Contents

List of Figures vi
List of Tables ix
List of Symbols x
Dedication xi
Acknowledgments xii

1 Introduction 1

1.1 Delay and Disruption Tolerant Network (DTN) scenarios . . . . . . 5
1.1.1 Deep Space Communication ........................................ 6
1.1.2 Mobile Ad Hoc Networks (MANETs) with no topology infor-
mation ................................................................. 6
1.1.3 Vehicular Networks ................................................ 7
1.1.4 Battle Field Networks ............................................. 8
1.2 Related Work ............................................................ 8
1.2.1 Markov based models for analyzing DTN ..................... 11
1.2.2 Mobility based models ............................................ 13
1.2.3 DTN routing based models .................................... 16
1.3 Problem Statement ..................................................... 19
1.4 Dissertation Contribution ........................................ 20
1.5 Organization of Dissertation ................................... 21

2 Theoretical Foundation .............................................. 22
  2.1 M/M/1 queue ...................................................... 22
  2.2 Open Queuing Networks ......................................... 23
  2.3 Kleinrock Independence Approximation ....................... 24
  2.4 Jackson’s Theorem ............................................... 25
  2.5 Long term behavior of link in DTN ............................ 30
  2.6 Performance Evaluation ......................................... 31
    2.6.1 Traces ...................................................... 31
    2.6.2 Simulation Tools ........................................... 39

3 Mean Time To Failure and Mean Time To Recovery for Predictable
   Delay Tolerant Networks (PDTNs) ................................ 43
  3.1 The Model ......................................................... 44
    3.1.1 Link Availability ........................................... 44
    3.1.2 Link Availability Analysis ................................ 46
  3.2 Mttf and Mttr for Jackson’s Network .......................... 48
  3.3 Simulation and Numerical Results ............................... 51
  3.4 Summary ......................................................... 54

4 Disruptive Link Modeling and Delay Analysis in Delay Tolerant Net-
   works ........................................................................ 56
  4.1 Modeling a DTN Link .............................................. 57
    4.1.1 Non-Disruptive Links ...................................... 57
    4.1.2 Disruptive Links ............................................. 59
    4.1.3 Delay Variance; Jitter ..................................... 69
  4.2 Network of Disruptive Queues .................................. 70
# List of Figures

1.1 Classification of models ................................................. 9  
1.2 Overview of our research contribution ................................. 20  
2.1 An M/M/1 queuing node .................................................. 22  
2.2 Merging of different traffic streams at link $ij$ ....................... 24  
2.3 The effect of exogenous and indigenous traffic that form aggregate traffic on node $j$ ................................................... 26  
2.4 An inter-planetary network (a) and its corresponding dynamic graph (b) ................................................................. 29  
2.5 Time series snapshots of evolving topology ............................ 31  
2.6 CDF of inter-contact periods for vehicular networks [1] ............... 34  
2.7 CDF of contact periods for vehicular networks [1] ..................... 34  
2.8 Contact periods distribution for vehicular networks [1] ............... 35  
2.9 Inter-contact periods distribution for vehicular networks [1] ........... 35  
2.10 CDF of inter-contact periods for iMote devices [2] .................... 36  
2.11 CDF of contact periods for iMote devices [2] .......................... 36  
2.12 Contact periods distribution for iMote devices [2] .................... 37  
2.13 Inter-contact periods distribution for iMote devices [2] ............... 37  
2.14 CDF of inter-contact periods for MIT reality [3] ...................... 38  
2.15 CDF of contact periods for MIT reality [3] ........................... 38  
2.16 Contact periods distribution for MIT reality [3] ...................... 39
2.17 Inter-contact periods distribution for MIT reality [3] . . . . . . . . . . 39
3.1 Four consecutive snapshots of the network connectivity due to mobility. 45
3.2 Network connection availability graph. . . . . . . . . . . . . . . . . . 45
3.3 Maximum spanning tree corresponding to link availability in Figure 3.2. 45
3.4 A representation of a sequence of ON-OFF periods and renewal process
\[ X_i \] . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 46
3.5 State transition of a link \( j \) . . . . . . . . . . . . . . . . . . . . . . 48
3.6 NASA 6 node topology as in [4] . . . . . . . . . . . . . . . . . . . . . 52
3.7 A generic tree topology . . . . . . . . . . . . . . . . . . . . . . . . . . 52
3.8 A network of EDSN nano satellites [5] . . . . . . . . . . . . . . . . . 52
3.9 end-to-end delay analysis of NASA 6 node topology (Figure 3.6) . . 53
3.10 end-to-end delay analysis EDSN satellites (Figure 3.8) . . . . . . . . 54
3.11 end-to-end delay analysis of tree topology (Figure 3.7) . . . . . . . . 54
4.1 Transition-rate diagram for the \( M/M/1 \) model. . . . . . . . . . . . 58
4.2 A representation of a sequence of OFF-ON periods. . . . . . . . . . . 60
4.3 States of link connectivity. . . . . . . . . . . . . . . . . . . . . . . . . 60
4.4 Transition rate diagram with time dependent breakdowns. . . . . . 61
4.5 Delay performance for various values of \( \alpha \) and \( \beta \) . . . . . . . 69
4.6 EDSN nano satellites network [5]. . . . . . . . . . . . . . . . . . . . . 71
4.7 EDSN delay comparison for various \( \alpha \) and \( \beta \) values. . . . . . . 74
4.8 A sink tree network with disruptive links. . . . . . . . . . . . . . . . . 74
4.9 Delay comparison for the tree network in Figure 4.8 for various values
of \( \alpha \) and \( \beta \). . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 78
4.10 A mesh network. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 79
4.11 Delay comparison for the mesh network in Figure 4.10 for various values
of \( \alpha \) and \( \beta \). . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 81
5.1 State transition of a link. . . . . . . . . . . . . . . . . . . . . . . . . . 85
5.2 (a) and (b) are connected states and (c) is a disconnected state. . . 85
5.3 State transition diagram. .......................................................... 86
5.4 end-to-end (E2E) delay analysis of four node topology. .............. 92
5.5 Multiple Spacecrafts as in [4] .................................................... 93
5.6 Spacecraft Topology similar to [6] ........................................... 93
5.7 E2E delay analysis of Nasa spacecraft topologies ....................... 93
List of Tables

5.1 Connected and disrupted states .................................. 86
5.2 Mean Connection Time ($m_t$) ................................... 91
List of Symbols

\begin{itemize}
\item \( \lambda \) packet or bundle arrival rate
\item \( \mu \) packer or bundle service rate
\item \( \gamma \) local traffic generated at a node
\item \( \alpha \) exponential distribution parameter for link ON(connection) period
\item \( \beta \) exponential distribution parameter for link OFF(disconnected) period
\item \( \mathcal{R} \) DTN routing algorithm
\item \( \pi_i \) probability of being in state \( i \)
\item \( M/M/1 \) Kendall’s notation for a queue with markov arrival rate, markov service rate and a single server
\end{itemize}
Dedication

I would like to dedicate my research work to my parents, Mr Yashpal Sehgal and Mrs Sharda Sehgal and my wife Dr. Meghana Sehgal for their constant support during my research endeavor.
Acknowledgments

I would like to thank Dr Hassan Peyravi for all his guidance, encouragement, and valuable comments during the progress of my dissertation. Under his guidance it was a great learning and research experience.
Chapter 1

Introduction

In today’s world, omnipresence of the Internet in different aspects of life is quiet evident. Mobile devices, personnel computers, web servers, data centers, cloud servers, etc. interact with each other and share information using the Internet. Today, when we think of the Internet we make an assumption that it is a continuous service which provides an end-to-end connection between different remote devices which want to communicate with each other. From networking perspective, the Internet comprises a set homogeneous and/or heterogeneous networks which use different communication protocols to communicate with each other. The Transport Control Protocol (TCP)/Internet Protocol (IP) is the most common protocol used for the Internet. There exist other protocols for various internet applications such as for email applications we have Simple Mail Transfer Protocol (SMTP), for file transfer applications we have File Transfer Protocol (FTP) and for secure file transfer we use Secure File Transfer Protocol (SFTP). For interaction between logical links(hyper links) or hypermedia we use Hypertext Transfer Protocol (HTTP). All these protocols make a fundamental assumption that at all time there exists an E2E path between a source and a destination, potentially with very low latency (delay) to send/receive data.

Apart from wired devices on internet, there also exist different wireless devices e.g.
cell phones, iPads and other smart devices. Wireless Local Area Network (WLAN) gives users with mobile device the power to go anywhere and still have access to the Internet. These networks have a combination of stationary or fixed devices and a small set of mobile devices which are connected to the fixed devices and disconnectivity is for very short period of time. However, these networks still have topologies with minimal changes and a fixed infrastructure where getting an E2E path between two devices is not difficult and the incurred delay can be predicted, and an E2E communication can be provided efficiently. In wireless networks there exist a class of networks, known as Mobile Ad Hoc Networks (MANETs) in which two devices are connected if they are within the communication range of each other. It is also possible that between any two devices there may exist multiple intermediate communication links(connections) and a link may switch between ON and OFF periods. This may happen in power-limited mobile devices due to scheduled ON and OFF periods, or due to mobility in MANETs which consist of mobile devices. In MANETs the devices update their communication paths, also known as routing paths, based on topological changes and the available paths. A routing decision is made to forward the information (message or packet) to the next device. These routing decisions or algorithms assume that there exists an E2E path between a source and a destination and path recalculation is predictable based on which a reliable communication can be provided. However, when we consider satellite communications where two satellites can communicate only when they are within the line-of-sight of each other, or in case of mobile devices, where an E2E path doesn’t exist at all time due to mobility, the existence of these paths cannot be predicted as in MANETs. Similarly, certain links may goes down in a network due to some malicious activities that can make the network partitioned in such a way that we can’t use the assumptions of glsE2E connectivity at all times to predict the latency(delay) between a pair of nodes. In such scenarios, predicting an E2E delay with non-deterministic path connectivity between a pair of nodes in the network becomes a challenging problem. To address these challenges and limitations
for existing communication protocols a new overlay network was developed [7, 8] which provides a mechanism to store packets for a longer period of time and then forward it once a connection is restored. This overlay network supports longer network disconnectivity and tolerate long delays. The networks which implements this overlay network are called Delay and Disruption Tolerant Networks (DTNs) [7,8].

