing for $\lambda = 6$



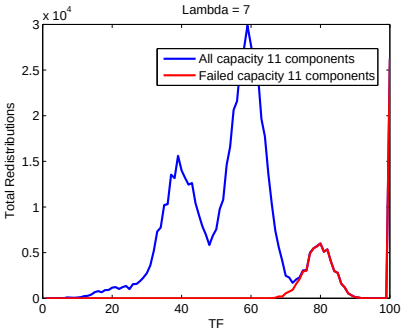
(a) Capacity = 8



(b) Capacity = 9



(c) Capacity = 10



(d) Capacity = 11

Figure 16: Load sharing for $\lambda = 7$

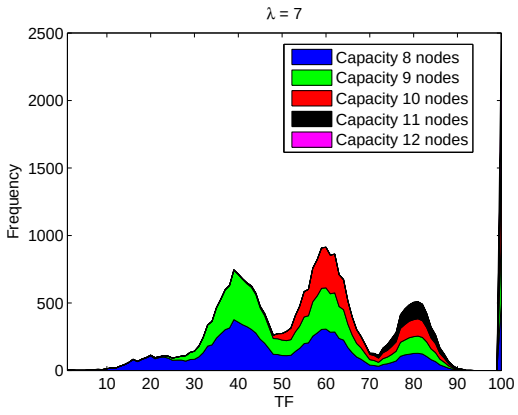
4.2 Extremal dynamics of the CS model

In chapter II we have illustrated the extremal dynamics of the CS model for $\lambda = 6$. Now we present the extremal dynamics of the CS model for $\lambda = 7$ in Fig. 17. Similar to the steps we took for $\lambda = 6$, we filter the TF distributions based on the capacities of the failed components to isolate and identify the different failure modes. In Fig. 17(a) we color code the TF distribution for $\lambda = 7$ [Fig. 6(e)] based on the capacities of the failed components. From Fig. 17(a) the composition of the different failure modes becomes clear. The TF distribution for $\lambda = 7$ is composed of a failure mode where only components of capacity= 8 fail, a second failure mode where only components of capacity= $(8, 9)$ fail, a third failure mode where only components of capacity= $(8, 9, 10)$ fail and so on.

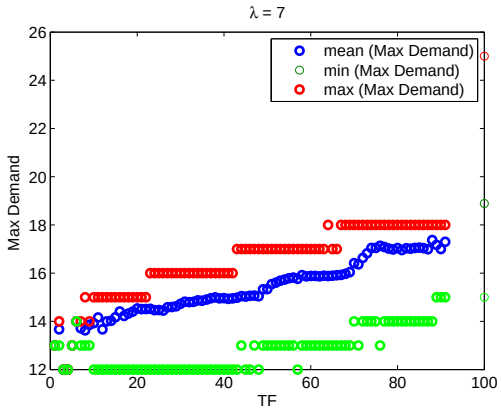
Similar to our findings for $\lambda = 6$, the CS model for $\lambda = 7$ is sensitive to the extremal behavior of the demand dynamics. The number, size and sequence of extreme demand constitute the extremal behavior of the demand dynamics. In Fig. 17(b) we plot the extremal behavior of the demand dynamics as a function of TF for $\lambda = 7$. The figure is constructed in the following way: in Fig. 17(a) for each TF $\in [1, 100]$, we first determine the maximum demand seen by each of the systems in their associated window $[0, TT]$. For each TF $\in [1, 100]$ we then compute and plot, the mean maximum demand (shown in blue), the maximum maximum demand (shown in red) and the minimum maximum demand (shown in green).

In Fig. 17(b) we can clearly observe the staircase like growth trend of mean maximum demand as a function of TF and the step function growth of the associated min/max bounds of maximum demand. The extremal behavior of the demand dynamics in conjunction with the structure of the component capacity topology is responsible for exciting the multiple distinct failure modes observed in Fig. 17(a).

As an example in Fig. 17 consider the interval TF $\in [30, 50]$; mean maximum demand in this interval roughly corresponds to around 15 with components of capacity=



(a) Multiple failures modes



(b) Maximum Demand as a function of TF

Figure 17: Extremal behavior of the CS model for $\lambda = 7$ (Color Online)

$8, 9$ failing. Noting that the queue size is 6, we can understand why components of capacity$= 8, 9$ are being overwhelmed by the mean maximum demand $(9+6 = 15)$ in this interval. However, in addition to the specific sequence and number of extremal demands, relatively stronger neighborhood capacity topologies are partly responsible for the left side of the bell shape and relatively weaker neighborhood capacity topologies are partly responsible for the right side of the bell shape in the interval TF $\in [30, 50]$. For a specific level and sequence of extremal demand, a relatively stronger neighborhood capacity topology provides components a greater opportunity to survive through load sharing. We could construct similar arguments for the other bell curve like waves in Fig. 17 such as the interval TF $\in [50, 70]$ where components of capacity$= (8, 9, 10)$ are failing and mean maximum demand is approximately 16.

4.3 Comparison of the extremal dynamics of the CS model and the Bak-Sneppen evolution model

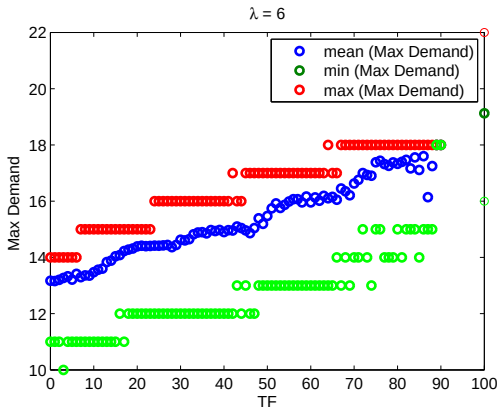
In this section we describe a self-organized critical (SOC) signature of the CS model. We do this by first presenting the SOC signature of the Bak-Sneppen evolution model. We then draw an analogy between the SOC signature of the Bak-Sneppen model and the extremal dynamics of the CS model. Through this exercise we establish the SOC behavior of the CS model.

The Bak-Sneppen evolution model is a self-organized critical (SOC) model discussed in detail in [6, 75]. Here we would like to discuss the similarity between the self-organization of the Bak-Sneppen model to the critical state and the extremal dynamics of the CS model. In the Bak-Sneppen evolution model, independent random numbers f_i are assigned to each site of a d-dimensional lattice of linear size L. The f_i are chosen from a uniform distribution $\mathcal{U}[0, 1]$. At each update step, the site with the smallest random number f_{min} is located. This site is the extremal site. That site and its $2d$ nearest neighbors are then assigned new random numbers drawn from the same uniform distribution, such that the new random numbers are greater than the current

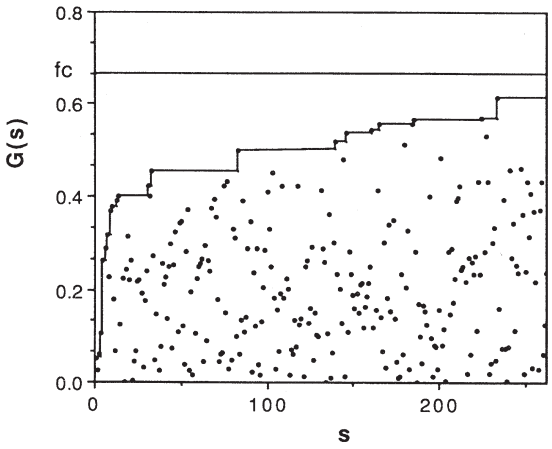
minimum f_{min}, $f_i > f_{min}$. Thereafter in the next update step, the new f_{min} in the system is located. After many update steps the system reaches a statistically stationary state in which the density of random numbers in the system vanishes for $f < f_c$ and is uniform above f_c. In the thermodynamic limit $L \to \infty$, no random number with $f > f_c$ is ever the extremal site [75]. In one dimension, all the random numbers in the system have values above the self-organized threshold $f_c = 0.66702 \pm 0.00003$.

To follow the transient process of the Bak-Sneppen evolution model, [75] tracks the value of the minimal random f_{min} as a function of sequential time s. Fig. 18(b) shows the signal f_{min} as a function of s during the transient process. In the figure, the staircase like solid line is an envelope function that tracks the increasing peaks in f_{min}. The solid staircase line is computed in the following way. At time 0, the first minimum value $f_{min}(0)$ is identified. By definition there are no random numbers smaller than $f_{min}(0)$ in the system. The quantity $f_{min}(0)$ is defined to be the initial gap, G in the distribution of f's, i.e. $G(0) = f_{min}(0)$. After s updates, a larger gap $G(s) > G(0)$ opens up, i.e. the new minimum $f_{min}(s)$ is larger then the previous minimum $f_{min}(s-1)$. The current gap $G(s)$ is the maximum of all the minimum random numbers chosen, $f_{min}(s')$, for all $0 \leq s' \leq s$. In Fig. 18(b) the solid staircase line shows the gap $G(s)$ as a stepwise increasing function of s. As shown in [75], clearly when the gap jumps to a new higher value for f_{min}, there are no sites in the system with random numbers less than the gap. Since the random numbers are chosen from a flat distribution $P(f)$, all the random numbers in the systems are uniformly distributed in the interval $[G(s), 1]$ at the moments in time when the gap jumps. The gap function $G(s)$ is tracking the evolution of the system to its statistically stationary state f_c.

Based on the above SOC signature of the Bak-Sneppen evolution model, we are now in a position to draw an analogy to the CS model extremal dynamics. In



(a) Maximum Demand as a function of TF for $\lambda = 6$



(b) Extremal dynamics of the Bak-Sneppen evolution model reproduced from [75]. Value of the extremal signal f_{min} as a function of s during the transient in a small ($L = 20$) one-dimensional evolution model. The full line shows the gap $G(s)$ as a step-wise increasing function of s. The gap is an envelope function that tracks the peaks in f_{min}.

Figure 18: Comparison of extremal dynamics of CS model with the Bak-Sneppen evolution model

Fig. 18(a) the CS model extremal dynamics for $\lambda = 6$ is shown again. The construction of the figure was explained in detail in chapter II. In Fig. 18(a) as the system organizes into its critical state of $TF = 100$, we observe the staircase like step functions of mean maximum demand (in blue), maximum maximum demand (in red) and minimum maximum demand (in green). These functions are also developing as an envelope function similar to the envelope function $G(s)$ in Fig. 18(b).

Fig. 18(a) demonstrates a clear link between the degree of system self organization, characterized by TF, and the extremal dynamics of maximum demand. For the class of systems we are studying using CS model dynamics for $\lambda = 6$, the path to the SOC state characterized by TF, demonstrates behavior similar to punctuated equilibria. In punctuated equilibrium models such as the Bak-Sneppen evolution model, the system evolves to the critical state, not in a slow and steady path, but in bursts of activity that are separated by long periods of little activity in terms of the evolution of species. In Fig. 18(a), the punctuated equilibriums correspond to the TF values. These TF values are reached through the corresponding extremal dynamics of maximum demand which occurs rarely (with much lower probability) compared to the average demand dynamics. As the systems evolve through the TF values to TF=100, the maximum demand dynamics also evolves as an envelope function similar to the envelope function $G(s)$ in Fig. 18(b). From this discussion we conclude that the class of CS model systems considered, for $\lambda = 6$, also demonstrate SOC signatures. A similar analysis is also possible for the $\lambda = 7$ case.

4.4 Failure progression dynamics in the LOS model

Figure 19 illustrates the profile of "failed cells" versus time for some representative simulations for the LOS model. For these simulations we set the strength of components to the constant $S_{ij} = 6$ and the load L_{ij} is initialized from a Gaussian distribution $\mathcal{N}(3, 0.5)$. For these simulations, we set the Chebyshev-distance parameter

$r = 1$. It can be seen in figure 19, as the simulation progresses, there are breakpoints in the mean slope of the "failed cells" profile. I.e, there exists a critical time when the "rate-of-failure" trajectory switches to an increased slope.

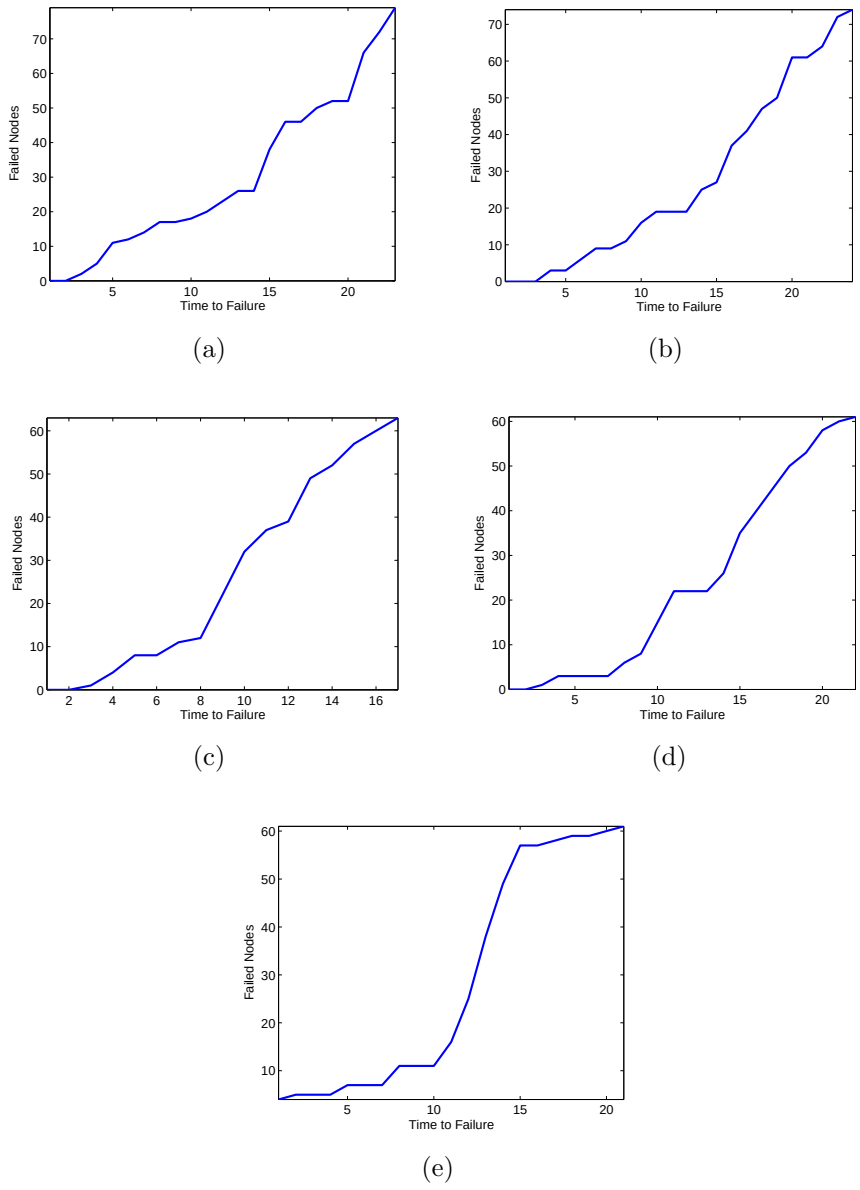
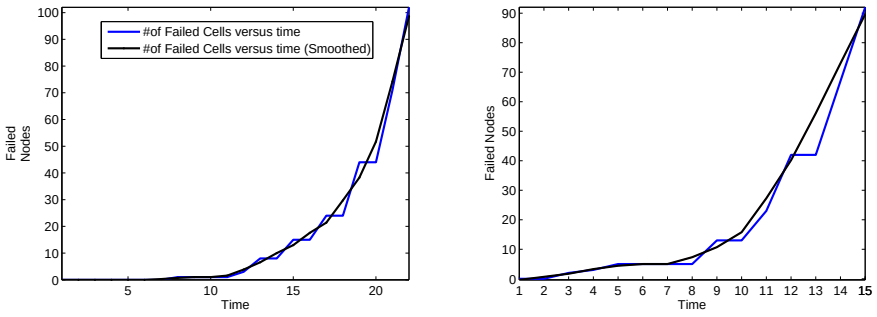


Figure 19: Profile of "Failed cells" versus time

To detect when this switch to an increased slope occurs it is necessary to first smooth the data. Figure 20 illustrates smoothing the "failed cells" time profile using Matlab's curve fitting toolbox. The data was smoothed using Local Weighted Linear

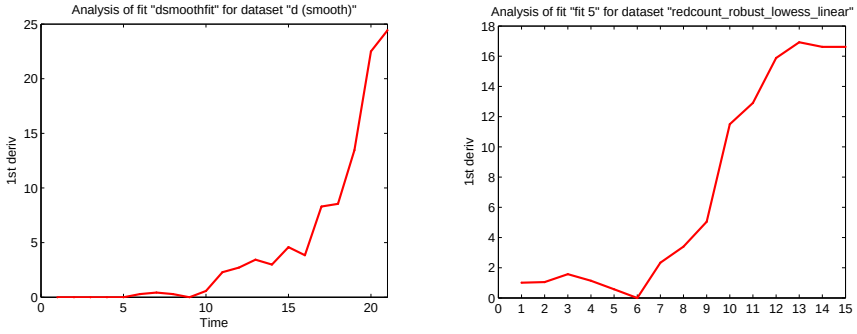
regression techniques.



(a) (b)

Figure 20: Smoothing via "Local weighted linear regression".

After smoothing the data, the data is fit using Linear interpolation and then the first derivative is approximated. The first derivative approximation is illustrated in figure 21.



(a) (b)

Figure 21: Fitting using "linear interpolation" and then taking a 1st derivative

4.4.1 Change-point detection

Detecting time points at which the properties of time-series data change are referred to as change-point detection problems. Change-point detection has been widely researched in areas such as Signal processing, Wavelet analysis, Graphics and Financial time series research etc. These techniques also go by other names such as: Step

detection, Jump detection, Edge detection etc. In our case, the intent is to detect the "critical time" when the mean slope of the "failed cells" profile transitions to an increased value. We would like to detect the change-point when the system dynamics switch to a accelerated rate-of-failure.

A change-point detection algorithm based on the Gradient-Threshold method was implemented. Using first derivative information it is possible to estimate the "critical time" when the mean slope of the "failed cells" switches. Figure 22 shows the distributions for 100k Monte-Carlo runs after the implementation of the change-point detection algorithm. The change-point distribution is plotted (in blue) to the left of the time-to-failure distribution (plotted in black). The mean-to-mean difference between the time-to-failure distribution and the change-point time distribution is 5-6 time steps.