Generaly, DTNs are sparsely connected networks where an E2E path between any two nodes doesn’t exist at all time and connection restoration between two nodes is unpredictable. DTNs include networks with intermittent connectivity (link failure or link disruption), large and often variable delays, and high bit error rates [8]. The link intermittency can be caused due to mobility, scheduled link ON/OFF periods to conserve energy, natural disasters, or malicious attacks on some part of the network infrastructure. Due to these limiting conditions, E2E connectivity does not exist at all time and messages are routed in a store-carry-and-forward fashion. While the demand for DTNs initially emerged in the area of space communication networks [7, 8], new applications of DTNs have recently emerged in other areas of networking including battle field networks, MANETs without topology information, animal tracking system with sensors, underwater communication with high attenuation of radio waves, rural networking, environmental monitoring networks, and vehicular networks. All these networks share the same major characteristic which is link intermittency. Link intermittency can be periodic (deterministic) or sporadic (non-deterministic). In sensor networks, transmitters can be turned OFF and ON to conserve energy in a predictable manner. Link disconnectivity (periodic or sporadic) occurs in deep space communication networks or in interplanetary explorations. In periodic link disconnectivity, it may happen that only a particular set of links switch between ON and OFF periods. In satellite communications this phenomenon is very common [4,6]. In such scenarios, the topology becomes predictable with intermittent link connectivity. Also in sensor networks in which a set of links goes ON and OFF by scheduling will provide a set of predictable topology. In networks with node mobility, link disruptions can occur for
various reasons including the impact of long/short or fast/slow fading phenomenon. Under these circumstances, the delay incurred between communicating nodes can be very large and unpredictable. The two important delay characteristics in DTNs are contact opportunity time and inter-contact time. The contact opportunity between two nodes is described by a period during which two nodes are within the transmission range of each other. The inter-contact time between two nodes is defined as a duration between two consecutive contact times. In mobile networks, the contact time and inter-contact times are dependent on the underlying mobility model. The distribution of these two time intervals significantly impact the point-to-point and the incurred end-to-end delays. Studies [9] have shown that for random way-point [10] and the random direction [11], mobility models exhibit an exponentially distributed inter-contact times with small scale mobility. For large scale mobility, experiments have shown that the tail of the inter-contact time distribution follows a power law decay in some finite range, but exhibits an exponential decay afterward [12].

Due to lack of E2E path between any pair of nodes, which is caused by the dynamic nature of the topology, routing becomes very challenging in DTN compared to networks with fixed topology. These challenges and limitations resulted in a new network architecture. In DTN architecture [7], the bundle layer is designed to complement the transport layer deficiencies, and its role is to provide reliable delivery under link intermittency through a routing algorithm that could be based on flooding packets or forwarding. In some DTN routing protocols, such as opportunity-based schemes [13], messages are randomly forwarded hop by hop with the expectation of eventual delivery. Other routing algorithms make a relay selection probabilistically by estimating relative successful delivery probability. Such delivery probability could be based on expected delay on each link or other performance metrics. The probabilistic forwarding scheme in [14], which is based on optimal stopping rule problem, maximizes the expected delivery rate while satisfying a certain constant on the number of forwarding per message.
The transmission of a bundle in DTNs can be disrupted due to a short contact time. In that case, the bundle can be partially forwarded if disruption occurs at the middle of the bundle transmission. However, a partial bundle cannot be forwarded to the subsequent node without being received in its entirety. Several factors contribute to an end-to-end delay a packet suffers including transmission delay (bit-rate), queuing delay which is a function of packet arrival rate, propagation delay which is a function of distance, and access delay that depends on link availability. While the transmission delay and propagation delay can be deterministic or bounded, queuing delay and access delay are mostly probabilistic. While several routing algorithms such as epidemic [15], spray-and-wait [13], delegation forwarding [16], and optimal probabilistic forwarding [14] offer routing strategies and produce a routing matrix, the proposed models in our work are more general in the sense that it can accommodate any routing protocol as long as the network relies on the assumption that node mobility exhibits a long-term steady state or regularity on link disruption probability, and the underlying algorithm produces a long-term probabilistic routing matrix.

1.1 DTN scenarios

A lot of real life network scenarios possess DTN properties in terms of link disruption and mobility [17]. In following sections we have discussed different DTN scenarios and different areas where DTN architecture can be used.

1.1.1 Deep Space Communication

As mentioned earlier presently, when we think about the Internet on earth we make an assumption (un)knowingly that an information network that is always interconnected or in other words its always on and has predictable latency. These assumptions need to be relaxed when we consider communication in space. Planets and satellites
orbit, and they are not always aligned in such a fashion that data transmission can occur immediately. The ability to send and receive data is disrupted. Information processing nodes, satellites or ground stations, need to be able to store the data that they receive until they are able to safely send it to the next node in the network. National Aeronautics and Space Administration (NASA) initiated the idea of internet in space. Space Communications and Navigation (SCaN) at NASA is developing a set of international standards, collectively referred to as Disruption Tolerant Networking (DTN) standards, to support internetworking in space. The DTN standards support a network service (similar to the Internet Protocol (IP)), reliability (similar to Transmission Control Protocol (TCP), but implemented very differently), and security. These are all designed to work in environments where end-to-end paths may not be available, such as when an orbiter needs to receive data from Earth and then wait, before it can forward it to a lander on another planet. The wait time also known as delay is an important performance parameter.

1.1.2 MANETs with no topology information

MANET is defined as infrastructure-less network of mobile devices which is continuously self-configuring. The mobile devices or nodes forward packets to other nodes based upon underlying routing algorithm which incorporates the dynamic nature of the topology. Maintaining route information in such scenario is very challenging. But different routing algorithms proposed for MANET, such as, Table-driven (proactive) (e.g. OLSR [18] and Babel [19]), On-demand (reactive) (e.g. AODV [20] and DSR [21]), Hybrid (both proactive and reactive) (e.g. ZHLS [22]) and Hierarchical routing protocols (e.g. FSR [23], CBRP [24]) make a fundamental assumption that when node wants to forward a packet end-to-end path exists between a source and destination. The only challenge in such routing approach is how fast a particular routing algorithm updates routing information to address dynamic nature of the
network. But when we consider scenarios in which this fundamental assumption of E2E path is violated then routing becomes very challenging. Thus, in a sparse and disruptive networking scenario such as sensor networks used for wildlife tracking and environmental monitoring or mobile devices used in battlefields to route a packet from a source to a destination requires different routing protocols.

1.1.3 Vehicular Networks

Recently, Vehicular DTN (VDTN) have become a popular research topic. In VDTN, the traffic sources and sinks do not have a direct transmission path among each other. However, a vehicle is used as a carrier to carry the data from traffic sources to sinks. When a vehicle moves into the transmission range of traffic source, the roadside unit of traffic source transmits the data to a mobile router deployed in that vehicle. This vehicle can travel among places and once it moves into the transmission range of the sink, a mobile router transmits data in its buffer to the roadside unit of the sink [25]. These networks help in effective traffic management, improve road safety and to provide more comfort for drivers and conductors of public transportation. Cars equipped with wireless devices can exchange traffic and road safety information with nearby cars and/or roadside units. Inter-vehicle communication (IVC) can increase the safety, efficiency, and convenience of transportation systems involving planes, trains, automobiles, and robots [26]. Due to the high speed and mobility of vehicles the topology is very dynamic and the short range of inter vehicular communication results in frequent disconnections. This disconnection duration increases when traffic density is low, as in the case of VDTNs. This property makes the DTN concepts attractive, as they were designed to deal with such network conditions.
1.1.4 Battle Field Networks

Battle Field Networks, those connect troops, aircraft and all other military logistic equipment, posses frequent disconnection of mobile and sensor devices due to its highly dynamic nature. In some scenarios the mobility of devices may posses semi-deterministic mobility model properties. Because in battlefields troops moves in a particular pattern or may follow Column movement model.

1.2 Related Work

In disrupted or failure prone systems delay analysis has been considered as an important performance parameter to evaluate a particular system. It gives an estimation about the expected time of completion of a task, that is, on the average how long it takes to complete a particular task when one or mor components fail. Alternatively, we can also estimate on the average how long a task has to wait before service started. Delay analysis have been studied extensively in modeling different systems including combinational logic [27], embedded systems [28], process analysis in manufacturing industry [29], etc. Other important performance aspects of disrupted systems include workload analysis, conformance checking, path analysis, etc.

Several models have been developed in the literature to measure various performance aspects of DTNs. There are models which capture the transient nature of different mobility models. They analyze opportunistic connectivity provided by the underlying mobility model and different states the network goes through to evaluate closed form expressions which describe the convergence behavior of DTNs. Once a closed form expression is evaluated, these models can be used to estimate the average E2E delay, delay variance, compare DTN performance of different routing protocols, measure performance of buffer management algorithms in DTN, content placement in interrupted environments, etc. Figure 1.1 shows the classification of our literature
survey and different models proposed for DTNs.

Figure 1.1: Classification of models

In the literature most of the proposed models study mobility properties and build their model based on mobility. Thus, mobility becomes an important aspect of DTNs performance. Mobility models can be classified into (i) deterministic models (e.g., satellites in space, urban traffic models, scheduled ON/OFF period of the links in fixed topology), (ii) semi-deterministic models (e.g., Column model, Pursue models), and (iii) random models (e.g., Brownian Motion and Random Waypoint models) [30]. Mobility can change the state of the routing matrix $R$ deterministically or probabilistically as defined by one of the above mobility models. Whether the mobility model is deterministic or probabilistic, the authors observe the long term behavior of different links in DTN and under reasonable assumptions build their models to measure different performance parameters. These include message dissemination delay, latency, optimal content placement, correlation between packet size and link availability, etc.
All these models make a fundamental assumption that eventually we get a path from a source to destination over a long period of time and link intermittency is transient.

Some models study different routing algorithms to analyze intermediate message spreading delay also known as message dissemination delay in DTN. Most of these models study flooding based routing such as *Epidemic routing* [15]. A flooding based routing algorithm gives a lower bound on flooding time (a.k.a diameter in dynamic network) [31]. In the flooding mechanism, every informed node (i.e., any node that has the source message) always sends the source message to all its neighbors. So, the source is clearly informed since the beginning. Any other node $v$ gets informed at time step $t$ if any of its neighbors is informed at time step $t-1$. The completion time of the flooding mechanism (or, simply, *flooding time*) is the first time step in which all nodes of the network are informed. It is important to observe that flooding time in dynamic networks plays the same role of *diameter* in static networks. Flooding time represents lower bounds for broadcast protocols in DTNs.

On the contrary there are routing algorithms such as *Spary and Wait* [13] and its different variants such as *Source Spray and Wait* and *Binary Spray and Wait* [13] which spread limited number of copies of a message in network and still achieve better performance in terms of reduced queuing delay, throughput and energy efficiency. It combines the speed of *Epidemic routing* with the simplicity and thriftiness of *direct transmission*. In [32] delay distribution for *Spary and Wait* has been analyzed. In following sections we discuss different models proposed for DTNs and their pros and limitations.

### 1.2.1 Markov based models for analyzing DTN

Given that topology in DTNs is not fixed, links which are available at time $t$ may not be available at time $t+\Delta t$. In many DTN scenarios, the long-term topological changes may converge to a predictable behavior that can be represented by a distribution.
In [33], the authors compare the stochastic properties of different mobility models in MANETs based on the number of states visited in a fixed time, the time to visit every state in a region, the first passage time, the full coverage time, and the transient behavior. They derive a model to get the transient probability distributions and First Passage Times for 1-dimensional correlated random walks. Then they used experiments to study these stochastic properties for different mobility models. This work provides insight into the dynamic nature of the topology and experimental results shows exponential behavior of the contact and inter-contact distributions of various links.

In [34], the authors capture various snapshots of mobility to obtain the Ad-Hoc network topology architecture at any moment. They proved experimentally that by analyzing the Ad-Hoc network topology snapshots that some dynamic characteristic parameters of Ad-Hoc network, such as, the number of network topology in steady state or unsteady state appearing during a certain time, as well as the time duration of network topology in steady state or unsteady state, could be obtained statistically. Furthermore, they showed that the probability of the network topology invariability and variability event could be predicated by adopting the discrete time and continuous time Markov process.

In [25], a queueing model for efficient buffer management and resource sharing for a mobile router in Vehicular Delay Tolerant Network (VDTN) is proposed. The VDTN scenario that authors have considered for their model consists of a set of mobile nodes (mobile routers) which visit different points, each of these points are either traffic source or sink. The traffic sources and sinks do not have a direct transmission path among each other. However, a vehicle is used as a carrier to carry the data from traffic sources to sinks. When a vehicle moves into the transmission range of traffic source, the roadside unit of traffic source transmits the data to a mobile router deployed in that vehicle. This vehicle can travel among places and once it moves into the transmission range of the sink, a mobile router transmits data in its buffer
to the roadside unit of the sink. The queueing model proposed in [25] is used as a tool to study the behavior of the traffic source in a competitive environment which is due to the fact that the transmission resources of a mobile router are shared among multiple traffic sources. Therefore, the traffic sources non-cooperatively optimize their transmission strategies to achieve the highest utility. Also, in the proposed model, various performance measures of the mobile router to forward data from the traffic sources to the sink can be obtained. In addition, since the data from multiple traffic sources share the resources (i.e., buffer and transmission time) of a mobile router, the transmission parameters (i.e., transmission rate or transmission probability) of one traffic source will affect the performance and hence the utility of other traffic sources (i.e., opponents). The proposed model also measure delay and throughput of VDTN with shared resource system. However, the model doesn’t consider the impact of contact and inter-contact time along with traffic load while evaluating E2E delay. Also, the scenario considered for the model focuses on a specific type DTN.

In [35], an edge-markovian based model for DTN is proposed. In this model, edges come up or down at the beginning of each time step, but the topology then remains static until the next time step. Edge-Markovian dynamic random graphs were first introduced in [31] as a generalization of time-independent dynamic random graphs to capture the strong dependence between the existence of an edge at a given time step and its existence at the previous time step. This model focuses on understanding the intermediate nature of message spreading in a DTN scenario.

In [36], the delay performance of a small mobile ad hoc network has been analyzed by a tandem queuing system, in which the queue is modeled as $M/G/1$ queue. The authors present an exact packet-level analysis but due to the state-space expansion, the analysis cannot efficiently be applied for all model parameter settings. As a result, an analytical approximation was constructed and validated.

In [37], a model has been developed in which different pairwise contact rates are supported by time-varying movement parameters.
In [38], the authors developed a model for $M/M/1$ queues with break down and customer discouragement. Customer discouragement means at the time of a breakdown customer may become discouraged and leave the system with a constant probability, independently of other customers. The system alternates between working and repair periods. Formulas are found for the expected queue size at the end of a working-repair cycle.

1.2.2 Mobility based models

In [39], an insight into deep space topology formation is provided. The authors, specified the topology formation in terms of the relative satellite positions and absolute satellite orientations. The redundancy in the relative position specification generates a family of control topologies with equivalent stability and reference tracking performance. This work provides good foundation for estimating satellite topology formation and proves that we can effectively calculate relative positions of satellites in space.

In [40], the Little’s theorem has been used to develop an efficient queue management technique for satellite DTN. For the performance metrics, the model tries to achieve two goals: (i) increasing DTN device throughput via efficient link exploitation, and (ii) increasing application satisfaction. There model consists of a primary Frist In First Out (FIFO) queue (Connectivity buffer) and a secondary supportive queue (Non-Connectivity buffer), which serves high-priority bundles. Given the high propagation delay, and furthermore, the potentially very high storage delay involved in typical Space applications. Therefore, a Priority Queue (PQ) or a PQ-derived scheduling model for incoming packets does not present conceptually a tempting approach, since it may fail with long-stored packets. So, their approach departs from a FIFO scheme, and therefore, calls for a straightforward comparison with a typical FIFO scheme. However, they extend their stochastic analysis also for PQ, which they
consider as a theoretical upper bound (only) when connectivity is always present.

In [41], long-term behavior of predictable delay tolerant networks has been analyzed based on spatial, temporal and connection probability information to solve topology control problem. In [42], statistical properties of node mobility have been studied through simulations. The impact of node mobility on the performance of routing methods in DTNs have been analyzed.

In [43], the Localized Random Walk (LRAW) is used to model DTNs. In LRAW, each node is assigned a fixed Cell and makes a list of all of its neighbors at each time slot $\tau$. A node moves from its current location with a fixed probability $s$, and selects one of the neighboring cells with probability $p$. The model offers an approximation for average message delay. However, the model does not consider the effect of message arrival rate or traffic load.

Heterogeneous mobility based models incorporates dynamic nature of the topology and makes analysis and routing more challenging. In [44], a two-hop network-coded model in which a pair of source and destination nodes move independently in a square area under a grid-based random walk mobility model has been analyzed. The model iteratively compute the cumulative distribution function of the block delivery delay. In [45], an analytical framework, using a time-homogeneous discrete-time Markov chain, has been developed to predict the performance for utility-based algorithms with general heterogeneous mobility. The model combines mobility properties with actions determined by the DTN algorithm to compute the transition probabilities. This model maps an optimization problem into a Markov chain, where each state (e.g. assignment of content replicas to nodes) is a potential solution. The model incorporates both single copy and multi copy algorithms for unicast routing and content placement. The accuracy of the predictions performance has been measured against simulations over a range of synthetic and mobility traces. The model decouples DTN algorithms effect from mobility, but allows one to derive performance metrics (convergence delay, delivery probability) using transient analysis of the Markov chain.
The model, however, doesn’t incorporate multicast routing algorithm and affect of traffic density on end-to-end delay in DTN under heterogeneous mobility. The model presented in this chapter is quite simple and gives an excellent performance in terms of long-term behavior of the network.

The message propagation has been estimated in [46]. It gives a detailed expression of average information dissemination delay based on message size, social dynamics and number of nodes in the network. The model assumes both the contact time and inter-contact period are exponentially distributed. They apply exponential distribution both to contact and inter contact periods to quantify the impact of message size as larger messages usually require longer contact period to propagate. The model considers other social characteristics including distinct inter-contact periods, as various people might exhibit totally different behaviors owning to their working, living place and friends circle. It divides applications into different groups according to their communication intervals with each other. Sometimes people would be unwilling to forward a message to others due to energy and storage constrain or prefer to forward information to people in the same group with them. The two kinds of activities are called individual selfishness and social selfishness respectively. The model concentrates on social selfishness which has a considerable effect on communication between groups. This model shows that as the number of nodes increases delay decreases. This approach doesn’t relate delay to traffic density or routing scheme. [47], eraser based coding technique is used for sending messages between source and destination. Random set of links are analyzed for contact-time and inter-contact time distribution.