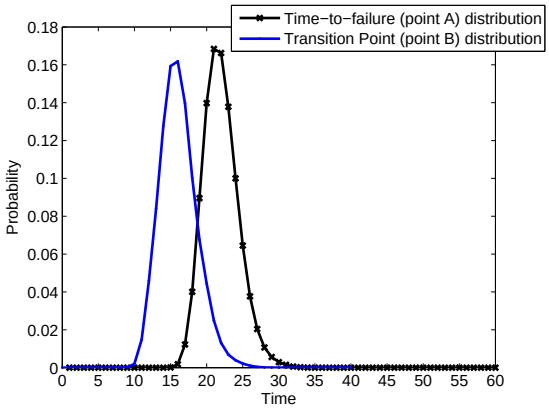
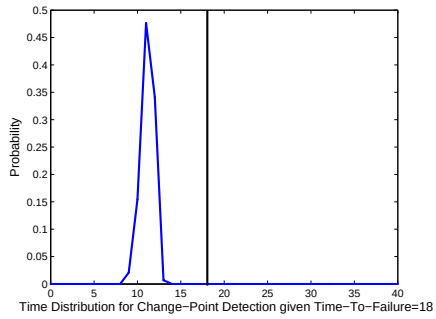


Figure 22: Distributions for time-to-failure (in black) and change-point (in blue)

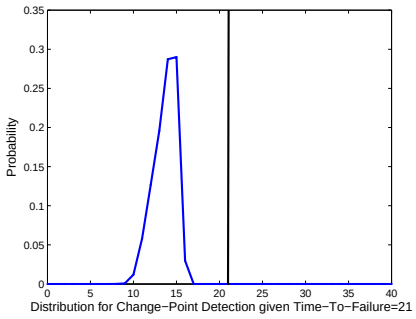
4.4.2 Change-point conditional distributions

Also plotted in figure 23 is the "change-point conditional distributions". These plots are generated in the following way. First, from the simulation data all failures at a particular time step (for example: system failures at time step 18) are extracted. Then change-point distribution for this extracted data set is plotted. Figure 23 shows

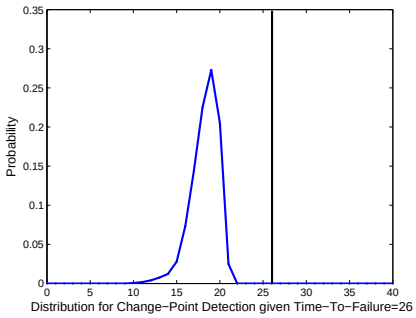
this "conditional change-point distributions" for time-to-failures at time steps: 18, 21, 26 and 28.



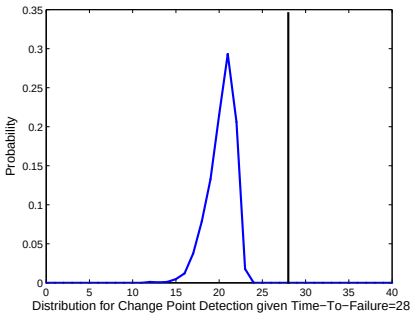
(a) Distribution for Change-Point detection conditioned on Time-to-Failure=18



(b) Distribution for Change-Point detection conditioned on Time-to-Failure=21



(c) Distribution for Change-Point detection conditioned on Time-to-Failure=26



(d) Distribution for Change-Point detection conditioned on Time-to-Failure=28

Figure 23: Conditional distributions for change-point detection.

Also it is informative to view the time evolution of the number of failures for a specific case in a table. Referring again to figures 20(a) and 21(a), we note the time-to-failure is at time unit 22. The change-point is detected at time unit 15. Thus the change-point is detected 7 time units before the actual system failure. Table 3 tabulates the number of failed cells at these time points. At time unit 15, the number of failed cells is 15. At time unit 21 and 22, the number of failed cells is 74 and 103 respectively. The failure criteria is 75 failed cells.

Table 3: # of failed cells at time 15, 21 and 22

Time unit	#of Failed Cells	
15	15	Change-point detected
21	74	
22	103	System failure detected

CHAPTER V

APPLICATIONS

In this chapter we will discuss the relationship between the extremal models presented in the previous chapters to real world applications in creep-rupture studies, air-traffic management and road traffic congestion.

5.1 Creep-rupture studies using the LOS model

The failure of a component after long-term exposure under constant load is called stress-rupture or creep-rupture. In multi-phase materials, such as fiber-matrix composites, the failure process has several aspects. Under high levels of steady stress, such materials demonstrate time-dependent mechanical degradation. Randomly distributed local flaws which grow stochastically in time eventually lead to microcracks in the fiber-matrix composite. In turn, the local stress loss in these microcracks lead to load redistribution to neighboring fiber elements. This results in accelerating the flaw growth in the neighbors and thus causing microcracks in the neighbors. Ultimately several such microcracks join together forming a catastrophic crack. In this chapter we will see that the statistical nature of the failure process and choice of fiber-matrix composite parameters lead to not only highly variable composite lifetime distributions but also to different failure modes.

5.1.1 Creep-rupture model of Mahesh/Phoenix

Mahesh and Phoenix introduced a creep-rupture model for fiber-matrix composites in [62]. As described in [62] the statistics of fiber failure are governed by the breakdown model of Coleman [30] which embodies a Weibull hazard functional of fiber load history imparting power-law sensitivity to fiber load with exponent ρ, and Weibull

lifetime characteristics with shape parameter β. Fiber load redistribution at breaks are calculated using a "shear-lag" mechanics model which [62] has found to be more realistic than idealized load sharing rules such as equal, global or local load sharing. The study in [62] is concerned with the "avalanche" regime of failure as discussed by Curtin and Scher in [34] where the composite lifetime follows weakest-link scaling. The "avalanche" regime of failure will also be our focus for the LOS model. We shall compare the failure mechanisms of the LOS model and the Mahesh/Phoenix model in this "avalanche" regime. The study by Mahesh/Phoenix reveals two distinct failure modes within the avalanche regime. For small ρ, fiber failure is is not very sensitive to load level and the fiber-matrix composite demonstrates 'tough' behavior. In the 'tough' regime, random fiber failures cause progressive distributed damage until a critical volume fails along with its catastrophic extension. For large ρ, fiber failure is very sensitive to load level and the fiber-matrix composite fails in a 'brittle' manner. In the 'brittle' regime, there is a gradual growth of a cluster of mostly contiguous fiber breaks, these then abruptly transitions into a catastrophic crack. As we shall see subsequently, depending on the chosen model parameters, the LOS model demonstrates both 'tough' and 'brittle' failure modes.

5.1.2 Different rules for load sharing

The model by Mahesh and Phoenix [62] adopts a "shear-lag" mechanics model for load sharing in fiber-matrix composites. The authors have found that the shear-lag mechanics model is more realistic than idealized rules for load sharing such as equal, global or local load-sharing.

Equal load sharing rules were the first to be considered in the literature for the fiber-bundle lifetime models in [27, 28, 29, 30, 80]. In equal load-sharing when a fiber fails its load is redistributed equally to the surviving fibers in the bundle.

The concept of equal load-sharing was extended by Ibnabdjalil and Phoenix in

[48] to incorporate the idea of global load-sharing. Global load-sharing is a more continuous version of equal load-sharing in which the load of a failed fiber is divided and transferred equally onto a fairly large number of surviving fibers which are determined over a certain characteristic distance from the failed fiber.

In addition to equal and global load-sharing rules there are also local load-sharing rules. Here the load of a failed fiber is transferred laterally to its nearest neighbors and fibers further away feel little overload. Tierney in [95] and Phoenix and Tierney in [81] have considered local load-sharing rules in their fiber-matrix composite lifetime model.

5.1.3 Load sharing based on the Chebyshev-distance parameter

Before presenting the failure mechanism relationships between the LOS model and the creep-rupture model of Mahesh/Phoenix, we discuss the load redistribution mechanism in the LOS model which is controlled by the Chebyshev distance parameter r.

When a node fails, the corresponding load is transferred to the adjacent nodes. The load is redistributed to the neighboring nodes according to the Chebyshev distance r. For the LOS model the Chebyshev-distance parameter r is set to 1 by default. This means only nodes immediately neighboring the failed node will be considered for load redistribution. The Chebyshev-distance parameter can be modified to values other than 1. For example, if $r = 2$ then only neighboring nodes up to a Euclidean distance of 2 from the failed node is considered for load redistribution. The effects of varying the parameter r will be investigated in this section.

Fig. 24 illustrates how Chebyshev distances are considered. In Fig. 24, the red node at the center has failed at the current time step and the load is redistributed to the neighboring nodes. The nodes immediately bordering the red node, labeled '1', are at a Chebyshev distance $r = 1$. Also labeled in the figure are nodes that

are at Chebyshev distances $r = 2$ and $r = 3$ from the red node. Nodes at the same distance from a failing node are said to be at the same Chebyshev level. Hence, Fig. 24 illustrates 3 different Chebyshev levels.

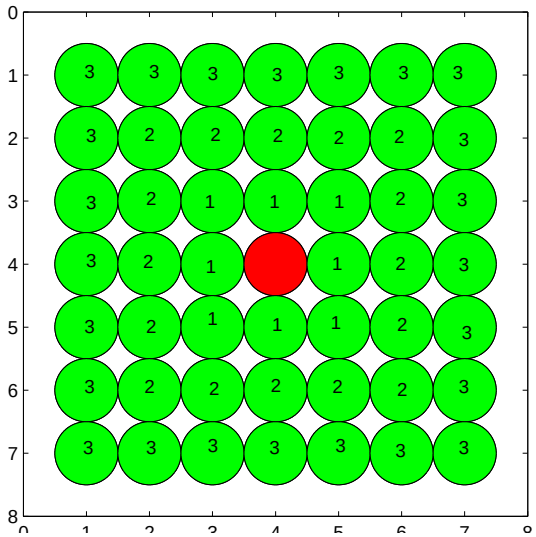


Figure 24: Illustration of Chebyshev distances of the green nodes from the red node.

Next we discuss the exact mechanism of load redistribution. The load can be redistributed equally among nodes or following different weighting schemes. These weighting schemes can be thought of as load redistribution policies. Three different weighting schemes are considered for redistributing the load of the failed node to the neighboring nodes. The loads from failed nodes are not redistributed to failed neighboring nodes. Also the different load redistribution schemes do not take into account the local existing loads at neighboring nodes that are fully operational or stressed.

In weighting scheme #1, the load is redistributed equally to all fully operational or stressed nodes regardless of the Chebyshev distance from the failing node. For $r = 1$, this scheme would correspond to local-load sharing. On the other hand for $r > 1$, this would correspond approximately to equal load-sharing.

Table 4: Load redistribution weighting scheme #3 for the different Chebyshev levels

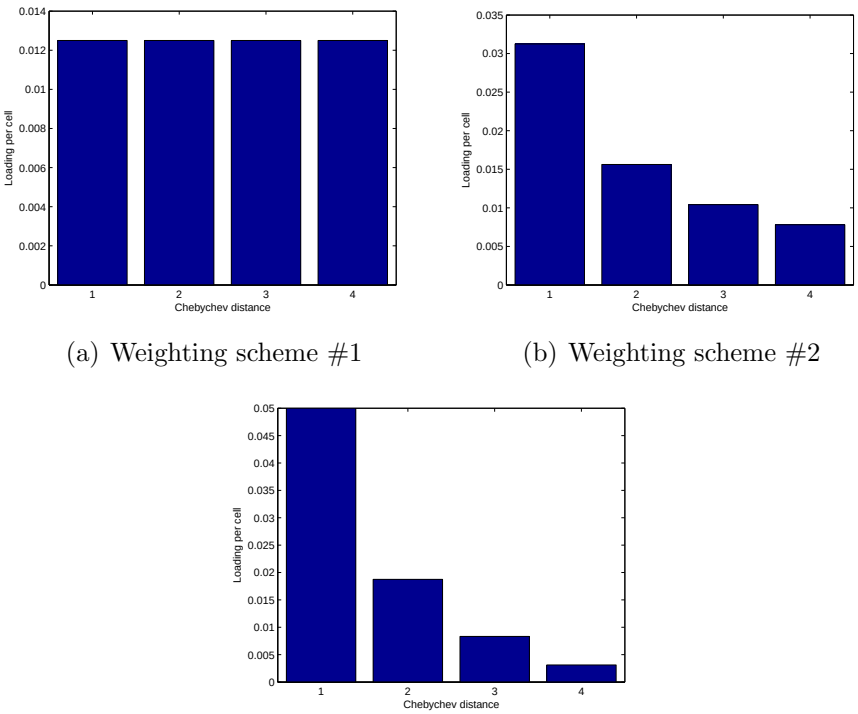
Chebyshev distance (r)	Level 1	Level 2	Level 3	Level 4	Level 5
1	1				
2	0.6	0.4			
3	0.5	0.3	0.2		
4	0.4	0.3	0.2	0.1	
5	0.4	0.25	0.15	0.1	0.1

In weighting scheme #2, the load is first distributed equally to the different Chebyshev levels, then the distributed load is divided equally among the "not failed" nodes existing at that particular Chebyshev level. For $r > 1$ this scheme would correspond approximately to variants of global load-sharing.

In weighting scheme #3, each Chebyshev level is assigned a specific weight w_i and the load is assigned to each level according to those weights. Note, $\sum_{i=1}^{n} w_i = 1$, where n is the number of Chebyshev levels. The "not failed" nodes at a particular level then divide the load equally among themselves. Typically Chebyshev levels at a greater distance are assigned a lower weight or proportion of the load while Chebyshev levels closer to the failed nodes are assigned a higher proportion of the load. For $r > 1$ this scheme would correspond approximately to variants of global load-sharing.

Fig. 25 illustrates these three different weighting scheme on a per node basis for the case when the number of Chebyshev levels $r = 4$. Table 4 tabulates one possible option for weighting scheme #3 for different Chebyshev levels $r = 1..5$.

Depending on the system size n and how the system is decomposed (e.g the size of the fiber bundles in the fiber-matrix composite model), using the Chebyshev distance r and the different weighting schemes, it is possible to model the different load sharing rules. For example with $r = 1$ and weighting scheme #1 it is possible to model local load-sharing. On the hand with $r = 4$ and weighting scheme #3 it is possible to model a variant of global load-sharing. With $r = 4$ and weighting scheme #1 one could tentatively model equal load-sharing.



(a) Weighting scheme #1

(b) Weighting scheme #2

(c) Weighting scheme #3

Figure 25: Different Load redistribution weighting schemes on a per node basis for Chebyshev distance $r = 4$.

5.1.4 Comparison between the LOS model and the Mahesh/Phoenix model

To compare the failure mechanisms of the LOS model with the failure mechanisms demonstrated by the Mahesh/Phoenix creep-rupture model, we will initialize the LOS model parameters such that we obtain similar failure patterns as obtained by the Mahesh/Phoenix model.

For LOS model we have the following parameters that we need to tune appropriately. The parameter $\eta \in (0, 1)$ is the strength degradation parameter. η controls the threshold when loss of component strength initiates. For low values of η, for example $\eta = 0.3$, components will be overcome by lower amounts of load but components will spend greater lengths of time in LOS dynamics. For high values of η, for example

$\eta = 0.8$, higher amounts of load will be needed to overcome components but components will spend comparatively less time in LOS dynamics. In general, the parameter η controls the level of load required to initiate LOS dynamics and also the time a component engages in LOS dynamics. At this point we could postulate that higher values of η might lead to 'brittle' failure since failure will take place abruptly with little precursory activity in the form of LOS dynamics. On the other hand, low values of η might lead to ''tough' behavior as component failure will occur after multiple components have undergone considerable LOS dynamics.

The second parameter we will tune is the Chebyshev-distance parameter r. For the simulations to be presented we have used weighting scheme #3 as shown in Table 4. In the simulations we consider Chebyshev distances $r = 1, 3, 5$. For $r = 1$, we essentially model local-load sharing. For $r = 3, 5$ we are essentially modeling global load-sharing which is a more refined and continuous version of equal load-sharing as discussed previously.

Since we would like to model fiber-matrix composite creep-rupture with the LOS model we need to set the component strength appropriately. In strength, fibers, strands and pressure vessels tend to follow a Weibull distribution [81]. Experimental values for the Weibull shape parameter range from 5 to 9 for fibers and from 25 to 35 for strands and pressure vessels. In failure time, strands and pressure vessels also appear to follow a Weibull distribution [96]. Based on this information, in the simulations, we initialize the LOS model component strength by sampling from a Weibull distribution with scale parameter 9 and shape parameter 25. Consequently, we are attempting to mimic the strength behavior of pressure vessels with the LOS model. In conjunction with the parameters η and r, we will next discuss the similarity of the failure mechanisms of the LOS model and the Mahesh/Phoenix model.

5.1.5 Simulation setup

For each setting of the parameters η and r we run 30,000 monte-carlo simulations. We set the system size to $n = 10$. The system strength and r is initialized as described in the previous section. For the parameter η, we investigate the values 0.5, 0.7 and 0.9. With $\eta = 0.5$, components will spend greater amounts of time in LOS dynamics compared $\eta = 0.9$. However, the load required to initiate LOS dynamics will greater for $\eta = 0.9$ compared to $\eta = 0.5$.

Following Mahesh and Phoenix, we are particularly interested in analyzing the 'avalanche' regime of failure which is characterized by components failing in a chain or cascade one after the other. In the 'avalanche' regime, we capture the failure time for the complete system or 100 components and also the failure time for the first 20 components to fail. The 'avalanche' regime is reached by identifying the loads that causes the entire system to collapse for a given setting of system strength, η and r. In particular, we are interested in establishing whether we can capture tough and brittle failure modes in the 'avalanche' regime and also whether the failure time distributions agree with those in the literature.

5.1.6 $\eta = 0.5, 0.7$ with Chebyshev distance $r = 1$

5.1.6.1 Tough failure regime

First, we set $\eta = 0.5$ and run 30,000 simulations for each value of $r = 1, 3, 5$. For each value of r, we identify the minimum load required for complete system failure. By increasing the load above this minimum load, we identify the different failure modes. We then repeat this process for $\eta = 0.7$.