1.2.3 DTN routing based models

In routing based models, DTN is analyzed from routing perspective. E2E and throughput become the basic parameters to measure performance of a given routing protocol. Some literature emphasize on a particular routing algorithm, flooding
based or forwarding based. In particular, the DTN routing problem has many input variables such as dynamic topology characteristics and traffic demand. Complete knowledge of these variables facilitates the computation of optimal routes. However, with partial knowledge, the ability to compute optimal routes is hampered, and the performance of the resultant routing is expected to be inferior. To understand this fundamental trade-off between performance and knowledge, a set of abstract knowledge oracles have been proposed in [48]. Each of these oracles are able to answer questions we ask of them. In [48], the authors propose routing evaluating framework based upon knowledge of topology and compare their performance. Knowledge Ora-
cles which are notational elements used to encapsulate particular knowledge about the 

network required by different algorithms. These oracles provide information regarding contacts, instantaneous buffer occupancies (queuing) at any node at any time and traffic demand, that is, information regarding the present or future traffic demand. It is able to provide the set of messages injected into the system at any time. The oracles listed by the authors are:

**Contacts Summary Oracle:** This oracle can answer questions about aggregate statistics of the contacts. In particular, the contacts summary oracle provides the average waiting time until the next contact for an edge. Thus, the contacts summary oracle can only respond with time-invariant or summary characteristics about contacts.

**Contacts Oracle:** This oracle can answer any question regarding contacts between two nodes at any point in time. This is equivalent to knowing the time-varying DTN multi-graph. The contacts summary oracle can be constructed using the contacts oracle, but not vice versa.

**Queuing Oracle:** This oracle gives information about instantaneous buffer occupancies (queuing) at any node at any time, and can be used to route around congested nodes. Unlike the other oracles, the queuing oracle is affected by
both new messages arriving in the system and the choices made by the routing algorithm itself. We expect it to be the most difficult oracle to realize in a distributed system.

**Traffic Demand Oracle:** This oracle can answer any question regarding the present or future traffic demand. It is able to provide the set of messages injected into the system at any time.

The routing algorithms may use or not use these oracles for making routing decisions. Based on usage of these oracles they are divided into three classes. These algorithms do not utilize any oracles and obviously they perform poorly. This class consists of algorithms that utilize all the oracles (contacts, queuing and traffic demand). These assumptions are far too strong to operate in a widely distributed, dynamic routing environment envisioned by DTNs, and

These algorithms route in the absence of the traffic demand oracle and use one or more of the other oracles (congestion, queuing). Messages are routed independently of the future traffic demand. This is a more practical assumption from an implementation perspective.

A taxonomy of various routing algorithms with some basic analysis has been provided in [49] with emphasis on opportunistic approaches to DTN routing. [50] and [51] compare single-copy and multi-copy based routing algorithms respectively. For single-copy case, direct comparison, first-contact and utility-based routing with transitivity are compared. Whereas, [51] compares flooding and forwarding based routing algorithms in which multiple copies of a message may exist at a given time in network. Upper bounds in terms of number of copies have been calculated to perform better than flooding based approaches. [52] and [53] have compared different routing protocols and their limitations both from theoretical and experimental point of view.

In [54], a delay analysis for epidemic routing has been given in which the underlying network is divided into different communities, each of which is a square with
\ell$ meters length of edges. These communities are ordered, the inter-meeting time of two nodes located in the adjacent communities are measured, and a Markovian chain model for packet spreading has been developed. The models in [35] and [55] are based on the underlying routing algorithm which is also Epidemic Routing [15]. In [35] the model is based on flooding on an edge-Markovian evolving graph in which at each time slot an edge changes its state according to a two-state Markovian process with probabilities $p$ (edge birth-rate) and $q$ (edge death-rate). It assumes finite number of nodes, finite link capacities and finite message sizes. On the other hand, [55] uses a random graph approach to gain insight into temporal nature of the epidemic routing, i.e. how a packet spread in mobile wireless networks as a function of time.

In [56] nodes estimate connectivity and expect inter-encounter time with sink nodes. Connectivity is estimated based on ratio of past and present connections. When the connectivity is unreliable, nodes delay the transmission for the remaining inter-encounter duration or per-hop lifetime. Since packets are forwarded if the connectivity reaches a reliable threshold before delay time expires, the authors shows that delivery latency is significantly reduced.

1.3 Problem Statement

Most of the models and routing protocols that we discussed previously are based upon contact and inter-contact time distribution and their joint impact on DTNs. However, queuing delay which is an important aspect of Delay and Disruption Tolerant Network (DTN) performance has not been incorporated in these models. With respect to DTN, queuing delay is the time spent by a packet in a node’s buffer (queue) before getting transmitted to next node. The delay a packet suffers at each node along the path towards a destination may vary significantly due to queue length of a queue and/or output link disruption. Thus, in order to get a close form expression for E2E delay we also calculate average queuing delay and average output link
disruption time. The average queuing delay at each node contributes to the average E2E delay as well as average jitter (delay variance) calculation in DTN. Thus, in order to provide a stronger and more complete analysis of DTN, we need to incorporate queuing delay along with packet service rate ($\mu$), traffic load ($\rho$), contact and inter-contact time distributions. The problem we are trying to solve is to approximate average E2E delay in DTNs under various traffic loads where the packet arrival process and/or service rate are disrupted.

### 1.4 Dissertation Contribution

The contribution of this research work is three-fold. First, we deploy an exponential traffic distribution model for DTN that incorporates dissemination of packets under a probabilistic routing algorithm in DTN. Second, we employ a strong, yet simple, open queuing system based on Jackson’s theorem [57] that facilitate the computation of average E2E delay and we also derive an expression for measuring jitter under various DTN network conditions. Third, for these models, we measure the performance of DTNs, that is, Average Link Availability (ALA), disruptive link and queuing delay distribution and a first contact based exhaustive state space model to incorporate any mobility model and DTN routing algorithms. All the proposed models are based upon the long-term behavior of DTNs. Figure 1.2 shows different aspects of DTN that our model incorporates.

### 1.5 Organization of Dissertation

This dissertation is organized in six chapters as follows. Chapter 2, provides the theoretical foundation which is common to the different models proposed in chapters 3, 4 and 5 respectively. We have also highlighted different assumptions for respective models and analysis while laying out their theoretical foundation. This chapter also
includes the performance evaluation section, where we have compared and confirmed analytical results with results obtained from real life DTN traces. Chapter 3 discusses Average Link Availability (ALA) based E2E delay calculation for Predictable Delay Tolerant Network (PDTN). Chapter 4 derives relationship between disruptive queues, traffic load and long-term behavior of DTN. Based on this relationship, we calculate the average E2E delay and jitter. In Chapter 5 we discuss a more generic model for evaluating performance of DTN based on transitions from bad(disconnected) to good(connected) states. This model, which is useful for relatively small networks, uses an state space approach to evaluate all states the network can go through and then evaluates the average E2E delay for entire network. Chapter 6 concludes the dissertation remarks and highlights future work and areas to explore based on our work.
Chapter 2

Theoretical Foundation

In this chapter, we introduce a theoretical foundation for our proposed models for DTN. First, we describe an $M/M/1$ queue that represents a node in a DTN. This follows by a short discussion on open queuing networks of $M/M/1$ queues and the long-term behavior of the network, that is, studying the network in the steady state. We consider different properties and limitations of open queuing networks in the context of DTN and incorporated them in building our models which are discussed in detail in following chapters.

2.1 $M/M/1$ queue

An $M/M/1$ queue refers to a queueing system where customers (packets) arrive according to a Poisson process and are served by a single server with an exponential service-time distribution, as shown in Figure 2.1 [58]. The arrival rate $\lambda$ and service
rate $\mu$ do not depend upon the number of customers(packets) in the system so they are state-independent. The notation $M/M/1$ is known as Kendall’s notation which denotes a system with Poisson arrivals, exponentially distributed service times and a single server. All the involved random variables, e.g. number of packets in the system (queue length) at any time $t$, are supposed to be independent of each other. The $M/M/1$ queue model is the most elementary of queueing models and can be studied using closed-form expression [58]. An $M/M/1$ queue can be represented as a stochastic process whose state space is $0, 1, 2, 3, \cdots$, where the value corresponds to the number of customers(packets) in the system, including any currently being served.

Arrivals occur at rate $\lambda$ and move the process from state $i$ to $i + 1$. $\mu$ is the service rate parameter with mean service time $1/\mu$. A single server serves packets one at a time from the front of the queue, according to a first-come, first-served discipline. When the service is complete the packets leaves the queue and the number of packets in the system reduces by one. The buffer is of infinite size, so there is no limit on the number of packets it can contain. We discuss the mathematical representation of $M/M/1$ queue and its disrupted variant in detail in chapter 4.

### 2.2 Open Queuing Networks

An open queuing network is characterized by one or more sources of job arrivals and correspondingly one or more sinks that absorb jobs departing from the network. Each node can be considered as a service center, which serves the requests. Thus, in a model for computer-system performance, we may have a service center for the CPU(s), a service center for each I/O channel, a node in DTN which forwards messages to different destinations. A service center may have one or more servers associated with it. In our models we have considered a single server model to serve a packet. If a job requesting service finds the server at the service center busy, it will join the queue.
associated with the center, and at a later point in time, when one of the servers becomes idle, a job from the queue will be selected for service according to some scheduling discipline. After completion of service at one service center, the job may move to another service center for further service, reenter the same service center, or leave the system. In our models we don’t consider reenter phenomenon, that is, if a packet (bundle) gets serviced at a particular node and departs, it doesn’t reenter that node in future [59].

2.3 Kleinrock Independence Approximation

Kleinrock Independence Approximation states an important property about the effect of different packet streams passing through a transmission link in a tandem queueing network which can be used to restore $M/M/1$ property of queues in the network. We explain this property as follows.

Let us consider a transmission link $(ij)$ between two nodes (queues) $i$ and $j$ in a tandem queuing network. We assume there are different packet streams. A packet stream consists of several packets with unique source and destination pairs. Packets of a stream may take different paths between a source and destination as shown in Figure 2.2. We assume several packet streams (traffic) pass through link $ij$. We also assume that packets don’t travel in a loop. Let $\lambda_s$ be the arrival rate of packet stream $s$, and let $f_{ij}(s)$ denote the fraction of the packets of the stream $s$ that go through

$$\lambda_{ij} = \lambda_{s_1} + \lambda_{s_2}$$

Figure 2.2: Merging of different traffic streams at link $ij$. 

23
The total arrival rate at link \( ij \) is given as,

\[
\Lambda_{ij} = \sum f_{ij}(s) \lambda_s
\]  

(2.1)

where summation denotes the fraction of all packet streams \( s \) crossing link \( ij \) as shown in Figure 2.2. It is proven in literature [60] and [57] that even if the packet streams are Poisson with independent packet lengths at their point of entry into the network, this property is lost after the first transmission line, that is, if traffic enter at queue \( i \) then this property is not valid for traffic arriving at queue \( j \) via link \( ij \). To resolve the dilemma, it was suggested by Kleinrock in [60] that merging several packet streams on a transmission line has an effect akin to restoring the independence of interarrival time and packet lengths. It was concluded in [60] that it is often appropriate to adopt an \( M/M/1 \) queueing model for each communication link regardless of the interaction of traffic on this link with traffic on other links. This is known as Kleinrock independence approximation and seems to be a reasonably good approximation for systems involving Poisson stream arrivals at the entry points, packet lengths that are nearly exponentially distributed and moderate-to-heavy traffic load. Based on this approximation we discuss a close form expression for average E2E delay calculation for an open queueing network in section 2.4.

2.4 Jackson’s Theorem

The foundation of our proposed models is based upon interconnection of tandem open queuing systems in which packets move from one queue to the next queue under a routing algorithm \( R \) that produces an asymptotic routing matrix \( R \) over a long period of time. Consider a partial structure of the network model in Figure 2.3, where \( \gamma_j \) is the cumulative rate of exogenous (external) traffic flows entering node \( j \), \( \lambda_j \) is the aggregate traffic, both exogenous and indigenous (relay) flows, arriving to node \( j \). \( r_{ij} \)
is the probability of a packet routed from node $i$ to node $j$ under routing algorithm $\mathcal{R}$. The network forms an open tandem queuing model in which each node has a FIFO

![Figure 2.3: The effect of exogenous and indigenous traffic that form aggregate traffic on node $j$.](image)

buffer. After a packet is transmitted by a node, it may move to another node, or leave the system completely, as discussed in 2.2. In an open queuing network, packets enter and depart from the network. In tandem queuing systems, the arrival times to the receiving node are strongly correlated with departure time from the preceding nodes. There exists no analytical results for such networks in which inter-arrival and service times are dependent. However, Kleinrock independence approximation, as discussed in section 2.2 shows that merging several packet streams on a transmission line has an effect akin to restoring the independence of inter-arrival times and packet lengths, thus an $M/M/1$ queuing model can be used to analyze the behavior of each communication link.

A Jackson queuing network [57] is a network of an $n M/M/n$ state-independent queuing system with the following features. (i) There is only one class of packets arriving to the system. (ii) Exogenous packets arrive at node $j$ according to a Poisson process with rate $\gamma_j \geq 0$. (iii) The service times of the packets at $j$th queue are exponentially distributed with mean $1/\mu_j$. Upon receiving its service at node $i$, the packet will proceed to node $j$ with a probability $r_{ij}$ or leave the network at node $i$ with probability $(1 - \sum_{j=1}^{n} r_{ij})$. Finally, the queue capacity at each node is assumed to be infinite, so there is no packet dropping.

Let $\mathbf{R}$ be the $n \times n$ probability matrix describing the routing of packets within a Jackson network [57], $\lambda = (\lambda_1, \lambda_2, \cdots, \lambda_n)$ be the mean arrival rates of the relayed packets.
packets, and $\gamma = (\gamma_1, \gamma_2, \cdots, \gamma_n)$ be the mean arrival rates of the exogenous packets. Unlike the state transition used for Markov chains, the rows of $R$ matrix need not necessarily sum up to one, i.e., $\sum_j r_{ij} \leq 1$. The routing matrix $R$ is simply generated by the underlying DTN routing algorithm $R$ during a stationary period.

Assuming the network reaches equilibrium, then we can write the following traffic equation using the flow conservation principle, in which the total sum of arrival rates entering the system is equal to the total departure rate under steady-state condition.

$$\lambda_j = \gamma_j + \sum_i^n \lambda_i r_{ij}, \quad j = 1, 2, \cdots, n. \quad (2.2)$$

In the steady state, assuming the network is stable, the aggregate input rate $\lambda_j$ into node $j$ is equal to the aggregate output rate from node $i$, $i = 1, i, \cdots, n$. Therefore, we have a system of $n$ equations and $n$ unknowns. These equations can be written in matrix form as,

$$\vec{\lambda} = \vec{\gamma} + \vec{\lambda} R, \quad (2.3)$$

and the aggregate arrival rate vector can be solved by,

$$\vec{\lambda} = \vec{\gamma}(I - R)^{-1} < \vec{\mu}, \quad (2.4)$$

where, $\gamma = (\gamma_1, \gamma_2, \cdots, \gamma_n)$ and the components of the vector $I$ give the arrival rates into the various stations, and $\mu = (\mu_1, \mu_2, \cdots, \mu_n)$ is a vector representing service rates. The service times are assumed to be mutually independent and also independent of the arrival process at that queue, regardless of the previous service times of the same packet in other nodes. Jackson’s Theorem [57] also states that the joint
steady state distribution for the number of packets in each node is given by:

\[
Pr[L_1 = n_1, L_2 = n_2, \cdots, L_n = n_n] = Pr[L_1 = n_1] \times Pr[L_2 = n_2] \times \cdots \times Pr[L_n = n_n] = \prod_{i=1}^{n} (1 - \rho_i) \rho_i^{n_i}
\]

where, \( L_i \) is the length of the queue at node \( i \) and \( n_i \) is a random integer. Then \( Pr[N_i = n_i] \) for all \( n_i = 1, \cdots, n_m \) can be calculated using equations for independent \( M/M/s \) queues. The mean queue size and mean delay for the \( j \)th queue are given by,

\[
E[L_j] = \frac{\lambda_j}{\mu_j - \lambda_j} = \frac{\rho_j}{1 - \rho_j},
\]

\[
E[D_j] = \frac{1}{\mu_j - \lambda_j},
\]

\( j = 1, 2 \cdots, n. \)

where, we can compute the arrival rate \( \lambda_j \) and the expected queuing delay \( E[D_j] \) from Equations (2.4) and (2.6), respectively.

Consider a path \( x = x_\ell \rightarrow x_{\ell-1} \rightarrow \cdots \rightarrow x_1 \) from source node \( x_\ell \) to a destination node \( x_1 \), then the end-to-end delay for traffic originated at node \( x_\ell \) is:

\[
E[D_x] = \sum_{j=1}^{\ell} \frac{1}{\mu_{x_j} - \lambda_{x_j}}
\]

Let \( \mathcal{X} \) be the set of paths generated by routing algorithm \( \mathcal{R} \), then the average E2E delay \( \overline{D} \) can be computed by,

\[
\overline{D} = \frac{1}{|\mathcal{X}|} \sum_{x \in \mathcal{X}} E[D_x]
\]

Consider the network in Figure 2.4 in which some links may not always be con-
Figure 2.4: An inter-planetary network (a) and its corresponding dynamic graph (b).

Without lost of generality, assume that during a particular time interval $\Delta t$ all traffic generated by nodes 2-7 are heading towards node 1, then Equation (2.4) gives the following solutions.

$$
\overrightarrow{\lambda} = \begin{bmatrix}
\gamma_1 + \gamma_2 + \gamma_3 + \gamma_4 + \gamma_5 + \gamma_6 + \gamma_7 \\
\gamma_2 + 1/2 \gamma_4 + 1/2 \gamma_5 + 1/2 \gamma_6 + 1/2 \gamma_7 \\
\gamma_3 + 1/2 \gamma_4 + 1/2 \gamma_5 + 1/2 \gamma_6 + 1/2 \gamma_7 \\
\gamma_4 \\
\gamma_5 + 1/2 \gamma_7 \\
\gamma_6 + 1/2 \gamma_7 \\
\gamma_7
\end{bmatrix}
$$

(2.9)

If we assume the exogenous generated locally by each node has a rate $\gamma$, then the E2E delay for each traffic source can be computed by Equation(2.7) as,

$$
\overrightarrow{E}(D_i) = 1/(\mu_i - \gamma_i), \quad 1 \leq i \leq n.
$$

(2.10)

Note that $\gamma_i$ is bounded by the condition in Equation (2.4) and must hold for each node (i.e., $\lambda_i < \mu_i$).
2.5 Long term behavior of link in Delay and Disruption Tolerant Network (DTN)

In many DTN applications, mobility results in unusual and repetitive occurrences of network partitioning and topology changes. Thus, it is possible that two nodes that are connected at a given time $t$ may not be connected at time $t + \Delta t$ and it is also possible that at this time new connections are formed. We assume that link intermittency is transient. We also assume that over a long period of time mobility provides an asymptotic routing matrix $R$, for packet forwarding. That is, if we perform time series analysis of opportunistic connections due to mobility, over a very long period of time, it generates end-to-end paths between different pair of nodes in the network. The time series analysis serves two purposes. First, it obtains an understanding of the underlying forces and structure that have produced the observed data. Here it happens to be the mobility. Second, it fits a DTN model and proceeds to forecasting, monitoring or even feedback and feed forward control. Time series analysis is used for many applications such as economic forecasting, sales forecasting, budgetary analysis, stock market analysis, yield projections, process and quality control, etc. Recently, it starts being used in the field of computer networks communications [61]. The authors in [62] and [41] use time-evolving network topology, due to mobility in PDTNs, to solve different problems of topology design and topology control, respectively. Also in [45], the authors measure accuracy of their proposed stochastic model by analyzing different paths generated over a long period of time, due to heterogeneous mobility, starting from every node separately. For illustration purposes, Figure 2.5 demonstrates time-evolving snap shots of a tree topology. The time-series analysis of DTN topologies help us to make predictions about the topology information which can be used to build efficient approximation models for DTN.
2.6 Performance Evaluation

One of challenges in modeling and analysis of DTNs is how to determine the distribution of contact times and intercontact times, as well as the distribution of aggregate packet arrivals at DTN node. To better understand these essential ingredients, we looked into various real-life data sets. In this section we discuss about the real life traces and simulation tools we have used to measure the performance of DTNs.