The Chebyshev distance setting $r = 1$ corresponds to local load-sharing. For $r = 1$ and $\eta = 0.5$, we observe complete system failure for the minimum load $L = 3.75$. The corresponding failure time distributions are shown in Fig. 26. Similarly for $r = 1$ and $\eta = 0.7$, we observe complete system failure for minimum load $L = 4.5$. The

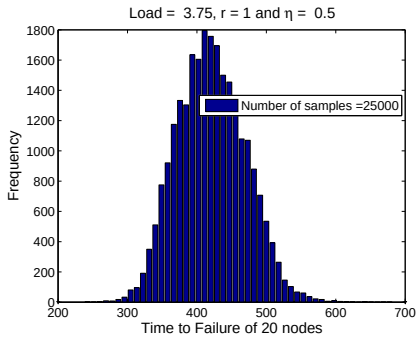
corresponding failure time distributions are shown in Fig. 27. In both cases, the failure time distributions follow a Log-normal distribution. This result is in agreement with Mahesh and Phoenix where they demonstrate for small ρ, the composite failure time follows a Log-normal distribution. They have identified this failure mode as 'tough' behavior. In this regime, fiber failure is is not very sensitive to load level. In the 'tough' regime, random fiber failures cause progressive distributed damage until a critical volume fails along with its catastrophic extension.

In the 'avalanche' regime, the small ρ parameter of Mahesh/Phoenix corresponds to our low values of load $L = 3.75$ for $\eta = 0.5$ and $L = 4.5$ for $\eta = 0.7$. These loads are high enough to initiate LOS dynamics in some components in the system. When these components fail, their load is redistributed to neighboring components which then initiate LOS dynamics. This cycle of LOS dynamics and load redistribution keeps on repeating in cascade from one component to the next until the entire system fails. Thus for our system, the value of load $L = 3.75$ for $\eta = 0.5$ and $L = 4.5$ for $\eta = 0.7$ corresponds to the 'tough' failure regime.

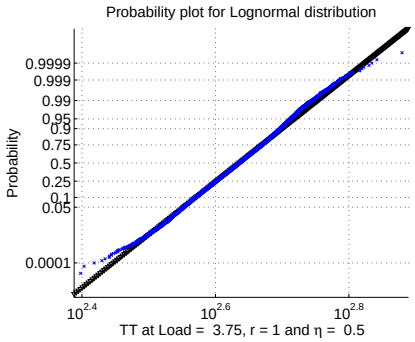
5.1.6.2 Brittle failure regime

As the loads are iteratively increased we observe the failure mechanism transition from the tough failure mode to a brittle failure mode. At loads greater than $L = 5.5$ for $\eta = 0.5$ we observe that the failure time distribution follows a Weibull distribution. This is shown in Fig. 28. Also at loads greater than $L = 6$ for $\eta = 0.7$ we observe that the failure time distribution follows a Weibull distribution. This is shown in Fig. 29.

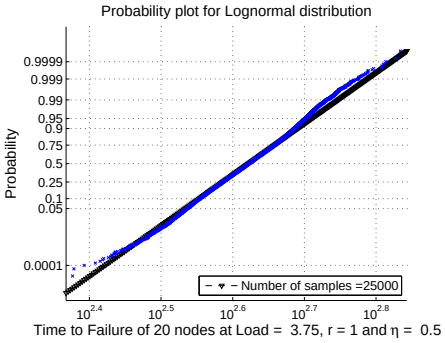
At these load setting the load is high enough that components fail with minimum LOS dynamics. Once a component fails its load redistribution is sufficient enough to cause neighboring components to fail and redistribute their load thus triggering a cascade of failures. The entire system abruptly collapses in a 'brittle' manner. Load settings greater than $L = 5.5$ for $\eta = 0.5$ and $L = 6$ for $\eta = 0.7$ correspond to the



(a) $L = 3.75$

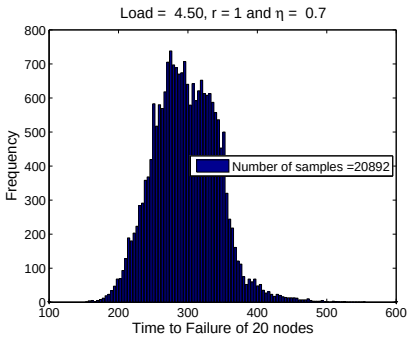


(b) $L = 3.75$

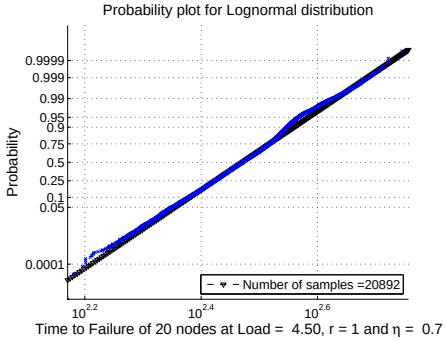


(c) $L = 3.75$

Figure 26: Log-Normal fit. r=1, $\eta = 0.5$



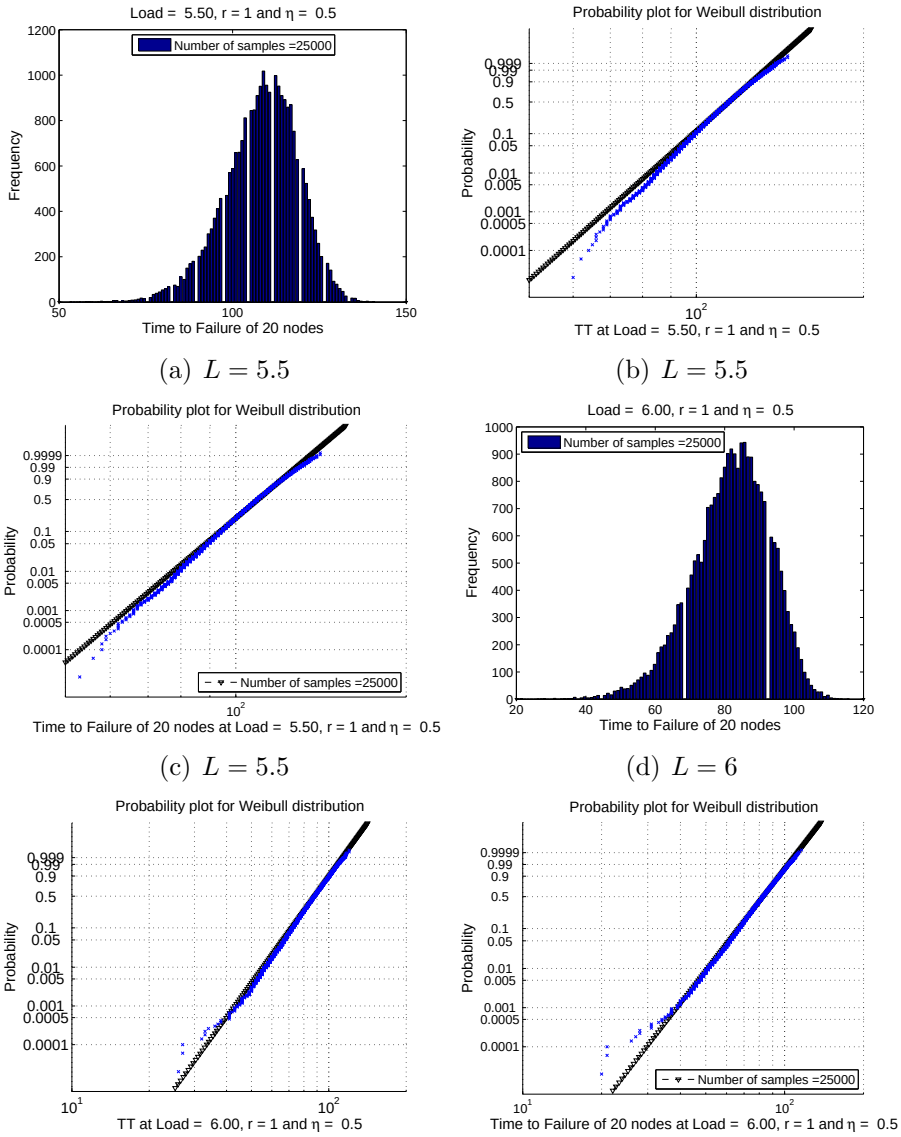
(a) $L = 4.5$



(b) $L = 4.5$

Figure 27: Log-Normal fit. r=1, $\eta = 0.7$

large ρ settings of Mahesh and Phoenix. They have also found that for large ρ, fiber-matrix composite failure time follows a Weibull distribution. For large ρ, the fiber is very sensitive to load level and the fiber-matrix composite fails abruptly.

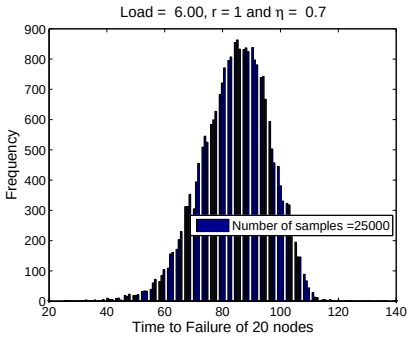


(a) $L = 5.5$ (b) $L = 5.5$

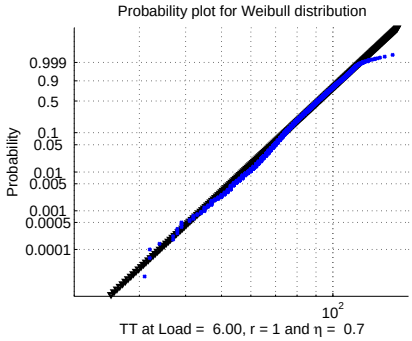
(c) $L = 5.5$ (d) $L = 6$

(e) $L = 6$ (f) $L = 6$

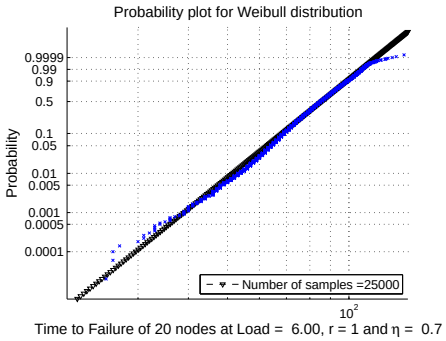
Figure 28: Weibull fit. r=1, $\eta = 0.5$



(a) $L = 6$



(b) $L = 6$



(c) $L = 6$

Figure 29: Weibull fit. r=1, $\eta = 0.7$

5.1.7 $\eta = 0.5, 0.7$ with Chebyshev distance $r = 3$ and $r = 5$

5.1.7.1 Tough failure regime

Chebyshev distance $r = 3$ and $r = 5$ with weighting scheme #3 corresponds to a form of global load-sharing as described in Ibnabdjalil and Phoenix [48]. Global load-sharing is a more continuous from of equal load-sharing. For load settings in the 'avalanche' regime we can identify two different failure modes for the parameter settings $\eta = 0.5, 0.7$ for $r = 3, 5$.

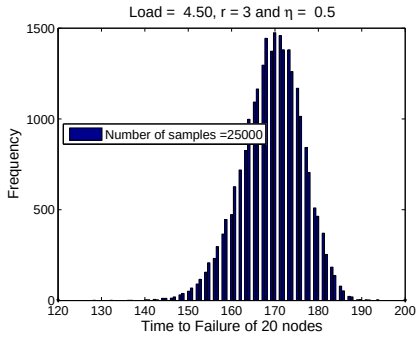
For both $\eta = 0.5, 0.7$, at low load settings the failure time distribution follows a normal distribution for both $r = 3, 5$. For $\eta = 0.5$, this is shown in Fig. 30 for $r = 3$ and Fig. 31 for $r = 5$. Similarly for $\eta = 0.7$, the normal distribution failure times are shown in Fig. 32 for $r = 3$ and Fig. 33 for $r = 5$. This is in agreement with [80], where the author established that the lifetime of a equal load-sharing fiber bundle is asymptotically normally distributed. This result makes intuitive sense since as the Chebyshev-distance for load redistribution is increased to $r = 3$ or $r = 5$, the components in the system are provided greater opportunities to survive through load sharing. Thus, even though the system does ultimately collapse at these low load settings, the failure mode is more dispersed and spread over greater lengths of time. And the failure time distribution is under lied by the normal distribution.

The minimum load settings exciting the avalanche failures observed for the distributions in Figs. 30, 31, 32, 33 correspond to the tough or dispersed failure regime.

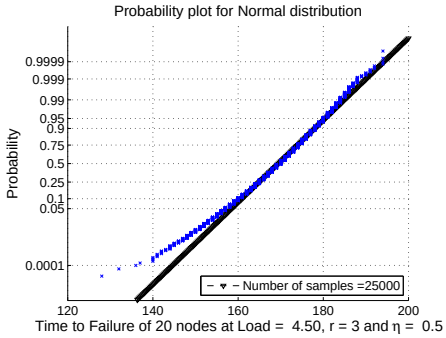
5.1.7.2 Brittle failure regime

Next, by iteratively increasing the load we arrive at load settings that cause the system to transition from the tough failure regime to the brittle failure regime for $\eta = 0.5, 0.7$ at $r = 3, 5$.

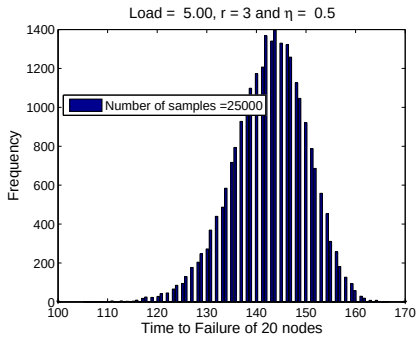
For both $\eta = 0.5, 0.7$, at high load settings the failure time distribution follows a



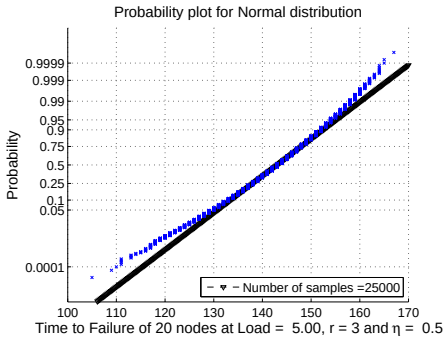
(a) $L = 4.5$



(b) $L = 4.5$

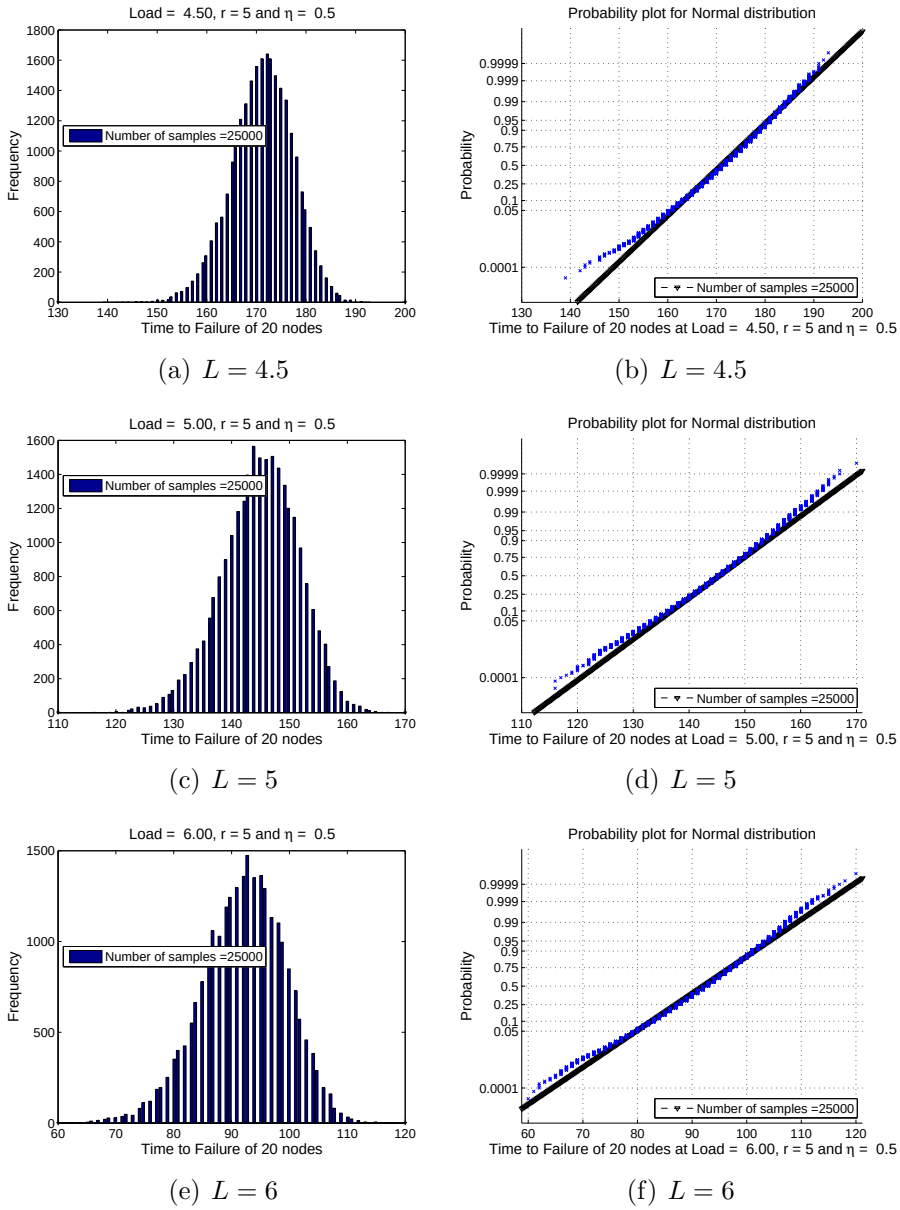


(c) $L = 5$



(d) $L = 5$

Figure 30: Normal fit. r=3, $\eta = 0.5$

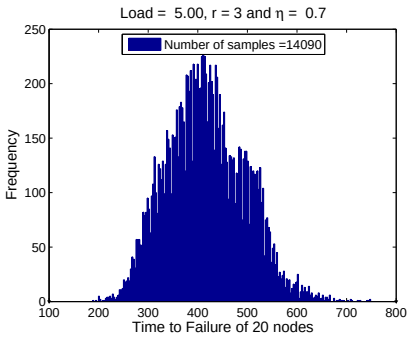


(a) $L = 4.5$ (b) $L = 4.5$

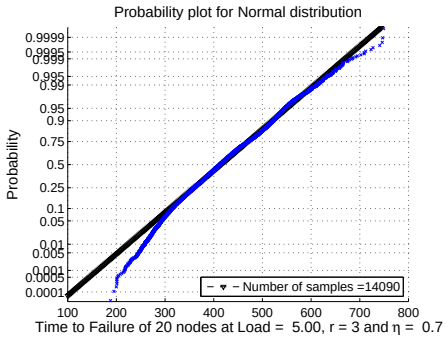
(c) $L = 5$ (d) $L = 5$

(e) $L = 6$ (f) $L = 6$

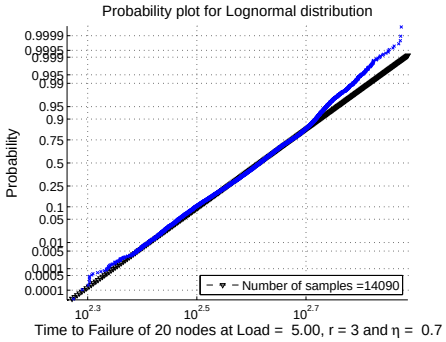
Figure 31: Normal fit. r=5, $\eta = 0.5$



(a) $L = 5$

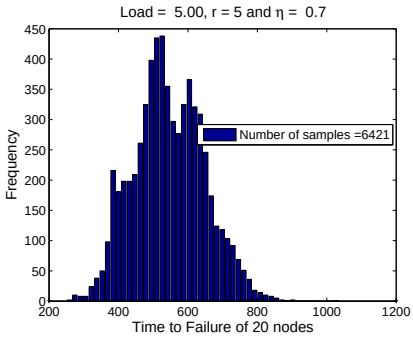


(b) $L = 5$

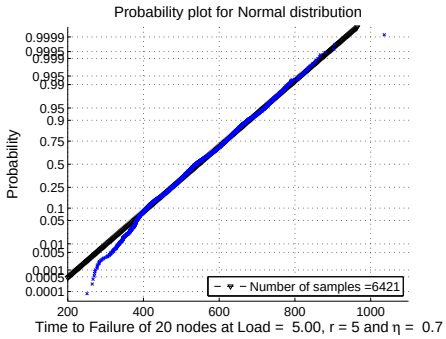


(c) $L = 5$

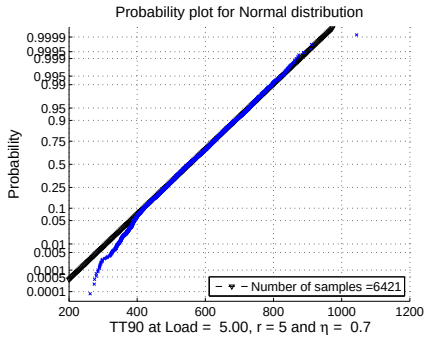
Figure 32: Normal fit. r=3, $\eta = 0.7$



(a) $L = 5$

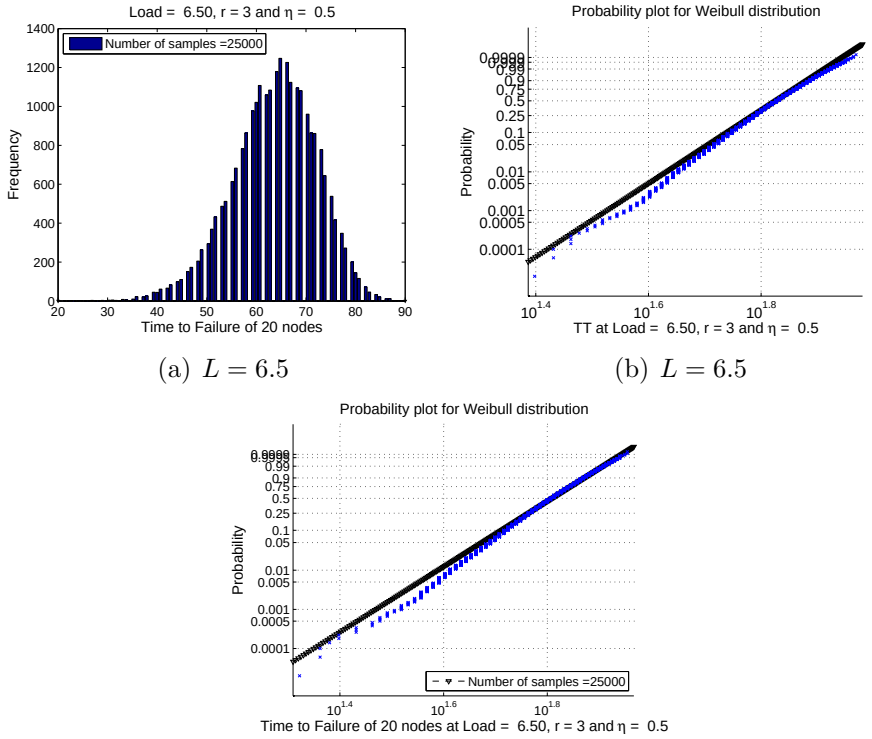


(b) $L = 5$



(c) $L = 5$

Figure 33: Normal fit. r=5, $\eta = 0.7$



(a) $L = 6.5$ (b) $L = 6.5$

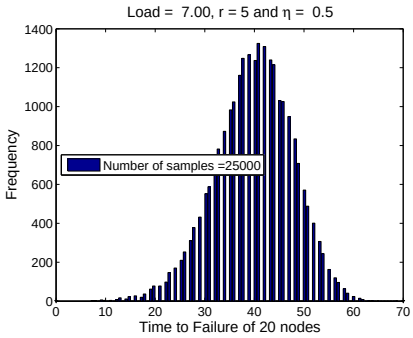
(c) $L = 6.5$

Figure 34: Weibull fit. r=3, $\eta = 0.5$

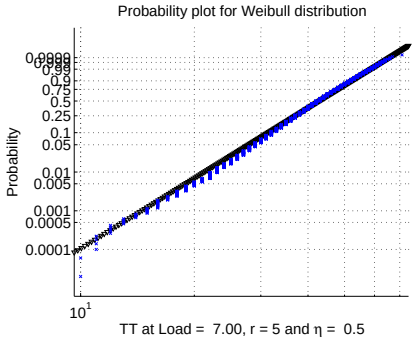
Weibull distribution for both $r = 3, 5$. For $\eta = 0.5$, this is shown in Fig. 34 for $r = 3$ and Fig. 35 for $r = 5$. Similarly for $\eta = 0.7$, the Weibull distribution failure times are shown in Fig. 36 for $r = 3$ and Fig. 37 for $r = 5$. This results makes intuitive sense because as the load is increased the, the advantage gained by redistributing the load over increased Chebyshev distances is diminished. So even though $r = 3$ or $r = 5$, for high load settings the system fails in a brittle manner. Correspondingly the high load settings for the distributions in Figs. 34, 35, 36, 37 correspond to the brittle failure regime.

5.1.8 $\eta = 0.9$ with Chebyshev distance $r = 1$, $r = 3$ and $r = 5$

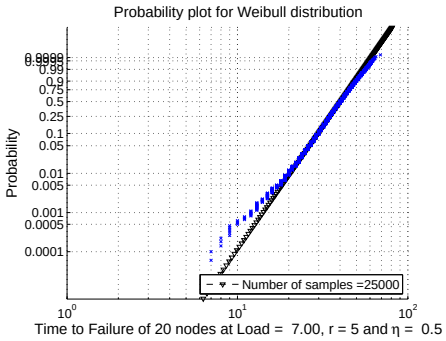
The failure behavior of the system at $\eta = 0.9$ is very interesting since for this parameter value the system engages in minimal LOS dynamics. At this setting, large values of load are required to initiate LOS dynamics and then components spend



(a) $L = 7$

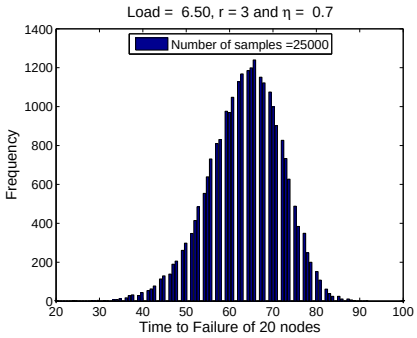


(b) $L = 7$

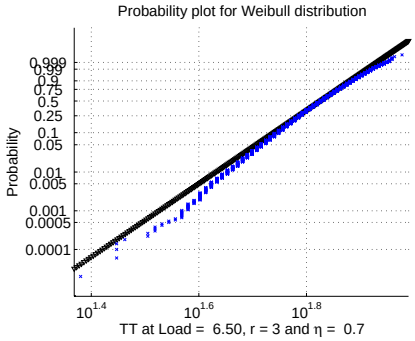


(c) $L = 7$

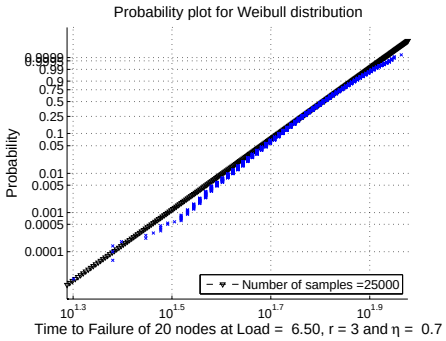
Figure 35: Weibull fit. r=5, $\eta = 0.5$



(a) $L = 6.5$

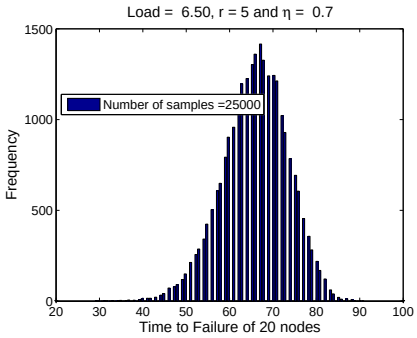


(b) $L = 6.5$

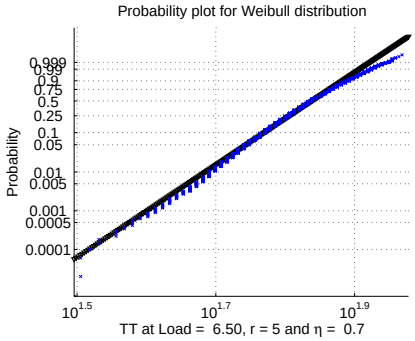


(c) $L = 6.5$

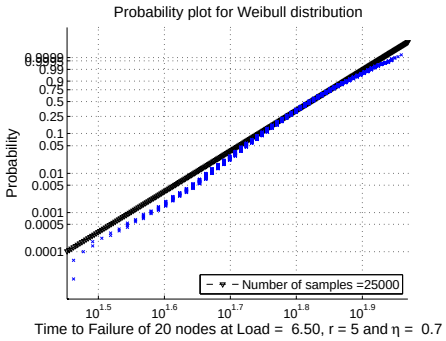
Figure 36: Weibull fit. r=3, $\eta = 0.7$



(a) $L = 6.5$

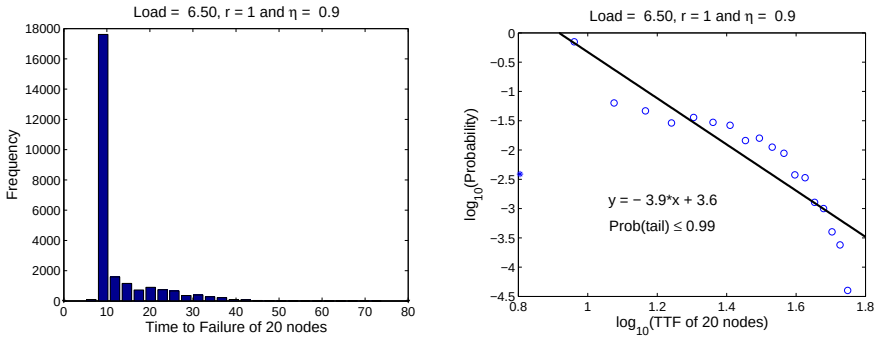


(b) $L = 6.5$

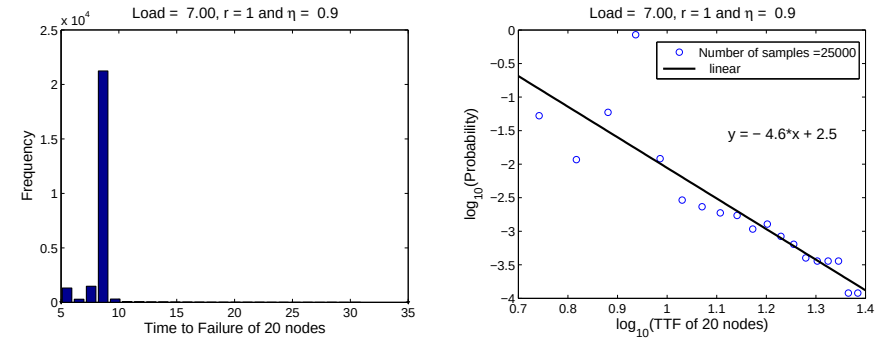


(c) $L = 6.5$

Figure 37: Weibull fit. r=5, $\eta = 0.7$



(a) $L = 6.5$ (b) $L = 6.5$



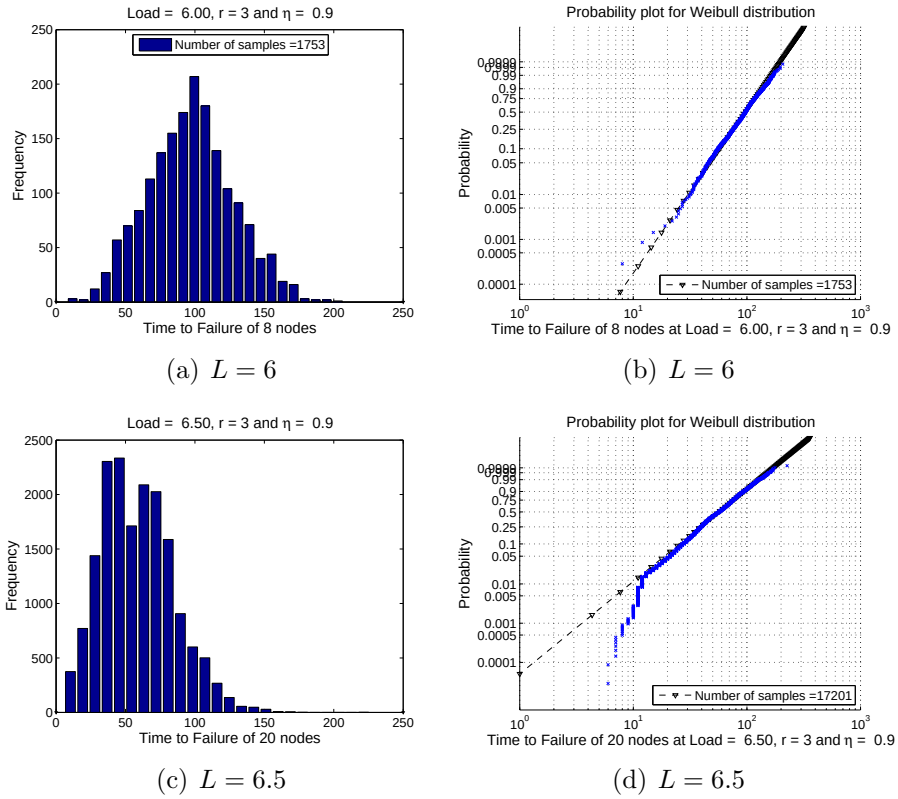
(c) $L = 7$ (d) $L = 7$

Figure 38: Power Law fit. r=1, $\eta = 0.9$

minimum time in the loss of strength process. Consequently the load redistribution is sufficient to trigger failure in neighboring components which in turn trigger chains of load redistribution and failure. Through out this process component time spent in LOS is minimum. Also at $\eta = 0.9$, the effects of the chebyshev parameter r is diminished. For all values of r, we observe the system collapsing through a power-law distribution.

5.1.8.1 Chebyshev distance $r = 1$

For $r = 1$, the power-law distribution for system failure time is shown in Fig. 38. The power-law distribution is observed for load values greater than $L = 6.5$. Below $L = 6.5$, the system is resilient and cascading failure is not triggered. This is due to the high threshold value of $\eta = 0.9$.



(a) $L = 6$ (b) $L = 6$

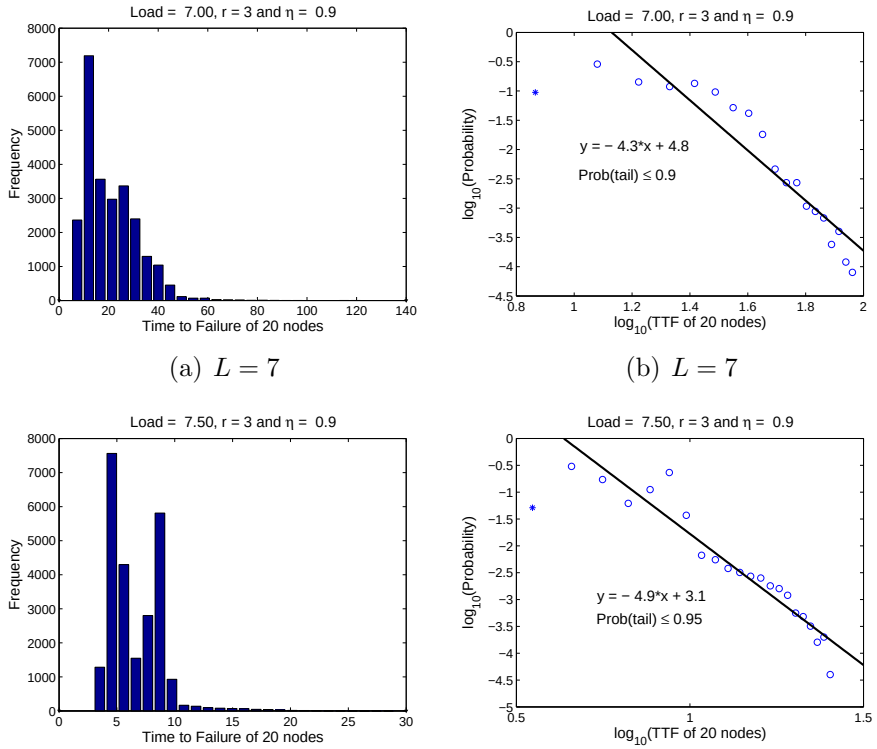
(c) $L = 6.5$ (d) $L = 6.5$

Figure 39: Weibull fit. r=3, $\eta = 0.9$

5.1.8.2 Chebyshev distance $r = 3$

For $r = 3$, we do observe a transition in failure time distribution. From a Weibull distribution for $L = 6$ to a power-law distribution for $L = 7$. The Weibull distribution for failure time is shown in Fig. 39 and the power-law distribution is shown in Fig. 40. However, even though we have different distributions for the failure time at these loads, the system is still collapsing in the brittle failure regime. This is due to the high threshold value of $\eta = 0.9$ which is resulting in little LOS dynamics taking place. However failure is taking place mainly due to cascading chains of load redistributions from one component to the next which is causing the entire system to collapse abruptly in a brittle manner.