2.6.1 Traces

The traces we have obtained are based upon the node mobility in real life scenarios, such as, satellite mobility in space, terrestrial mobility (sensor networks, human mobility, vehicular mobility) and few synthetic mobility data. Traces for satellite...
mobility is obtained from NASA research papers [4] and [6] and early topologies used by NASA to study DTN. Terrestrial mobility traces are obtained from CRAWDAD (A Community Resource for Archiving Wireless Data At Dartmouth). These traces consist of vehicular mobility, human mobility and sensor devices mobility. We have also used synthetic stationary mobility data to test base cases and relationships between different performance parameters. Synthetic data helped us to verify certain topologies (Tree and Meshes) which may evolve due to a mobility model over a long period of time and helped to find bottlenecks in the proposed models.

**Satellite Mobility Trace**

As mentioned earlier we gathered satellite mobility traces based upon research papers and other white papers NASA published regarding different satellite topologies. The mobility in such scenarios was predictable and topology were predictable and showed stationary behavior. Here stationary behavior means there were fixed set of links and only particular set of links were switching between ON and OFF states. The topologies formed by the nodes were mostly meshes [4], [6] and [5]. We have discussed results regarding these topologies in detail in respective chapters 3, 4 and 5. These topologies helped us to evaluate our model and also helped to check some base cases while developing our models. Some examples of base cases are, comparing E2E delay performance with no link failure and varying load, varying service rate based upon failure distribution and comparing results with no link failure cases and also observing traffic distributions at each queue. Our models obtained very close approximations for E2E delay analysis and calculating variance for these traces.

**Vehicular and Human Mobility Trace**

To build and evaluate our model for terrestrial DTN (e.g. vehicular networks, sensor networks, animal tracking, human mobility, etc.), one of the most important aspect
that we need to understand is nature of inter-contact and contact duration. Most of the proposed models in literature make some fundamental assumptions regarding contact and inter-contact times. Mostly the distribution of these two time periods are considered exponential. And recently it has been observed that some human mobility pattern deviates from this fundamental assumption [63], rather human mobility exhibits power-law + exponential tail distribution behaviors for inter-contact times. With these real life traces we tried to find asymmetric nature of contact and inter-contact periods. Based upon which we measure how closely our proposed models approximates E2E delay. Vehicular mobility and sensor mobility deviates from this behavior and are closer to exponential distribution for inter contact and contact periods.

To analyze these traces we have obtained Cumulative Distribution Function (CDF) and distributions of contact/inter-contact periods. Then, we used the inter-contact and contact time distributions from these traces to approximate E2E delay and variance using our models.

![Figure 2.6: CDF of inter-contact periods for vehicular networks [1]](image)

The oviedo/asturies-er dataset (v.2016-04-12) [1], this data set contains connectivity traces extracted from GPS traces collected from the regional Fire Department
of Asturias, Spain. The original data source is one year of GPS traces extracted from a Geographical Information System (GIS). The traces were generated by GPS devices embedded mainly in cars and trucks, but also in a helicopter and a few personal radios. A total of 229 devices reported 19,462,339 locations. A new location is reported with an interval of approximately 30 seconds when the GPS device detects movement. To convert GPS traces into ONE connectivity traces, they have assumed circular communication ranges of 10, 50 and 200 meters. For our work we
have considered trace with 50 meter range because that range doesn’t make topology too sparsely or too densely connected. There is a connection between nodes that are closer than the given range. For simplicity, we assume that the position of a device is always the last position reported. Their analysis show several important findings for the design of network protocols from the physical to the application layer. The networks examined are heterogeneous in the contact duration and the number of nodes contacted (degree centrality). In addition, they are sparse and partitioned, but delay- tolerant routes connecting these partitions exist. Finally, there are patterns in the connection between nodes that can ease the discovery of these routes and the deployment of delay-tolerant services [1].

In Figures 2.6 and 2.7, show CDF of inter-contact and contact time respectively for [1] trace.

Figures 2.8 shows exponential distribution for contact and inter-contact periods respectively. The inter-contact time shows certain spikes for few links which have very large inter-contact time.

Cambridge/haggle data set [2], are traces of Bluetooth sightings by groups of users carrying small devices (iMotes) for a number of days in office environments and conference environments.
In Figures 2.10 and 2.11, show CDF of inter-contact and contact time respectively for [2] trace.

![CDF OFF periods](image)

Figure 2.10: CDF of inter-contact periods for iMote devices [2]

![CDF ON periods](image)

Figure 2.11: CDF of contact periods for iMote devices [2]

Figures 2.12 and 2.13 show contact and inter-contact periods distribution respectively. Clearly, these distributions show a group mobility model, which is also observed in [63]. Cluster of histograms in Figures 2.12 and 2.13 show that different set of nodes (human beings carrying iMote devices) visit similar locations hence shows clus-
tering behavior. This behavior is observed as power law + exponential inter-contact distribution in [63].

Figures 2.16 and 2.17 show contact and inter-contact periods distribution respectively. Although MIT reality trace is based upon human mobility model but contact and inter-contact distributions show exponential behavior.

Mit/reality data set [3], traces of communication, proximity, location, and activity information from 100 subjects at MIT over the course of the 2004-2005 academic year. This data represents over 350,000 hours (40 years) of continuous data on human
behavior. Such rich data on complex social systems have implications for a variety of fields.

Figure 2.14 and 2.15, shows CDF of inter-contact and contact time respectively for [3] trace.

The exponential distribution for inter-contact and contact time in DTN have also
been observed in [64], [45], [35] and [54] and have been considered as a realistic assumption.

2.6.2 Simulation Tools

The Opportunistic Networking Environment (ONE) simulator [65], is a discrete event simulation engine and its developed in Java language. The simulator is open source. The simulator was specifically designed for evaluating DTN routing and application protocols. It allows users to create scenarios based upon different synthetic movement
models and real-world traces and offers a framework for implementing routing and application protocols. This simulator includes six well-known routing protocols, First Contact [48], Epidemic [15], Spray And Wait [13], MaxProp [66], Prophet [67], Direct Delivery [48].

The main functions of the ONE simulator are the modeling of node movement, inter-node contacts, routing and message handling. Result collection and analysis are done through visualization, reports and post-processing tools.

Node movement is implemented by different mobility models. These are either synthetic models or existing movement traces. Connectivity between the nodes is based on their location, communication range and the bit-rate. The routing function is implemented by routing modules that decide which messages to forward over existing contacts. Finally, the messages themselves are generated through event generators. The messages are always unicast, having a single source and destination host inside the simulation world. Simulation results are collected primarily through reports generated by report modules during the simulation run. Report modules receive events (e.g., message or connectivity events) from the simulation engine and generate results based on them. The results generated may be logs of events that are then further processed by the external post-processing tools, or they may be aggregate statistics calculated in the simulator. Secondarily, the graphical user interface (GUI) displays a visualization of the simulation state showing the locations, active contacts and messages carried by the nodes.

A node models a mobile endpoint capable of acting as a store-carry-forward router (e.g., a pedestrian, car or tram with the required hardware). Simulation scenarios are built from groups of nodes in a simulation world. Each group is configured with different capabilities. Each node has a set of basic capabilities that are modeled. These are radio interface, persistent storage, movement, energy consumption and message routing. The focus of the simulator is on modeling the behavior of store-carry-forward networking, and hence the authors have deliberately refrained from detailed
modeling of the lower layer mechanisms such as signal attenuation and congestion of the physical medium. Instead, the radio link is abstracted to a communication range and bit-rate. These are statically configured and typically assumed to remain constant over the simulation.

Node movement capabilities are implemented through mobility models. Mobility models define the algorithms and rules that generate the node movement paths. Three types of synthetic movement models are included: 1) random movement, 2) map-constrained random movement, and 3) human behavior based movement.

Interactive visualization and post-processing tools support evaluating experiments and an emulation mode allows the ONE simulator to become part of a real-world DTN testbed.