Again due to the high threshold value $\eta = 0.9$, the system is resilient and cascading failure is not triggered for load values lower than $L = 6$ for $r = 3$.



(a) $L = 7$ (b) $L = 7$

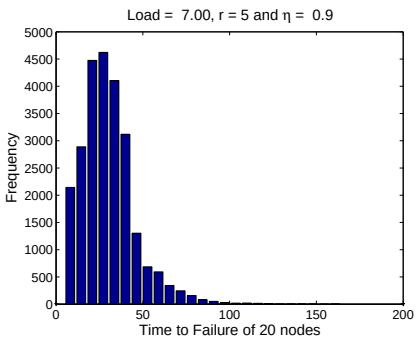
(c) $L = 7.5$ (d) $L = 7.5$

Figure 40: Power Law fit. r=3, $\eta = 0.9$

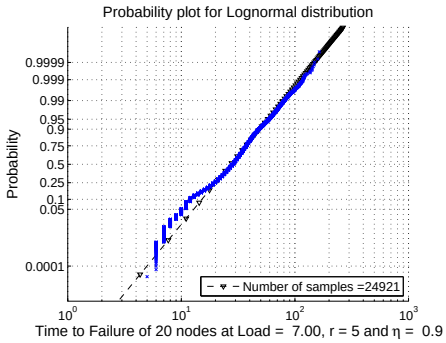
5.1.8.3 Chebyshev distance $r = 5$

Also for $r = 5$, we observe a transition in failure time distribution. From a Log-normal distribution for $L = 7$ to a power-law distribution for $L = 7.5$. The Log-normal distribution for failure time is shown in Fig. 41 and the power-law distribution is shown in Fig. 42. However, even though we have different distributions for the failure time at these loads, the system is still collapsing in the brittle failure regime. This is due to the high threshold value of $\eta = 0.9$ which is resulting in little LOS dynamics taking place. However failure is taking place mainly due to cascading chains of load redistributions from one component to the next which is causing the entire system to collapse abruptly in a brittle manner.

Again due to the high threshold value $\eta = 0.9$, the system is resilient and cascading failure is not triggered for load values lower than $L = 7$ for $r = 5$. We also note the load required for the 'avalanche' failure for $r = 5$ is slightly higher than the loads for $r = 1, 3$. This indicates there is some sensitivity to the load redistribution chebyshev-distance. The system is slightly more resilient when $r = 5$ compared to when $r = 1, 3$.

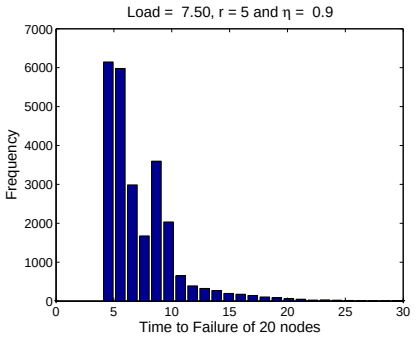


(a) $L = 7$

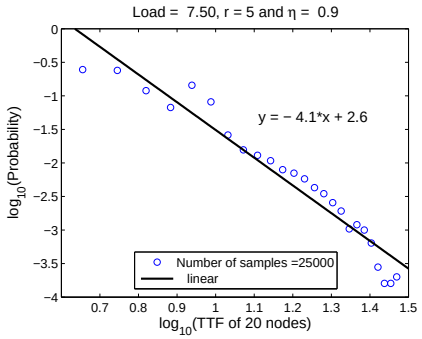


(b) $L = 7$

Figure 41: Log-Normal fit. r=5, $\eta = 0.9$



(a) $L = 7.5$



(b) $L = 7.5$

Figure 42: Power Law fit. r=5, $\eta = 0.9$

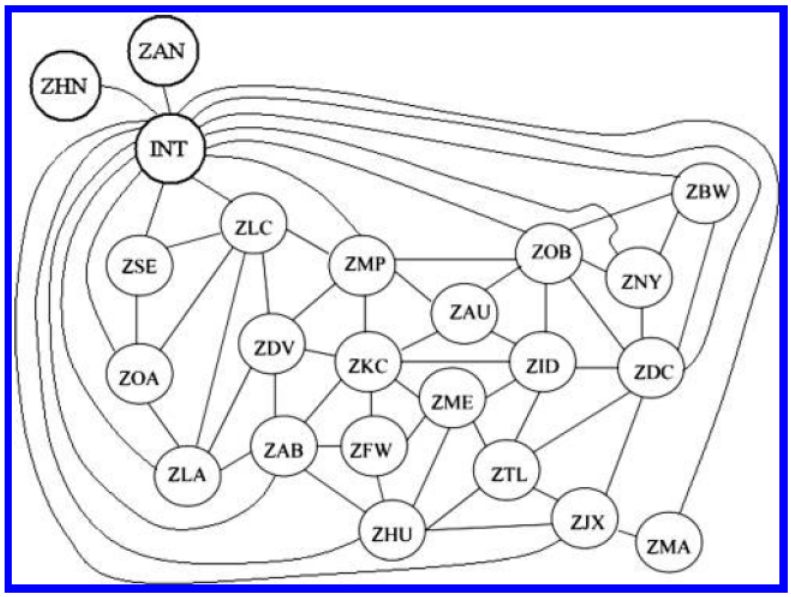
5.2 Air traffic management using the CS model

In this section we discuss the relationship between the CS model and Aggregate flow models described in air traffic management literature [86, 92, 93]. We will detail how the aggregate flow models can be reformulated in a form that completely matches the CS model description.

5.2.1 Aggregate flow models for air traffic management

We refer the reader to [86, 93] for a detailed description on Aggregate flow models for air traffic management. Here we only provide the details necessary to demonstrate the method of reformulating the Aggregate flow model to match the CS model specification.

(a) Continental United State airspace



(b) Center network model

Figure 43: Continental United States airspace center model. Reproduced from [93].

The airspace in the continental United States is divided into 20 centers. This is shown in Fig. 43(a). The flow relationship between neighboring centers is shown via links in Fig. 43(b). Fig. 43 is reproduced from [93]. It is clear from Fig. 43 that the centers lend themselves to a graph theoretic network structure.

The flow through a center i is composed of a inflow component and a outflow component. The inflow is composed of the number of aircraft departures (takeoffs from

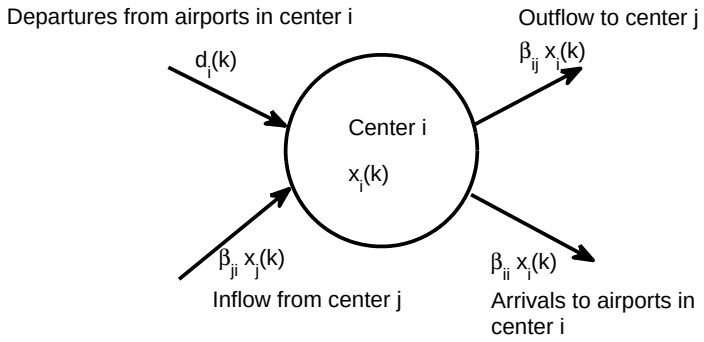


Figure 44: Aircraft flow contributions in center i. Reproduced from [93].

airports in the center i) and the number of aircraft entering center i from neighboring centers j. The outflow is composed of the number of aircraft arrivals (landings at airports in center i) and the number of aircraft leaving the center i to neighboring centers j in a time interval Δt. This flow mechanism is depicted in Fig. 44.

Utilizing the principle of conservation of flow in a center, the number of aircraft in center i at the next time instant $k+1$ can be related to the number of aircraft in the center at the current time k through the difference equation Eq. 32,

$$x_i(k+1) = x_i(k) - \sum_{j=1}^{N} \beta_{ij} x_i(k) + \sum_{j=1, j\neq i}^{N} \beta_{ji} x_j(k) + d_i(k) \tag{32}$$

The number of aircraft in center i is denoted by $x_i(k)$. Departures from airports within center i are denoted by $d_i(k)$. The fractions β_{ij} and β_{ji} are transition probabilities obtained as described in [86]. β_{ij} represents the transition probability of an aircraft from center i traveling to center j. β_{ji} represents the transition probability of an aircraft from center j entering center i.

From the above description of the aggregate flow model, the inflow and outflow

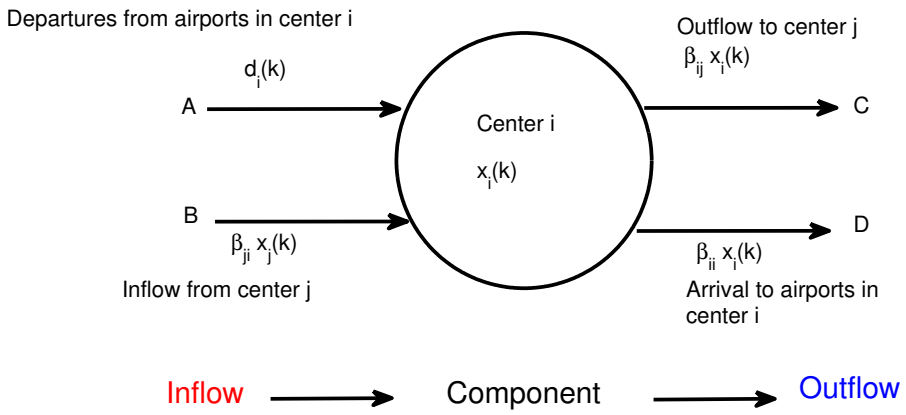


Figure 45: Inflow and outflow of aircraft flow in center i.

into each center becomes clear and we can visualize flow in the centers through Fig. 45. The flow in Fig. 45 is similar to component inflow/outflow in the CS model. Using Eq. 32, the aggregate flow model can be reformulated in terms of inflow and outflow for a CS model component, these terms are shown in Eq. 33.

$$Inflow = \sum_{j=1, j\neq i}^{N} \beta_{ji} x_j(k) + d_i(k) \tag{33a}$$

$$Outflow = \sum_{j=1, j\neq i}^{N} \beta_{ij} x_i(k) + \beta_{ii} x_i(k) \tag{33b}$$

In Eq. 33 the inflow and outflow is composed of the number of aircraft entering and exiting a center. Eq. 32 and Eq. 33 together completely specify the aggregate flow model in a CS model component framework. The number of components would be equal to the number of centers that are considered for analysis. In Fig. 46 we show

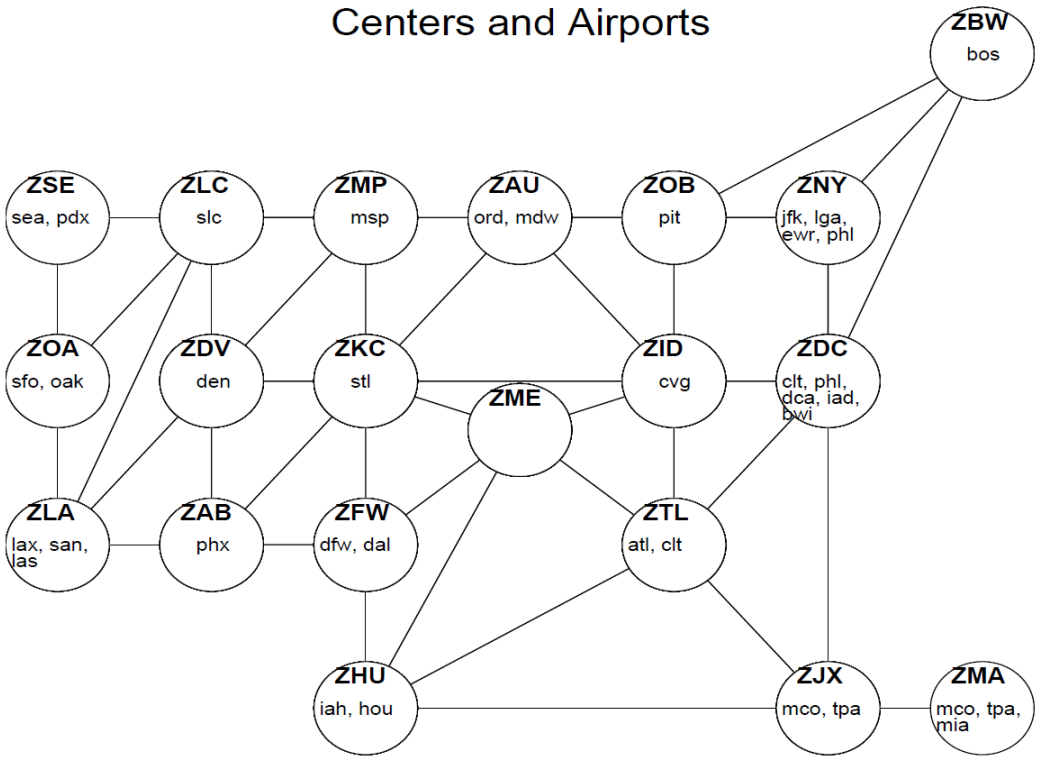


Figure 46: Continental United States airspace centers on a graph network representation. Reproduced from [86].

the centers in the continental United States using a graph network representation which is reproduced from [86]. In the previous chapters we have implemented the CS model on a scale-free network topology. Clearly using Eqs. 32, 33 and Fig. 46 we can model the aggregate flow model for the continental United States in a CS model framework on a graph topology. Note, so far the only thing we have done is to take the aggregate air traffic model and reformulate it in a framework that fits the CS model description. All simulations, analysis and results of the aggregate flow model demonstrated in [86, 91, 92, 93, 22] would also hold in the CS model framework.

5.2.2 Air traffic congestion management using the CS model representation

In this section we investigate the possibility of using the CS model framework for air traffic management developed in the previous section for congestion management in the national airspace system (NAS).

Demand-Capacity imbalances in the national airspace system (NAS) have been reported to cause 215,000 hours of reportable delay [45] between January 2003 and October 2004. Also from the same reference the cost of these delays to airlines have been calculated to be around $700 million. The impact of airport capacity constraints on the NAS delays have been studied in [23]. The authors investigate the impact of arrival or departure capacity reduction at each of the major airports in the NAS on the arrival and departure delays at other major airports in the continental United States. The FAA's Air Traffic Control System Command Center (ATCSCC) regularly utilizes playbook reroutes, ground delay programs (GDPS), ground stops (GSs), miles-in-trail restrictions to mitigate congestion resulting from these demand-capacity imbalances [45].

For congestion management using the CS model air traffic framework we invision using the load sharing property of the CS model to deal with demand-capacity imbalances in the national air space system. A hypothetical congestion management system would work in the following way. The load would be shared within centers by shifting the number of aircraft between them. To understand how this would work we refer the reader again to Fig. 45. Suppose a center i is weather impacted. In that case both the arrival and departure capacity of center i would be reduced. This means the terms A and D in Fig. 45 would undergo a net decrease as would term C but term B would remain the same. This would result in a demand-capacity imbalance at center i. Also the load sharing capacity of center i to accept arriving aircraft would decrease. The demand due to term B would be load shared with the neighboring centers in

the system assuming those centers themselves have not undergone a aircraft arrival capacity reduction. It is our opinion using this scheme it would be possible to study in a structured and aggregate way the impact of demand-capacity imbalances in the national airspace system arising due to weather or other factors.

To test whether this scheme would provide insight and knowledge on congestion management in the national airspace system, as part of future investigations we propose to use the NAS data from [93, 23] in the framework discussed above. It is our opinion, using real aircraft arrival-departure data from [93, 23] would allow us to fully measure the capabilities of the CS model air traffic congestion management framework.

5.3 Road traffic congestion studies using the CS model

In this section we provide an example of how the CS model could be used to develop policy to manage and mitigate road traffic congestion. This example is motivated by the so called "Atlanta Snow Jam 2014" traffic congestion event[1].

The "Atlanta Snow Jam 2014" traffic congestion event occurred on Tuesday, January 28$^{\text{th}}$, 2014. A snow storm was forecast for regions south of the metro Atlanta area. However, the storm arrived early and changed direction slightly. Eventually the metro Atlanta area experienced two and half inches of snow starting around noon. State, city and businesses decided to let employees leave around 1 pm. Schools also decided to send students home around 1 pm. What resulted next was traffic congestion and gridlock chaos of epic proportions. Commutes that usually took 30-45 minutes ended up taking 10-15 hours in bumper to bumper traffic. Many school children spent the night in the cold in their school buses without adequate heating, warm clothes and food. In addition, school children who didn't make it onto school buses spent the night bedding down in their schools.

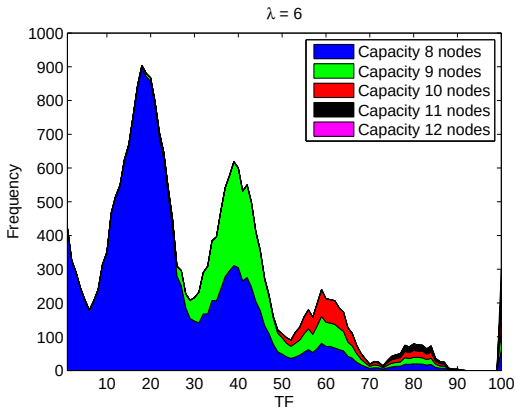
[1]Alexis Stevens, "Metro commuters recount their hours-long trip home", *The Atlanta Journal-Constitution*, January 29, 2014

In hind sight, it became clear that allowing such a large number of vehicles on Atlanta roads at the same time was a monumentally incorrect decision. This resulted in tremendous psychological trauma for those involved and also endangered the lives of school children, the sick and the elderly who were stuck in traffic. However, no policy was in place at the state, city and local levels for the different responsible agencies to communicate and decide on not permitting such a large number of vehicles to enter Atlanta roads at the same time.

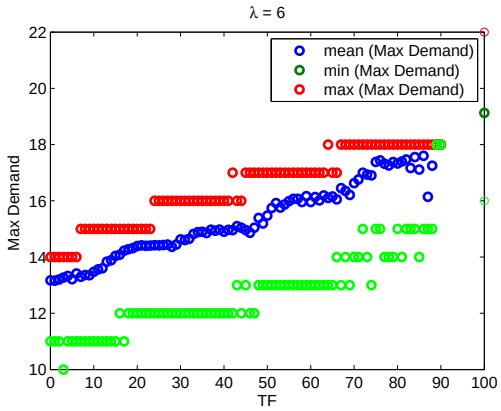
In this section, we use the CS model to provide a traffic congestion interpretation that allows policy makers to observe how the number of vehicles, above a particular threshold introduced at the same time in metropolitan roads, might result in massive congestion. By using the CS model with parameters set to values corresponding to real-world traffic rates and capacities, policy makers possess a tool by which to decide on the level of traffic to introduce to metropolitan area roads at the same time.

The interpretation works in the following way. Consider the maximal dynamics of the CS model from Chapter II which we reproduce again in Fig. 47(a) and 47(b). The figures correspond to the mean traffic arrival rate $\lambda = 6$. We interpret the components of the CS model as corresponding to "segments of metropolitan roads". In this interpretation we do not track the source and destination of traffic. Interpret "Max Demand" as "Maximum number of vehicles observed in a time sample over a observation window".

In Fig. 47, interpret TF=100 to correspond to massive congestion which is a low probability event. An example would be the congestion that was observed during the "Atlanta Snow Jam" event. Interpret a regular metropolitan traffic commute day as TF=15 which represents normally observed congestion. Note we could also interpret TF=10 or TF=20 as a regular commute day. From Fig. 47(b), we can clearly see that the "Max Demand" responsible for TF=100 is much larger than the "Max Demand" for TF=15. Implying that maximal traffic dynamics is responsible for the level of



(a) Multiple failures modes



(b) Maximum Demand as a function of TF

Figure 47: Extremal behavior of the CS model for $\lambda = 6$ (Color Online)

congestion. For both cases the average traffic $\lambda = 6$ is the same.

Based on the above analysis, policy makers could adapt the CS model parameters to real-world traffic rates and capacities of different metropolitan areas. CS model components would represent road segments of interest. They could then forecast the threshold value of vehicles which would cause different levels of congestion in the metropolitan area of interest. Based on the forecasts, policy could be formulated to manage and mitigate traffic congestion in the metropolitan area of interest.

CHAPTER VI

MOTION PLANNING FOR DISTRIBUTED MULTI-AGENT SYSTEMS USING THE DUBINS VEHICLE MODEL

In this chapter, we implement the Markov-Dubins vehicle model to study air traffic congestion phenomena for distributed multi-agent systems. We extend the Markov-Dubins vehicle model to 3-dimensional space and time. The model is implemented using the NETLogo programming language. We refer to the implementation as NET-Dubins. We demonstrate NETDubins for multi-agent simulations utilizing user defined trajectories, optimal trajectories and helical maneuvers. We provide an example of using NETDubins for simulating notional air traffic vectoring at Los Angeles International Airport.

6.1 Introduction

NETDubins is a trajectory generation toolbox written in the NETLogo programming dialect. NETDubins is designed for multi-agent system simulations in a 3D environment. The toolbox was implemented as part of NASA's NextGen research effort.

The algorithm employed by the toolbox is based on the Markov-Dubins vehicle model, which is a 2D vehicle. The implemented algorithm extends the Markov-Dubins model to incorporate vehicles in a 3D world. This extension allows for the generation of 3-dimensional trajectories for multi-agent simulations.

Although the toolbox is implemented for use in NETLogo, a MATLAB version of the code exists. The MATLAB version is mainly used as a test bed for rapid prototyping and debugging of new procedures and concepts.

6.2 History

The Markov-Dubins model deals with the problem of curvature-constrained, shortest paths in the plane with prescribed initial and terminal positions and orientations.

As described in [8], this problem can be traced back to the end of the nineteenth century when the Russian mathematician A. A. Markov first posed the problem for joining pieces of railway tracks. Characterizing the minimal path completely was only done recently by Dubins [38] and by Reeds and Shepp [85]. Another way of solving this problem, using optimal control theory, has recently been proposed in [19]. In this effort synthesizing the shortest or optimal path is achieved through purely geometric techniques. This is because, sophisticated mathematical libraries that are available in MATLAB are not readily available in NETLogo. Examples include Linear Algebra, Optimization, Differential equation libraries, which although not readily available in NETLogo can be developed with necessary investment. The object for this development is for a toolbox to generate trajectories for multi-agent systems using NETLogo's existing software infrastructure.

6.3 2-Dimensional Theory

Given two oriented points in the plane, (x_i, y_i, θ_i) and (x_f, y_f, θ_f) , it is required to determine the shortest path of bounded curvature (ρ) joining them. The Markov-Dubins kinematic model is given by the following equations where (κ) is the control input:

$$\begin{aligned} \dot{x} &= cos\theta \\ \dot{y} &= sin\theta \\ \dot{\theta} &= \kappa/\rho \end{aligned} \tag{34}$$

Each oriented point can travel in either a clockwise (denoted CW_i and CW_f) or counter-clockwise (denoted CC_i and CC_f) circle. Between the two departure circles

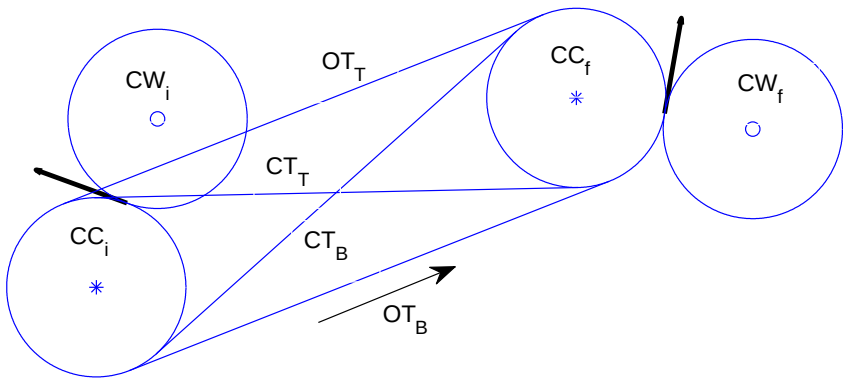


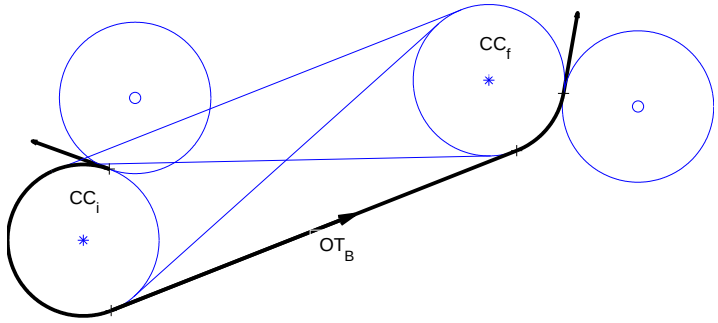
Figure 48: Two oriented points in the plane. Also shown are the two possible circles (radius $\frac{1}{\rho}$) of travel for each orientation and the four tangents connecting one possible circle combination

and the two arrival circles there are four possible departure/arrival circle combinations.

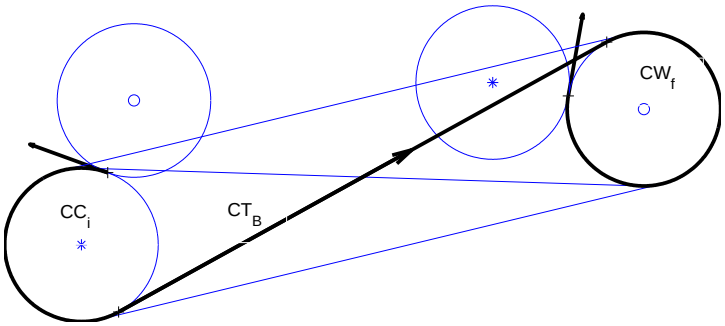
There exists four tangents between two circles that don't intersect at any point and also do not lie within each other. Two of these tangents intersect and are referred to as "cross tangents" (denoted CT_T and CT_B). The other two tangents are referred to as "outer tangents" (denoted OT_T and OT_B).

For a given initial orientation θ_i, due to the directional constraint on motion, the vehicle can proceed along two of the four tangents. For example, in Figure 48 the vehicle can only proceed along tangents CT_B and OT_B. Of these two tangents the vehicle can arrive at orientation θ_f only along tangent OT_B otherwise the directional constraint on motion would be violated. This implies for a given departure/arrival circle combination there exists only one optimal path from a orientation θ_i to a orientation θ_f. Since there are four possible circle combinations, there exists four optimal paths. The shortest path is selected from these four optimal paths.

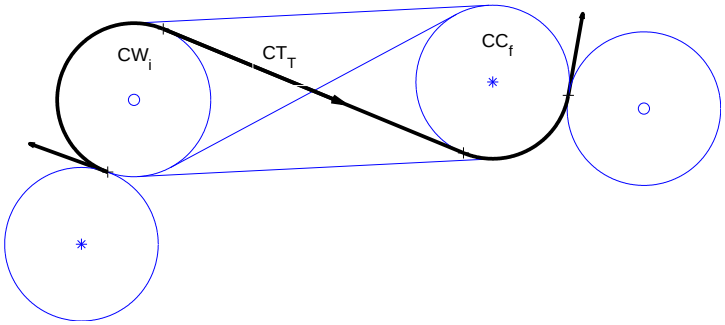
Below in Figure 49, the four optimal paths are shown for the oriented points of Figure 48. Of these four optimal paths Figure 49(c) corresponds to the shortest path.



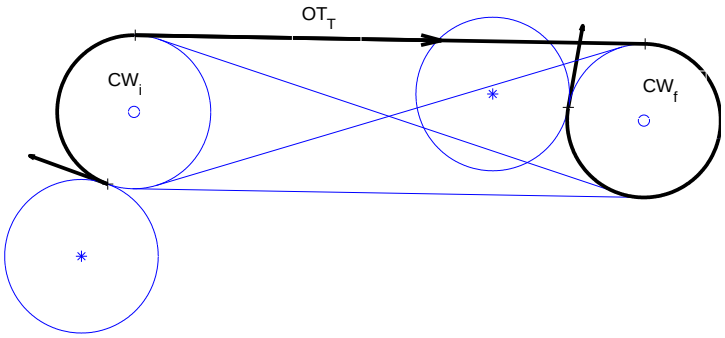
(a) Optimal path from circle CC_i to circle CC_f



(b) Optimal path from circle CC_i to circle CW_f



(c) Optimal and shortest path from circle CW_i to circle CC_f



(d) Optimal path from circle CW_i to circle CW_f

Figure 49: The four optimal paths between two oriented points x_i, y_i, θ_i and x_f, y_f, θ_f

6.4 Extension to 3-Dimensions

Given two points with coordinates and orientation, (x_i, y_i, z_i, ψ_i) and (x_f, y_f, z_i, ψ_f), it is required to find the shortest path. In this case it is assumed the orientations, ψ_i and ψ_f, correspond to the "heading" of the vehicle in a 3D environment or the angle between the positive y-axis and the projection of the velocity vector on the 2D plane in a right handed system. The radius, $1/\rho$, relates to the "roll" of the vehicle. With these assumptions, the shortest 3D path of the vehicle, d_{3d}, is determined through the Pythagorean theorem and the "Flight path angle", γ, is determined through trigonometry.

First, the 3D geometry is projected on to the 2D plane, from (x_i, y_i, z_i, ψ_i) and (x_f, y_f, z_i, ψ_f) to (x_i, y_i, ψ_i) and (x_f, y_f, ψ_f). Then, as described in section 6.3, the shortest 2d path, d_{2d}, is determined. The length of the shortest path, d_{2d}, and the altitude drop, $z_i - z_f$, is used to calculate the length of the shortest 3D path, d_{3d}, and the flight path angle, γ as:

$$d_{3d} = \sqrt{d_{2d}^2 + (z_i - z_f)^2}$$
$$\gamma = \cos^{-1}(\frac{d_{2d}}{d_{3d}})$$

With the prescribed heading information, ψ (from the 2D analysis) and the computed flight path angle, γ, it is possible to move around optimally in the 3D world. The examples in the next sections illustrates this.

Subsequently, the kinematics for the "Extended Markov-Dubins" model is given by the equations:

$$\begin{aligned} \dot{x} &= cos\psi \\ \dot{y} &= sin\psi \\ \dot{z} &= \dot{d_{2d}} tan\gamma \\ \dot{\psi} &= \kappa/\rho \end{aligned} \tag{35}$$

6.4.1 Dealing with limits on Pitch or Flight Path Angle

In many cases it is desirable that the Pitch or Flight path angle of a vehicle not exceed a certain threshold, denoted γ_{limit}. The toolbox deals with this scenario in the following way. First, a shortest path is computed. If the trajectory γ violates γ_{limit}, then the radius's are increased until γ satisfies γ_{limit}. The radius's also have an upper limit imposed on them, denoted r_{limit}. If γ_{limit} is not satisfied even though r_{limit} has been reached, in that case a helical maneuver is executed. The helical maneuver is discussed in section 6.8

6.5 Computational algorithm

6.5.1 Directional Vector and Circle center determination

The position and orientation of the initial and final point is given. Using this information the directional vector of the orientation is determined. The directional vector is tangent to both the clockwise and counter-clockwise circle. Moving $\pm 90^o$ in either direction of the directional vector for a length of $1/\rho$ (the circle radius) provides the circle centers for both the clockwise and counter-clockwise motion.

6.5.2 Tangent calculation algorithm

Previously, we have described the procedure for determining the circle centers and the radius of the circles. For a given departure/arrival circle configuration we can calculate the possible four tangents. Travel is possible for only one of these tangents for a given departure/arrival circle combination. Using rules from trigonometry we can calculate the position and orientation of the two "cross tangents" and the two "outer tangents".

6.5.3 Travel tangent deduction algorithm

A given departure/arrival circle combination results in four possible tangents. Of these four tangents, motion is possible only across one of these tangents. Using

initial and final heading information, the orientation information of the four tangents and vector cross product rules it is possible to deduce the tangent that will be utilized for vehicle motion. The travel tangent deduction algorithm is illustrated in Fig. 50.

6.5.4 Motion planning algorithm

Once the trajectory has been determined, motion can be scheduled along the path. The path is composed of three separate components. These are the departure arc, the travel tangent and arrival arc. The motion planning algorithm is illustrated in Fig. 51.

6.6 Example: Shortest path maneuver

Figure 52(a) and Figure 52(b) demonstrates the shortest 3d path using the algorithm described in section 6.4.

6.7 Example: User selected optimal trajectory

A user can select which one of the four departure/arrival circle combinations the vehicle should traverse. These four combinations are demonstrated in Figure 52(c) and Figure 52(d) for the same initial and final position and orientation.

6.8 Example: Helical maneuver

The user also has the option of executing a helical maneuver if desired. Currently the helical maneuver is invoked based on the γ_{limit} parameter. The procedure works as follows, if the γ_{limit} constraint is not satisfied by γ even after increasing the radius's of the departure/arrival circle combination than the vehicle initiates a helical maneuver from (x_i, y_i, z_i, ψ_i) to (x_i, y_i, z_t, ψ_i). In the helical maneuver γ_{limit} is not violated. Then the vehicle follows a optimal/shortest path from (x_i, y_i, z_t, ψ_i) to (x_f, y_f, z_f, ψ_f). z_t is at an altitude such that γ_{limit} is not violated. Figure 53 illustrates a spiral maneuver. The helical maneuver can be modified as necessary.

6.9 Example: LAX CDA (Continuous Descent Approach) vectoring maneuver

Continuous Descent Approach (CDA) is a method by which aircraft approach airports prior to landing. It is designed to minimize fuel consumption and noise compared to the conventional landing. A description of CDA is provided in [26]. The hazard scenarios of CDA at Los Angles airport (LAX) has been modeled in [42, 43]. A diagram of the LAX CDA is shown in Figure 54.

In [42, 43] a simulation model implemented in NETLOGO ([105]) for the LAX CDA approach is described. This simulation model is used as a test bed in the subsequent discussion. Due to conflict arising between incoming aircraft on the LAX CDA a vectoring maneuver is executed for conflict resolution as a last resort. This vectoring maneuver has been modeled using the NETDubins toolbox.

Each vectoring trajectory is modeled by concatenating two optimal paths. First, a required trajectory length (based on airspeed) that will de-conflict the aircraft is established. This trajectory is broken into two individual components and the point where the two trajectories are concatenated (the "join-point") is established. The first trajectory moves the aircraft from the CDA vectoring start point to the join-point. The second trajectory moves the aircraft from the join-point to the CDA trajectory merge point. The calculation of the join-point is discussed next.

6.9.1 Trajectory join-point calculation

Figure 55 illustrates the trajectory join-point calculation. The trajectory "join-point", denoted by J, is calculated using trigonometric techniques. Let the vectoring start and end points be A and B respectively. First, A and B are projected on to the 2d plane and the 2d distance between A and B, d_{AB}, is calculated. In the same manner the required 2d trajectory length, T_L, is calculated.

From Figure 55, we must have $T_L = s_1 + s_2$. First, the parameter θ is selected.

Next the first segment length, s_1, is calculated using the following formula derived from trigonometric relations:

$$s_1 = \frac{T_L^2 - d_{AB}^2}{2(T_L - d_{AB}\cos\theta)}$$

Once s_1 is known, the coordinates of J can be calculated as,

$$J_x = A_x + s_1\cos\theta$$

$$J_y = A_y + s_1\sin\theta$$

The z coordinate of J is determined using the ratio s_1/T_L which is used to determine the corresponding drop in altitude.

In Figure 55, θ explicitly controls h due to the constraint $T_L = s_1 + s_2$. The parameter effect of θ on h is demonstrated in Figure 56. Making θ smaller makes h larger. In Figure 56 all the trajectories length are approximated by fitting triangles and are of equal length. In the simulations θ is picked so that h is minimized but the aircraft does not fly "backwards".

Figure 57 illustrate two trajectories, A and B, that are of equal lengths in a triangle approximation. The synthesized lengths of A and B are within 3% of each other. Trajectory A was generated by setting $\theta = 60^o$ and trajectory B was generated by setting $\theta = 90^o$. Trajectory A has higher h than trajectory B.