As mentioned earlier that the ONE simulator was designed for evaluating DTN routing and application protocols considering node mobility. However, the simulator doesn’t capture the relationship between packet arrival, interrupted service rate, queuing delay and nature of inter-contact and contact time distributions for calculating E2E delay or variance (jitter). We tested this by creating few DTN scenario as per ONE simulator guidelines, first we fixed tree topology (i.e. stationary movement model) and made buffer (queue) size infinite, so there is no packet drop or message loss. We also made exponential packet arrival and service rates. We then observed average E2E delay by varying packet inter-arrival time and made packet size distribution geometric and later exponential. For the purpose of studying the effects of different parameters for our models we tested the simulator with no link failure. We observed that increasing load on the network doesn’t effect average E2E delay. Specifically, increasing load ((ρ > 40%)) showed negligible effect on E2E delay. We observed similar behavior for other topologies, such as mesh or star topology.

These limitations of ONE simulator encouraged us to develop a new simulator and measure correctness of our proposed models. We wrote our own event based continuous time simulator (dtnSim) in C++. dtnSim incorporates the effect of queu-
ing delay, interrupted service rate for a given inter-contact and contact distribution, exponential packet arrival and varying load on the system to evaluate average E2E delay and variance in DTNs. We maintained all the assumptions those we made in theory. We have tested the simulator for various traces (real life and synthetic) as mentioned above. We have verified our simulator by performing white box and black box testing for different parameters those we have considered in our model. The simulator evaluates our model efficiently and results have shown that our model approximates DTN performance efficiently.
Chapter 3

Mean Time To Failure and Mean Time To Recovery for PDTNs

Mobility changes topology dynamically but also provides new connection opportunities due to which eventually there is a path between a source and destination pair over a long time [62]. For certain types of DTNs, the temporal characteristics of topology could be known a priori or can be predicted from historical data. Such DTNs are known as PDTNs [62]. Examples of such DTN topologies are satellite communication [68], public transportation (buses, trains, etc.) [69] and spatio-temporal based human mobility model [70]. The link intermittency due to mobility could be deterministic or probabilistic. In this chapter we consider DTN scenarios in which intermittent nature of links are known and based on which topology information can be predicted. This class of DTN are also known as time-evolving and predictable DTNs. In this chapter, we have developed a stochastic model that is based on an open queuing system and ALA. ALA is derived from Mean Time To Failure (MTTF) and Mean Time To Recovery (MTTR) that predicts the end-to-end delay in closed form for DTN. Using this model we try to achieve two goals. First, we discuss the concept of MTTF and MTTR for a link in DTN and subsequently we will discuss
about deploying a traffic distribution model for DTN that incorporates dissemination of packets. Second, we employ an open queuing system based on Jackson’s tandem queuing network [57] and MTTF/MTTR [59] that facilitate the computation of an end-to-end delay performance under various traffic load and link failure conditions under PDTN.

3.1 The Model

3.1.1 Link Availability

We first illustrate the impact of mobility on long-term connectivity behavior and probabilistic paths generated due to it. We then propose a generic model for link availability. For the sake of the argument, we consider the mobility model addressed in [70], where the authors studies the spatio-temporal correlation of mobility. Figure 3.1 illustrates four consecutive snapshots of a mobility model and its impact on the underlying network connectivity. Initially 9 nodes were randomly placed in 4 zones (communities). Mobility within a zone is governed by random waypoint and between zones is driven by a power law. The model has been adopted in [70] has been observed on real traces [71]. In this simulation, we considered temporal and spacial movement similar to those used in [70]. For the purpose of illustration and visualization, Figure 3.1 represents a small scale network. Given the non-uniformity of movements, some links provide higher connectivity of service than the others. Figure 3.2 illustrates the long-term connectivity graph and shows connection availability between each pair of nodes. The color density of a vertex represents the relative participation of the node in packet forwarding.

Generally, a spanning tree from/to is formed for packet forwarding/receiving from based on link availability. The tree may not be connected all the time or such a tree may not exist for a relatively long time. We will discuss how such a delay can be
Figure 3.1: Four consecutive snapshots of the network connectivity due to mobility.

Figure 3.2: Network connection availability graph.

Figure 3.3: Maximum spanning tree corresponding to link availability in Figure 3.2.
approximated in Section 2.4, but first we model a node as a queue that receives traffic aggregates and serves them on its outgoing links. We show that a link behavior in terms of its connectivity between two nodes is a renewal process.

Let $X_i$ be the time between the $(i-1)$st and the $i$th events. A renewal process is defined to be a discrete-time independent process $\{X_n|n = 1, 2, \ldots\}$ where $X_1, X_2, \ldots$, are independent, identically distributed (iid), non-negative random variables [59]. As an example of such a process, consider a link in a DTN network, which alternates between connected (ON) and disconnected (OFF) periods. Assuming, ON (connected or operative) period for each link is identically distributed, independent and can be presented by a non-negative random number. Then, we can call the link functioning period as an independent process. Similarly, the OFF (disconnected or inoperative) period of the link is an independent process.

### 3.1.2 Link Availability Analysis

In a DTN scenario, assume that a link fails (disruption or disconnection) at time $t$ and is repaired some time later at $t + \Delta t$, and becomes ready for transmission. Let $C_i$ be the duration of the $i$th functioning (connectivity) and $D_i$ be the duration of link downtime (disruption or disconnection) for the $i$th period. In this formulation we assume that $C_i$ and $D_i$ are iid’s, non-negative continuous random variables, say $C_1, D_1, C_2, D_2, \ldots$. We also assume that $C_i$ and $D_i$ are independent of the queue size at node $i$. This process is illustrated in Figure 3.4. Since $C_i$ and $D_i$ are iid, we

![Figure 3.4: A representation of a sequence of ON-OFF periods and renewal process $X_i$.](image)

assume that the sequence of random variables $\{X_i = C_i + D_i| i = 1, 2, \ldots\}$ is mutually
independent. Thus, \( X_i \)'s are also identically distributed. Hence \( \{X_i|i = 1, 2, \ldots \} \) is a renewal process. A renewal point of this process corresponds to the event of the completion of a repair (connection restored). Further we assume the probability density functions (pdf) for connectivity \( C_i \) and disruption \( D_i \) periods are represented as \( w(t) \), and \( g(t) \) respectively. The underlying density \( f(t) \) of the renewal process \( X_i \) is the convolution of \( w \) and \( g \) (assuming \( C_i \) and \( D_i \) are independent). Using Laplace Transform we transform the probability density functions from time domain to frequency domain. Effectively, we express the probability density function \( f(t) \) in terms of the frequency of availability of a link, where the frequency of being ON (functioning) and OFF (repair) can be obtained from Laplace Transform of \( w(t) \) and \( g(t) \) respectively.

Now the instantaneous availability \( A(t) \) of a link is defined as the probability that the link is functioning (ON) at time \( t \). Let \( L_f(s) \) represents the Laplace transformation of \( f(t) \). \( L_w(s) \) and \( L_g(s) \), represents the Laplace transformations of \( w(t) \) and \( g(t) \), respectively. Thus, from the convolution of \( w \) and \( g \) we have:

\[
L_f(s) = L_w(s)L_g(s) \tag{3.1}
\]

and the renewal density is given by:

\[
L_m(s) = \frac{L_w(s)L_g(s)}{1 - L_w(s)L_g(s)} \tag{3.2}
\]

From Equation (3.1) and Equation (3.2) the instantaneous availability can be calculated as:

\[
L_A(s) = \frac{1 - L_w(s)}{s[1 - L_w(s)L_g(s)]} \tag{3.3}
\]

From Equation (3.3) we can conclude that if we are given the failure-time and repair-time distributions, the above equation enables us to compute the instantaneous availability \( A(t) \) as a function of time.
Since we are analyzing the state of a link in DTN after a sufficiently long period of time, we define the limiting availability (or simply availability) $A$ as the limiting value of $A(t)$ as $t$ approaches infinity.

$$A = \lim_{t \to \infty} A(t) = \lim_{t \to 0} sL_A(s) \quad (3.4)$$

From Equation (3.3) and Equation (3.4), we get availability $A$ as:

$$A = \frac{MTTF}{MTTF + MTTR} \quad (3.5)$$

Assuming that link $i$'s ON/OFF pdf parameters are given as $\alpha$ and $\beta$, respectively, then $MTTR = 1/\alpha$ and $MTTF = 1/\beta$.

### 3.2 Mttf and Mttr for Jackson’s Network

We consider the Jackson queuing network discussed in Section 2.4. The server (link) at node $j$ goes through alternating ON and OFF periods, distributed exponentially with parameters $\alpha_j$ and $\beta_j$ respectively. These periods are independent of the states of other nodes and of the number of packets in the queue. Transitions from the ON to the OFF state are called failure (or disruption) while those from OFF to ON state are repairs (or connection). If a failure occurs during a service, the latter is resumed from the point of interruption after the repair. Incoming packets continue to join
the queue during OFF periods. Due to exponential failure the average number of packets that server $j$ can complete per unit time can be obtained using Equation (3.5) as follows:

$$\mu'_j = \mu_j \frac{\alpha_j}{\beta_j + \alpha_j}, \quad j = 1, 2, \ldots, N$$  \hspace{1cm} (3.6)

where $N$ is total number of servers. The arrival rates and routing probabilities remain the same. The resultant network has a product-form solution. Delay at node $j$ can be calculated using Equation (3.6) as:

$$E'[D_j] = \frac{1}{\mu'_j - \lambda_j}, \quad j = 1, 2 \ldots, n.$$  \hspace{1cm} (3.7)

Thus, Equation (2.7) gets modified as:

$$E'[D_x] = \ell \sum_{j=1}^{\ell} \frac{1}{\mu'_{x_j} - \lambda_{x_j}}.$$  \hspace{1cm} (3.8)

and average end-to-end delay $\overline{D'}$ under failure can be computed by,

$$\overline{D'} = \frac{1}{|X|} \sum_{x \in X} E'[D_x].$$  \hspace{1cm} (3.9)