6.9.2 Optimization algorithm to enforce kinematic constraints

The kinematics of the extended Markov-Dubins vehicle is given by equation 35. The aircraft kinematic constraints are enforced using the standard aircraft performance metrics: Load factor, n, Bank angle, ϕ, Flight path angle, γ and Turn radius, R. Refer to [100] for an excellent discussion on turning flight in descending altitude. For the CDA vectoring maneuver the constraints are taken to be:

$$\gamma \leq 3^o$$

$$\phi \leq 25^o$$

For $\gamma \in (0, 3)$, we have $cos\gamma = 0.99 \approx 1$, hence γ dependence is weak. The equations for the performance metrics are given by:

$$\cos\phi = \frac{cos\gamma}{n} \tag{36}$$

From equation 36, minimization of ϕ can be done by minimizing n. From Newton's equations for 3d turning flight, we can derive the following expression for the turning radius, R:

$$R = \frac{(Vcos\gamma)^2}{g\sqrt{n^2 - cos\gamma^2}} \tag{37}$$

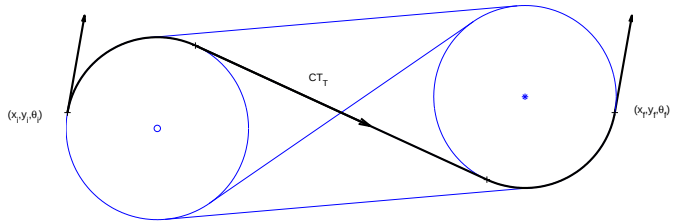
Here V is the true airspeed of the vehicle. Solving for n^2, we have

$$\begin{aligned} n^2 &= \frac{(Vcos\gamma)^4}{(Rg)^2} + cos\gamma^2 \\ \implies n^2 &\approx \frac{(V)^4}{(Rg)^2} + 1 \end{aligned} \tag{38}$$

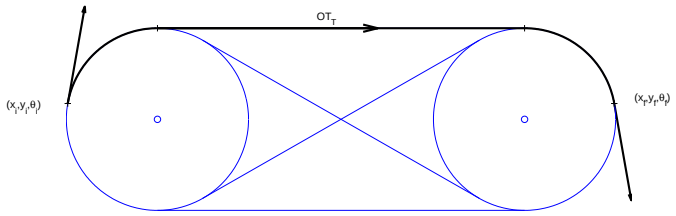
Thus maximizing R and minimizing V will minimize n and a basis for an optimization algorithm is established.

The implemented optimization algorithm works in the following way. First, a upper bound on the turning radius R is established based on the true airspeed V of the vehicle and the Bank Angle ϕ constraint. Then the turn radius is incrementally decreased in a loop while checking that the Bank angle ϕ and Flight path angle γ constraints are met. This algorithm was chosen due to the geometry of the LAX CDA approach. As mentioned in section 6.3, the optimal path algorithm only works in the case of non-intersecting departure/arrival circle combinations. A algorithm based on radius decrease implies less computation to synthesize a optimal trajectory.

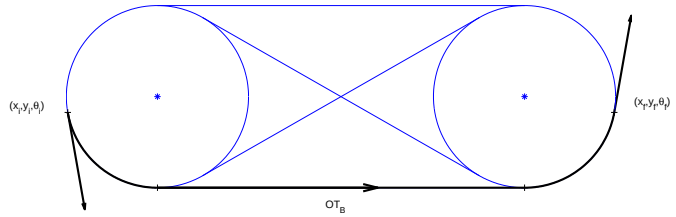
Using this approach, Figures 58(a) and 58(b) shows synthesized trajectories with $V = 265$ knots, $R = 2.4$ nautical miles, $\phi = 23^o$, $\gamma = -2.3^o$and $n = 1.08$. Figures 58(c) and 58(d) shows synthesized trajectories for $V = 400$ knots, $R = 5.2$ nautical miles, $\phi = 24^o$, $\gamma = -2.4^o$and $n = 1.09$.



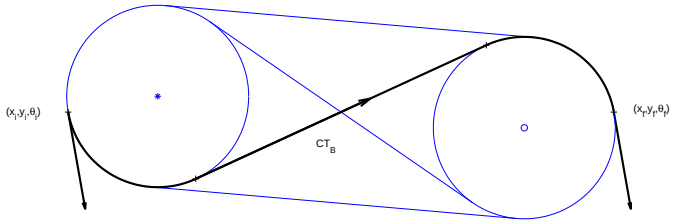
(a) Departure motion is clockwise and arrival motion is counter-clockwise.



(b) Departure and arrival motion is clockwise.

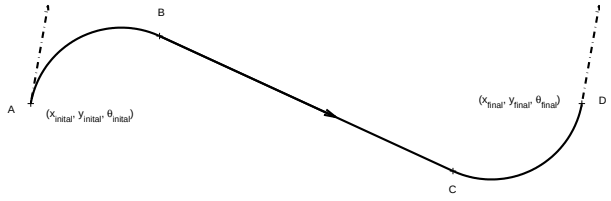


(c) Departure motion and arrival motion is counter-clockwise.

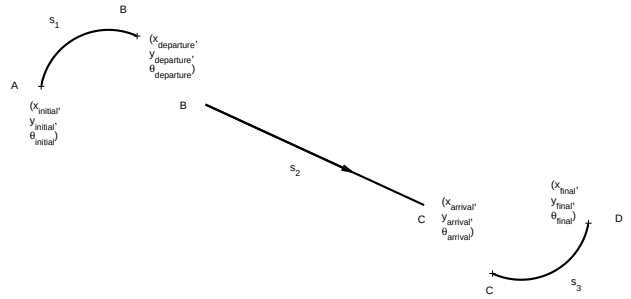


(d) Departure motion is counter-clockwise and arrival motion is clockwise.

Figure 50: Illustration to describe travel tangent deduction algorithm.

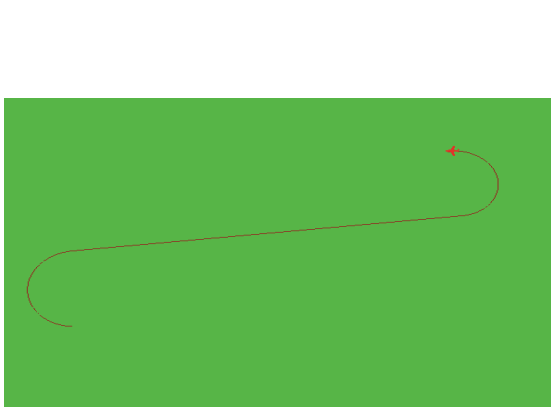


(a) A synthesized optimal path.

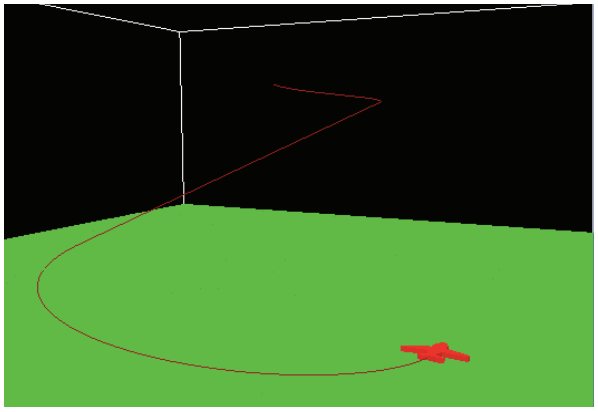


(b) Path discretized into three seperate components.

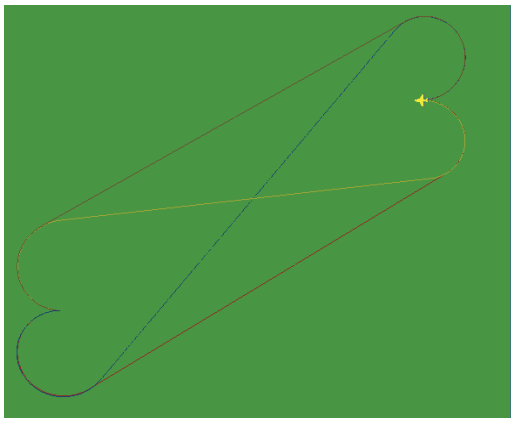
Figure 51: Illustration to describe motion planning algorithm.



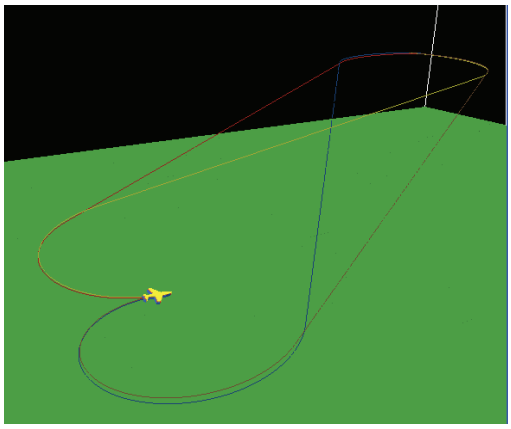
(a) Top view: Shortest 3d path



(b) Side View: Shortest 3d path



(c) Top view: Four different optimal paths selected by user



(d) Side View: Four different optimal paths selected by user

Figure 52: Figures (a) and (b) illustrate the shortest path maneuver, figures (c) and (d) illustrate the optimal path maneuvers.

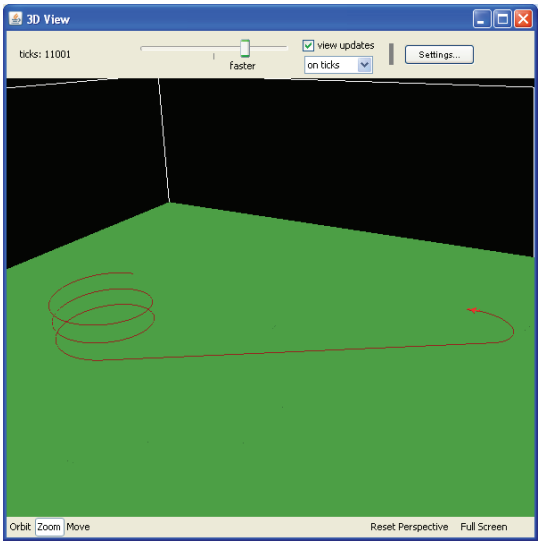


Figure 53: Helical Maneuver initiated by Dubins vehicle

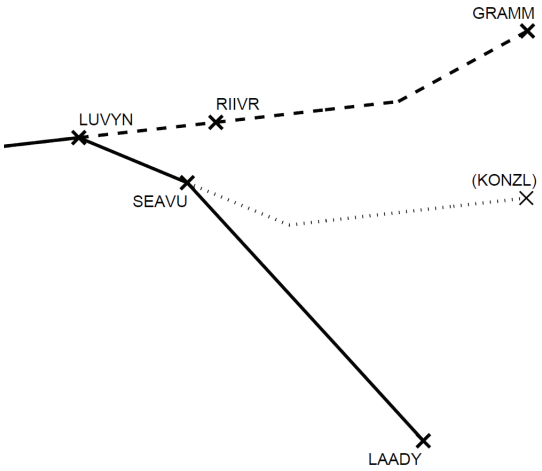


Figure 54: Diagram of the LAX CDA approach

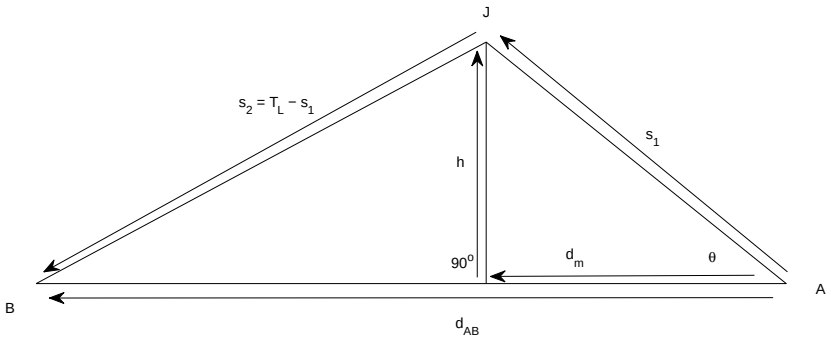


Figure 55: Calculating the point where the two trajectories are concatenated

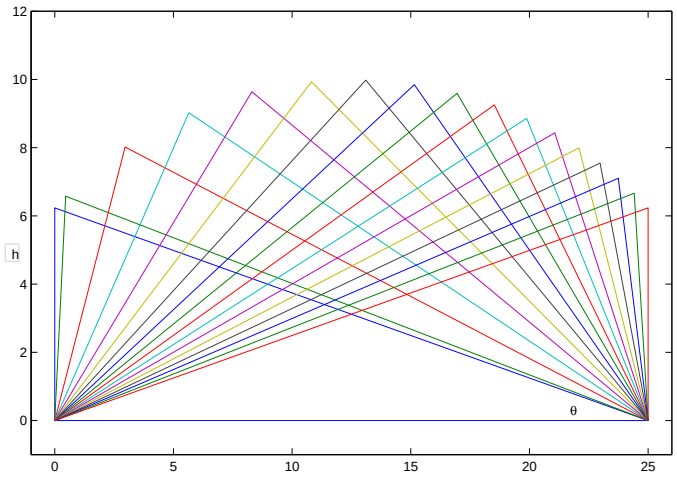


Figure 56: Effect of θ on the height (h) of the triangles

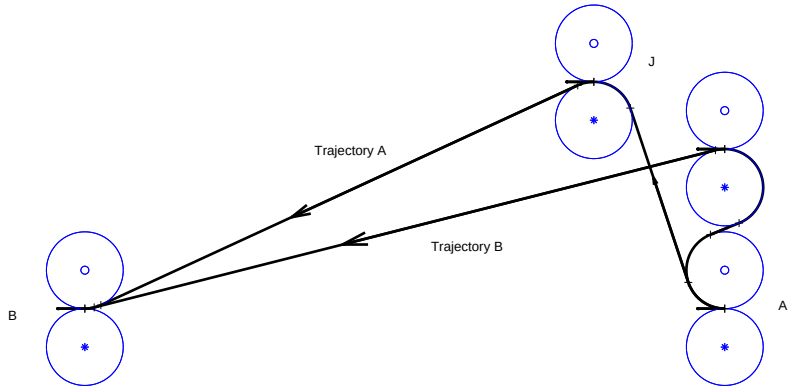


Figure 57: Trajectories A and B are of approximately equal length. Trajectory A was generate by setting $\theta = 60^o$ and trajectory B was generated by setting $\theta = 90^o$

(a) Top view: Vectoring (yellow trajectory) at 265 knots

(b) Side View: Vectoring (yellow trajectory) at 265 knots

(c) Top view: Vectoring (yellow trajectory) at 400 knots

(d) Side View: Vectoring (yellow trajectory) at 400 knots

Figure 58: Modeling of vectoring for CDA approach at LAX.

CHAPTER VII

CONTRIBUTIONS AND FUTURE WORK

7.1 Contributions

- In this work we investigated two different extremal models characterizing different failure mechanisms. The first one, a Loss of Strength (LOS) model where component strength undergoes degradation when component load reaches a certain threshold. The component fails when component load is greater than component strength. After the components fails its load is redistributed to its neighboring components.

  The second one, a Customer Service (CS) model where component demand is modeled as arriving customers to components and components possess fixed capacity to service arriving customers. The component demand/capacity interaction dictates load sharing with neighboring components. If demand exceeds capacity and unsuccessful load sharing results in component queue's being overwhelmed then the component fails and is removed from the network.

- We implemented both models on lattice and scale-free graph network topologies. The models exhibits different failure mechanisms depending on the network topology.

- At critical loads, the LOS model on a lattice network exhibits power-law failure time distributions. On scale-free graph networks, at critical loads with decreasing network connectivity, the LOS model loses power-law scaling in failure time distribution. At critical loads, the CS model exhibits exponential failure time distributions.

- The LOS model demonstrated greater resilience in scale-free graph network topologies compared to lattice topologies. For scale-free graph topologies, the LOS model demonstrated greater resilience with decreasing network connectivity.

- We established parallels between the SOC signatures of the Bak-Sneppen evolution model and the CS model. Similar to the Bak-Sneppen model, the CS model demonstrates punctuated equilibrium in its evolution to the critical state.

- Transition loadings, i.e. tipping points, excite different modes of failure for both the LOS and CS models. At transition loadings, both models, may or may not, descend into different scales of failure. Hence, we characterize transition loadings as 'tipping-points' for the systems.

- We described a Markov chain formulation of the CS model. We demonstrated that the Markov chain formulation agreed with the simulation results for the CS model.

- We provided a mathematical framework for cascading failure in the LOS model. We identified the conditions required for triggering, and also for mitigating, cascading failure in the LOS model. This description has potential for mitigating cascading failure in real systems such as smart grids and power grids.

- We used the LOS model to study creep-rupture phenomena in fiber-matrix composite structures. We established parallels between our results for the LOS model creep-rupture framework and the Mahesh and Phoenix creep-rupture model.

- We provided a framework for studying air traffic congestion management using the CS model. Specifically we described methods of reformulating aggregate

flow models for air traffic management in terms of the CS model on a graph network topology.

- We interpreted metropolitan road traffic congestion using the CS model. Specifically we provided a general description on mapping CS model parameters to real-world traffic rates and capacities. Using more detailed models developed based on this description, policy-makers and responsible agencies would be able to forecast and manage traffic congestion in metropolitan areas of interest.

7.2 Future work

A number of directions for future studies of complex system failure mechanisms are outlined next.

- In this work we considered networks of relatively small sizes for both lattice and scale-free graph topologies. A logical next step would be to increase the size of the network. This would facilitate studies of real networks such as the Internet, power grids and transportation networks which tend to be large.

- In this work we analyzed the CS model using Markov chain theory. One drawback of Markov chain theory is the explosion in state space when considering large networks. On the other hand, mean-field theory provides a compact but powerful means to study multi-component systems. Thus modeling the CS model utilizing mean-field theory is a logical next step.

- In our work we have described, mathematically and in simulations, the cascading failure property of the LOS model. In electrical power grids, blackouts often occur when several components (generators) fail simultaneously or in quick succession thus inducing cascading failure events and ultimately shutting down the entire system. Thus the LOS model can be potentially used to study cascading failure events in electrical power grids and smart grids.

- In our work we have noted the parallels between the failure mechanisms of the CS model and draw-downs in financial markets. It is worth investigating the exact nature of the relationship between the CS model and models in financial applications which demonstrate similar failure patterns.

REFERENCES

[1] Albert, R., Jeong, H., and Barabási, A.-L., "Error and attack tolerance of complex networks," *Nature*, vol. 406, pp. 378–482, 2000.

[2] Alekseev, D. V. and Kazunina, G. A., "Simulation of damage accumulation kinetics with a probabilistic cellular automaton," *Physics of the Solid State*, vol. 48, no. 2, pp. 272–278, 2006.

[3] Amaral, L. A. N., Scala, A., Barthélemy, M., and Stanley, H. E., "Classes of small-world networks," *Proc. Natl. Acad. Sci. (USA)*, vol. 97, no. 21, p. 11149, 2000.

[4] Azzalini, A. and Bowman, A. W., "A look at some data on the old faithful geyser," *Applied Statistics , 357-365*, vol. 39, pp. 357–365, 1990.

[5] Bak, P., *How Nature works: The Science of Self-Organized Criticality*. Copernicus, New York, 1996.

[6] Bak, P. and Sneppen, K., "Punctuated equilibrium and criticality in a simple model of evolution," *Phys. Rev. Lett.*, vol. 71, p. 4083, December 1993.

[7] Bak, P., Tang, C., and Wiesenfeld, K., "Self-organized criticality: An explanation of 1/f noise," *Physical Review Letters*, vol. 59, p. 381, July 1987.

[8] Bakolas, E. and Tsiotras, P., "The asymmetric sinistral/dextral markov-dubins problem," *48th IEEE Conference on Decision and Control, Shanghai, China*, pp. 5649 – 5654, December 2009.

[9] Barabási, A.-L. and Albert, R., "Emergence of scaling in random networks," *Science*, vol. 286, pp. 509–512, 1999.

[10] Barabási, A.-L., Albert, R., and Jeong, H., "Mean-field theory for scale-free random networks," *Physica A*, vol. 272, pp. 173–187, 1999.

[11] Barlow, R. and Proschan, F., *Mathematical Theory of Reliability*. John Wiley and Sons, New York, 1965.

[12] Bertsekas, D. and Gallager, R., *Data Networks*. Prentice Hall, 1992.

[13] Biondini, F., Malerba, P., Bontempi, F., and Frangopol, D. M., "Cellular automata approach to durability analysis of concrete structures in aggressive environments," *Journal of Structural Engineering*, vol. 130, no. 11, p. 17241737, 2004.

[14] Birnbaum, Z. W. and Saunders, S. C., "A probabilistic interpretation of miners rule," *SIAM Journal on Applied Mathematics*, vol. 16, no. 3, p. 637652, 1968.

[15] Birnbaum, Z. W. and Saunders, S. C., "A new family of life distributions," *Journal of Applied Probability*, vol. 6, no. 2, p. 319327, 1969.

[16] Boccaletti, S., Latora, V., Moreno, Y., Chavez, M., and Hwang, D.-U., *Complex Networks: Structure and Dynamics*. Physics Reports 424, 2006.

[17] Bollobás, B., *Modern Graph Theory, Graduate Texts in Mathematics*. Springer, New York, 1998.

[18] Broder, A., Kumar, R., Maghoul, F., Raghavan, P., Rajagopalan, S., Stata, R., Tomkins, A., and Wiener, J., "Graph structure in the web," *Computer Networks*, vol. 33, p. 309, 2000.

[19] Bui, X.-N., Souérés, P., Boissonnat, J., and Laumond, J.-P., "Shortest path synthesis for dubins nonholonomic robot," *Proceedings of the 11th IEEE International Conference on Robotics and Automation, San Diego, California*, 1994.

[20] Carlson, J. M. and Doyle, J., "Highly optimized tolerance: A mechanism for power laws in designed systems," *Phys. Rev. E*, vol. 60, no. 2, p. 1412, 1999.

[21] Carlson, J. M. and Doyle, J., "Highly optimized tolerance: Robustness and design in complex systems," *Physical Review Letters*, vol. 84, no. 11, pp. 2529 – 2532, 2000.

[22] Chatterji, G. B. and Sridhar, B., "Some properties of the aggregate flow model of air traffic," *AIAA 5th Aviation, Technology, Integration, and Operations Conference (ATIO)*, no. 7456, 2005.

[23] Chatterji, G. B. and Zheng, Y., "Impact of airport capacity constraints on national airspace system delays," *AIAA 7th Aviation Technology, Integration and Operations Conference (ATIO)*, no. 7712, 2007.

[24] Chopard, B. and Droz, M., *Cellular automata modeling of physical systems*. New York, New York: Cambridge University Press, 1988.

[25] Chrzanowski, M. and Nowak, K., "Cellural automata in damage mechanics: Creep rupture case," *Archive of Mechanics*, vol. 59, no. 4-5, p. 329339, 2007.

[26] Clarke, J.-P., Ho, N., and Ren, L., "Continuous descent approach: Design and flight test for louisville international airport," *Journal of Aircraft*, vol. 41, pp. 1054–1066, September-October 2004.

[27] Coleman, B. D., "Time dependence of mechanical breakdown phenomena," *Journal of Applied Physics*, vol. 27, pp. 862–866, 1956.

[28] Coleman, B. D., "A stochastic process model for material breakdown," *Transactions of the Society of Rheology*, vol. 1, pp. 153–168, 1957.

[29] Coleman, B. D., "Time dependence of mechanical breakdown of fibers i. constant total load," *Journal of Applied Physics*, vol. 28, pp. 1058–1064, 1957.

[30] Coleman, B. D., "On the strength of classical fibers and fiber bundles," *Journal of the Mechanics and Physics of Solids*, vol. 7, pp. 60–70, 1958.

[31] Crucitti, P., Latora, V., and Marchiori, M., "Model for cascading failures in complex networks," *Phys. Rev. E*, vol. 69, p. 045104(R), April 2004.

[32] Crucitti, P., Latora, V., Marchiori, M., and Rapisarda, A., "Efficiency of scale-free networks: Error and attack tolerance," *Physica A*, vol. 320, p. 622, 2003.

[33] Crucitti, P., Latora, V., Marchiori, M., and Rapisarda, A., "Error and attack tolerance of complex networks," *Physica A*, vol. 340, p. 388, 2004.

[34] Curtin, W. A. and Scher, H., "Time-dependent damage evolution and failure in materials. i. theory," *Phy. Rev. B*, vol. 55, no. 18, 1997.

[35] de Castro, R. and Grossman, J. W., "Famous trails to paul erdos," *Math. Intell. 21 51*, 1999.

[36] De Martino, D., Dall'Asta, L., Bianconi, G., and Marsili, M., "Congestion phenomena on complex networks," *Phys. Rev. E*, vol. 79, p. 015101(R), January 2009.

[37] Dorogovtsev, S. and Mendes, J., *Evolutions of Networks: From Biological Nets to the Internet and WWW*. Oxford University Press, 2003.

[38] Dubins, L., "On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents," *American Journal of Mathematics*, vol. 79, pp. 497–516, 1957.

[39] Ebel, H., Mielsch, L.-I., and Bornholdt, S., "Scale-free topology of e-mail networks," *Phys. Rev. E*, vol. 66, p. 035103, 2002.

[40] Echenique, P., Gómez-Gardeñes, J., and Moreno, Y., "Dynamics of jamming transitions in complex networks," *Europhys. Lett.*, vol. 71, no. 2, p. 325, 2005.

[41] Embrechts, P., Kluppelberg, C., and Mikosch, T., *Modelling Extremal Events for Insurance and Finance*. Springer, New York, 1997.

[42] Fraccone, G. C., Valenzuela-Vega, R., Siddique, S., Volovoi, V., and Kirlik, A., "Nested modling for evaluating hazards associated with off-nominal scenarios in the national airspace system," *Proceedings of the 10th AIAA Aviation Technology, Integration, and Operations Conference (ATIO), Fort Worth, Texas*, Sep. 13-15 2010.

[43] Fraccone, G. C., Valenzuela-Vega, R., Siddique, S., Volovoi, V., and Kirlik, A., "Nested modeling of hazards in the national airspace system," *Journal of Aircraft*, vol. 50, no. 2, pp. 370–377, 2013.

[44] Galambos, J., *The Asymptotic Theory of Extreme Order Statistics 2nd ed.* R.E. Krieger Pub. Co., Malaba, Fla, 1987.

[45] Grabbe, S. R. and Sridhar, B., "Congestion management with an aggregate flow model," *AIAA Guidance, Navigation, and Control Conference and Exhibit*, no. 6277, 2005.

[46] Grossman, J. and Ion, P., "On a portion of the well-known collaboration graph," *Congr. Numer.*, vol. 108, p. 129, 1995.

[47] Gumbel, E., *Statistics of Extremes*. Columbia University Press, New York, 1958.

[48] Ibnabdeljalil, M. and Phoenix, S. L., "Creep rupture of brittle matrix composites reinforced with time dependent fibers : Scalings and monte carlo simulations," *J. Mech. Phys. Solids*, vol. 43, no. 6, p. 897, 1995.

[49] Jacobson, V., "Congestion avoidance and control," in *Symposium Proceedings on Communications Architectures and Protocols*, vol. 18 of *SIGCOMM '88*, (New York, NY, USA), p. 314, ACM, August 1988.

[50] Jeong, H., Mason, S., Barabási, A.-L., and Oltvai, Z., "Lethality and centrality in protein networks," *Nature*, vol. 411, p. 41, 2001.

[51] Jeong, H., Tombor, B., Albert, R., Oltvai, Z., and Barabási, A.-L., "The large-scale organization of metabolic networks," *Nature*, vol. 407, p. 651, 2000.

[52] Johansen, A. and Sornette, D., "Large stock market price drawdowns are outliers," *J. Risk*, vol. 4, no. 2, pp. 69–110, 2002.

[53] Kemeny, J. G. and Snell, J. L., *Finite Markov Chains*. Springer-Verlag, 1960.

[54] Kim, D.-H., Kim, B. J., and Jeong, H., "Universality class of the fiber bundle model on complex networks," *Physical Review Letters*, vol. 94, p. 025501, January 2005.

[55] Kinney, R., Crucitti, P., Albert, R., and Latora, V., "Modeling cascading failures in the north american power grid," *Eur. Phy. J. B*, vol. 46, no. 1, pp. 101–107, 2005.

[56] Lacasa, L., Cea, M., and Zanin, M., "Jamming transition in air transportation networks," *Physica A*, vol. 388, no. 18, p. 3948, 2009.

[57] Lamb, H., *Hydrodynamics 6th ed.* Cambridge University Press, 1932.

[58] Lemoine, A. J. and Wencour, M. L., "On failure modeling," *Naval Research Logistic Quarterly*, vol. 22, pp. 497 – 508, 1985.

[59] Lewin, R., *Complexity - Life at the Edge of Chaos.* Macmillan, New York, 1992.

[60] Li, L., Li, X. G., Dong, C. F., and Cheng, Y. F., "Cellular automaton model for simulation of metastable pitting," *Corrosion Engineering, Science and Technology*, vol. 46, no. 4, 2011.

[61] L'vov, V. S., Pomyalov, A., and Procaccia, I., "Outliers, extreme events, and multiscaling," *Phys Rev. E*, vol. 63, p. 056118, April 2001.

[62] Mahesh, S. and Phoenix, S. L., "Lifetime distributions for unidirectional fibrous composites under creep-rupture loading," *International Journal of Fracture*, vol. 127, no. 4, pp. 303 – 360, 2004.

[63] Marro, J. and Dickman, R., *Nonequilibrium Phase Transitions in Lattice Models.* Cambridge University Press, Cambridge, 1999.

[64] Matic, P. and Geltmacher, A. B., "A cellular automaton-based technique for modeling mesoscale damage evolution," *Computational Materials Science*, vol. 20, pp. 120–141, 2001.

[65] Meeker, W. and Escobar, L., *Statistical Methods for Reliability Data.* Wiley Series in Probability and Statistics, Wiley-Interscience, 1998.

[66] Moreno, Y., Gómez, J., and Pacheco, A., "Instability of scale-free networks under node-breaking avalanches," *Europhys. Lett.*, vol. 58, p. 630, 2002.

[67] Moreno, Y., Pastor-Satorras, R., Vázquez, A., and Vespignani, A., "Critical load and congestion instabilities in scale-free networks," *Europhys. Lett.*, vol. 62, p. 292, 2003.

[68] Motter, A. E., "Cascade control and defense in complex networks," *Physical Review Letters*, vol. 93, no. 9, 2004.

[69] Motter, A. E. and Lai, Y.-C., "Cascade-based attacks on complex networks," *Phys. Rev. E*, vol. 66, 2002.

[70] Nagel, K. and Paczuski, M., "Emergent traffic jams," *Phys. Rev. E*, vol. 51, p. 2909, 1995.

[71] Nagel, K. and Schreckenberg, M., "A cellular automaton model for freeway traffic," *J. Phys. I France*, vol. 2, p. 2221, 1992.

[72] Nakagawa, T., *Shock and Damage Models in Reliability Theory*. Springer, London, 2007.

[73] Newman, M., Barabási, A.-L., and Watts, D. J., *The structure and dynamics of networks*. Princeton University Press, 2011.

[74] Ohira, T. and Sawatari, R., "Phase transition in a computer network traffic model," *Phys Rev. E*, vol. 58, p. 193, 1998.

[75] Paczuski, M., Maslov, S., and Bak, P., "Avalanche dynamics in evolution, growth, and depinning models," *Phys. Rev. E*, vol. 53, p. 414, January 1996.

[76] Pastor-Satorras, R., Vázquez, A., and Vespignani, A., "Dynamical and correlation properties of the internet," *Phys. Rev. Lett.*, vol. 87, p. 258701, 2001.

[77] Pastor-Satorras, R., Vázquez, A., and Vespignani, A., "Topology, hierarchy, and correlations in internet graphs," *Lect. Notes Phys.*, vol. 650, p. 425, 2004.

[78] Pastor-Satorras, R. and Vespignani, A., *Evolution and Structure of the Internet: A Statistical Physics Approach*. Cambridge University Press, Cambridge, 2004.

[79] Paxson, V. and Floyd, S., "Wide-area traffic: The failure of poisson modeling," *IEEE/ACM Trans. Networking*, vol. 3, p. 226, 1995.

[80] Phoenix, S. L., "The asymptotic time to failure of a mechanical system of parallel members," *SIAM J. Appl. Math.*, vol. 34, no. 2, p. 227, 1978.

[81] Phoenix, S. L. and Tierney, L.-J., "A statistical model for the time dependent failure of unidirectional composite materials under local elastic load-sharing among fibers," *Eng. Frac. Mech.*, vol. 18, no. 1, p. 193, 1983.

[82] Pidaparti, R., Fang, L., and Palakal, M., "Computational simulation of multi-pit corrosion process in materials," *Computational Materials Science*, vol. 41, no. 3, p. 255265, 2008.

[83] Pisarenko, V. F. and Sornette, D., "Characterization of the frequency of extreme events by the generalized pareto distribution," *Pure and Applied Geophysics*, vol. 160, no. 12, pp. 2343–2364, 2003.

[84] Rausand, M. and Høyland, A., *System Reliability Theory: Models, Statistical Methods, and Applications, 2nd Edition*. John Wiley and Sons, New York, 2004.

[85] Reeds, A. and Shepp, R., "Optimal paths for a car that goes both forwards and backwards," *Pacific Journal of Mathematics*, vol. 145, 1990.

[86] Roy, S., Sridhar, B., and Verghese, G. C., "An aggregate dynamic stochastic model for air traffic control," *5th USA/Europe ATM R&D Seminar, Paper No. 3, FAA/Eurocontrol URL: http://atm2003.eurocontrol.fr/*, June 2003 Unpublished.

[87] Schadschneider, A. and Schreckenberg, M., "Cellular automaton models and traffic flow," *J. Phys. A*, vol. 26, p. L679, 1993.

[88] Siddique, S. and Volovoi, V., "Failure mechanisms of load-sharing complex systems," *Phys. Rev. E*, vol. 89, p. 012816, January 2014.

[89] Solé, R. and Valverde, S., "Information transfer and phase transitions in a model of internet traffic," *Physica A*, vol. 289, p. 595, 2001.

[90] Sornette, D., *Critical market crashes*. Physics Reports 378, 2003.

[91] Sridhar, B., Chen, N. Y., and Ng, H. K., "An aggregate sector flow model for air traffic demand forecasting," *AIAA 9th Aviation Technology, Integration, and Operations Conference (ATIO)*, no. 7129, 2009.

[92] Sridhar, B., Grabbe, S. R., and Mukherjee, A., "Modeling and optimization in traffic flow management," *Proceedings of the IEEE*, vol. 96, no. 12, 2008.

[93] Sridhar, B., Soni, T., Sheth, K., and Chatterji, G. B., "Aggregate flow model for air-traffic management," *Journal of Guidance, Control and Dynamics*, vol. 29, no. 4, p. 992, 2006.

[94] Stanley, H. E., *Introduction to Phase Transitions and Critical Phenomena*. Oxford University Press, Oxford, 1971.

[95] Tierney, L., "Asymptotic bounds on the time to fatigue failure of bundles of fibers under local load sharing," *Advances in Applied Probability*, vol. 14, p. 95, 1982.

[96] Toland, R. H., Sanchez, R. J., Freeman, D., Chiao, T. T., and Barlow, R. E., *Stress-rupture life of Kevlar-epoxy spherical pressure vessels*. Lawrence Livermore Laboratory Reports UCID-17755 Parts l-3, 1978-79.

[97] Turcotte, D. L. and Malamud, B. D., "Landslides, forest fires, and earthquakes: examples of self-organized critical behavior," *Physica A 340*, p. 580589, 2004.

[98] Valverde, S. and Solé, R., "Self-organized critical traffic in parallel computer networks," *Physica A*, vol. 312, p. 636, 2002.

[99] Vázquez, A., "Growing network with local rules: Preferential attachment, clustering hierarchy, and degree correlations," *Phys. Rev. E*, vol. 67, p. 056104, 2003.

[100] Vinh, N. X., *Flight Mechanics of High-Performance Aircraft*. Cambridge University Press, 1993.

[101] Volovoi, V., "Universal failure model for multi-unit systems with shared functionality," *Reliability Engineering and System Safety*, vol. 119, pp. 141–149, November 2013.

[102] Watts, D. and Strogatz, S., "Collective dynamics of 'small-world' networks," *Nature*, vol. 393, p. 440, 1998.

[103] Weber, N. A., "Dimorphism in the african oecophylla worker and an anomaly (hym.: Formicidae)," *Annals of the Entomological Society of America*, vol. 39, pp. 7–10, March 1946.

[104] West, D. B., *Introduction to Graph Theory*. Prentice-Hall, Englewood Cliffs, NJ, 1995.

[105] Wilensky, U., "Netlogo. http://ccl.northwestern.edu/netlogo," *Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL, USA*, 1999.

[106] Wolfram, S., *Cellular Automata and Complexity*. Addison-Wesley, Reading, MA, 1994.

[107] Zeng, H., Pukkala, T., Peltola, H., and Kellomaki, S., "Optimization of irregular-grid cellular automata and application in risk management of wind damage in forest planning," *Canadian Journal of Forest Research*, vol. 40, no. 6, pp. 1064–1075, 2010.

[108] Zhang, J., Song, N., Li, X., Ren, Z., and Wen, B., "Investigating morphology evolution of damage by a cellular automaton modelling," *Key Engineering Materials*, vol. 353-358, pp. 1060–1063, 2007.