Let the average availability of each link in Figure 3.6 be described by Equation (3.5) and average service rate of a server is given by Equation (3.6). The aggregate
traffic vector can be obtained from Equation (2.3) as,

\[ \vec{\lambda} = \begin{bmatrix} \gamma_1 + \gamma_2 + \gamma_3 + \gamma_4 + \gamma_5 + \gamma_6 + \gamma_7 \\ \gamma_2 + 1/2 \gamma_4 + 1/2 \gamma_5 + 1/2 \gamma_6 + 1/2 \gamma_7 \\ \gamma_3 + 1/2 \gamma_4 + 1/2 \gamma_5 + 1/2 \gamma_6 + 1/2 \gamma_7 \\ \gamma_4 \\ \gamma_5 + 1/2 \gamma_7 \\ \gamma_6 + 1/2 \gamma_7 \\ \gamma_7 \end{bmatrix} \]  

(3.10)

Based on the aggregate traffic at each node, Equation (3.11) represents a vector of queuing delay at each node.

\[ \vec{d} = \begin{bmatrix} (\mu_1 - \gamma_1)^{-1} \\ (\mu_2 - \gamma_1 - \gamma_2)^{-1} \\ (\mu_3 - 1/2 \gamma_1 - 1/2 \gamma_2 - \gamma_3)^{-1} \\ (\mu_4 - 1/2 \gamma_1 - 1/2 \gamma_2 - \gamma_4)^{-1} \\ (\mu_5 - \gamma_1 - \gamma_2 - \gamma_3 - \gamma_4 - \gamma_5)^{-1} \\ (\mu_6 - \gamma_1 - \gamma_2 - \gamma_3 - \gamma_4 - \gamma_5 - \gamma_6)^{-1} \end{bmatrix} \]  

(3.11)

Now, the end-to-end delay can be computed based on a path generated by a routing algorithm.
\[ \vec{p} = \begin{bmatrix} q(1) + q(2) + 1/2q(3) + q(5) + 1/2q(4) \\ q(2) + 1/2q(3) + q(5) + 1/2q(4) \\ q(3) + q(5) \\ q(4) + q(5) \\ q(5) \\ 0 \end{bmatrix} \]

(3.12)

Here \( \vec{p} \) is path vector. Given that node 6 is a sink node, it does not impose any queuing delay.

### 3.3 Simulation and Numerical Results

We have tested the model for various network topologies (Figures 3.7, 3.8 and 3.6) using both the model described in this chapter as well simulations. In these experiments, we have maintained an unsaturated state for each node (queue) in the system and the stability condition at each node under composite traffic aggregation by conforming with Equation (2.4). Without loss of generality, we assumed that all node generate traffic with various intensities and in each topology, all traffic lead to the node with highest label. For example node 6 is highest label in Figure 3.6. This allows us to measure the worse case delay scenarios.

We have assumed that time series analysis of long-term behavior of mobility model generates tree or mesh topologies as mentioned in section 3.1. These topologies are weighted by routing matrix \( \mathbf{R} \). To make sure that stability condition has been preserved, we calculated the maximum traffic aggregates in each network. Each node generates traffic at rate \( \gamma \). We varied \( \lambda \) to test the network under various load conditions. Let \( \rho = M\lambda/\mu \), where, \( M \) is the maximum aggregate traffic, then
for $0.1 \leq \rho \leq 0.9 \lambda$ has been calculated. In terms of simulation parameters, for the purpose of consistency, we chose packet lengths to be geometrically distributed with mean 250 bytes. The channel speed varied from 250KB to 2.5MB in various
experiments, not all shown here. We have run experiments long enough, to meet steady state requirement of the model.

The results obtained from the model in Sections 3.1 and 3.2, have been compared with the simulation results obtained under various traffic load conditions and failure distributions with different parameters. For each topology we have compared end-to-end delays with no link failure and link failure with various ON and OFF distributions. We observe that end-to-end delay approximation is extremely close to simulation results. The reason the model gives a slightly better performance than the simulation experiments is the result of finer timing used in model ($\Delta t \rightarrow 0$) and exponentially generated pseudo random numbers generated by the simulation. However, simulation results on various network topologies and under different traffic conditions confirm the accuracy of the model within the conventional bounds of statistical significance.

Figure 3.9: end-to-end delay analysis of NASA 6 node topology (Figure 3.6)
In this chapter, we have developed a new model that can be used to approximate the average end-to-end delay in PDTNs. The results shows that our model very closely
approximates end-to-end delay for various link failure distributions. The model is based on open tandem queuing system and link’s availability. We have augmented the model to incorporate various types of link failure distributions. The model is quite general in the sense that it can accommodate various probabilistic routing algorithms as long as the algorithms generates a routing matrix which is irreducible on each time interval. The limitation of the model relates to the scenarios when link failure is not transient and topology is very dynamic.
Chapter 4

Disruptive Link Modeling and Delay Analysis in Delay Tolerant Networks

In this chapter, we develop a stochastic model where continuous point-to-point connectivity does not always exist between a pair of nodes and calculated E2E delay based on queue length distribution under link intermittency. The model uses an open tandem queuing system with potential server (link) break down along with a probabilistic routing matrix to estimate the average end-to-end delay. The model in [72] which is discussed in detail in chapter 5 is based on a discrete-time discrete-state Markov chain, that provides the expected connection time between two end points. While it is applicable for a small or sparse network, it runs into state-space explosion for large networks.

The computational complexity of model discussed in this chapter is reduced significantly by an order of magnitude. This is mainly due to the fact that the approach reduces the network state-space explosion to a number of possible manageable states for in-coming and out-going probabilistic links of a given node to compute the aver-
age queuing delay at each node. The asymptomatic routing matrix and the resultant routing paths due to mobility take care of the rest. While several routing algorithms such as delegation forwarding [16], and optimal probabilistic forwarding [14] offer routing strategies and produce a probabilistic routing matrix, the proposed model in this chapter is more general in the sense that it can accommodate any probabilistic routing protocol as long as the network relies on the assumption that node mobility exhibits a long-term steady state or regularity on link disruption probability, and the underlying algorithm produces a long-term probabilistic routing matrix. The method provides an average end-to-end delay and its delay variance (jitter) in closed forms.

4.1 Modeling a DTN Link

4.1.1 Non-Disruptive Links

The queue associated with each node is modeled as an $M/M/1$ queuing system in which packet arrive according to a Poisson process and the transmission times of the packets are independent and identically exponentially distributed. Such a system can be described by a continuous time/discrete state Markov process with states $k$, $k = 0, 1, \cdots$, where $k$ is simply the number of packets in the system. The generator $Q$ of this Markov process is given by,

$$Q = \begin{bmatrix}
-\lambda & \lambda & 0 & 0 & 0 & \cdots \\
\mu & -(\lambda + \mu) & \lambda & 0 & 0 & \cdots \\
0 & \mu & -(\lambda + \mu) & \lambda & 0 & \cdots \\
0 & 0 & \mu & -(\lambda + \mu) & \lambda & \cdots \\
\vdots & \vdots & \vdots & \vdots & \vdots & \ddots
\end{bmatrix} \quad (4.1)$$

where $\lambda$ is the arrival rate and $\mu$ is the service rate ($\lambda < \mu$). The corresponding transition-rate diagram of the $M/M/1$ model is shown in Figure 4.1.
Let $\pi_k$ be the equilibrium probability of state $k$, $k > 0$, then we can write balance equations from the state probabilities recursively as,

$$
\begin{align*}
\lambda \pi_0 &= \mu \pi_1, & \Rightarrow \pi_1 &= \frac{\lambda}{\mu} \pi_0 \\
(\lambda + \mu) \pi_1 &= \lambda \pi_0 + \mu \pi_2, & \Rightarrow \pi_2 &= \left(\frac{\lambda}{\mu}\right)^2 \pi_0 \\
(\lambda + \mu) \pi_2 &= \lambda \pi_1 + \mu \pi_3, & \Rightarrow \pi_3 &= \left(\frac{\lambda}{\mu}\right)^3 \pi_0 \\
& \vdots & \vdots \\
(\lambda + \mu) \pi_k &= \lambda \pi_{k-1} + \mu \pi_{k+1}, & \Rightarrow \pi_{k+1} &= \left(\frac{\lambda}{\mu}\right)^{k+1} \pi_0, \\
& \vdots & \vdots \\
& k = 1, 2, \ldots
\end{align*}
$$

The set of equations in (4.2) has a (unique) geometric solution,

$$
\pi_k = \pi_0 \rho^k = (1 - \rho) \rho^k, \quad \rho = \lambda/\mu, \quad k = 0, 1, \ldots
$$

The average queue size can be computed as,

$$
Q = \sum_{k=0}^{\infty} k \pi_k = (1 - \rho) \sum_{k=0}^{\infty} k \rho^k = \frac{\rho}{1 - \rho}
$$

The balance equations balance the flow out of a state and the flow into that state. The flow is the mean number of transitions per time unit. The limiting probabilities, or equilibrium probabilities, can be computed from the balance equations. The events that cause the system to make a transition from state $i$ to state $j$ occur with a frequency or rate $q_{ij}$. Therefore, the mean number of transitions per time unit from
state $i$ to state $j$ is $\pi_k q_{ij}$. With $S$ as a set of states, the balance equations can be written as,

$$\pi_k \sum_{j \neq i} q_{ij} = \sum_{j \neq i} \pi_j q_{ji}, \quad \text{or} \quad \sum_{i \in S} \pi_j q_{ji} = 0, \quad i \in S \quad (4.5)$$

In matrix notation,

$$pQ = 0, \quad \sum_{i \in S} \pi_k = 1 \quad (4.6)$$

The average queuing delay (system waiting time) can be obtained by Little’s formula,

$$E(Q) = \frac{\rho}{\lambda(1 - \rho)} = \frac{1}{\mu - \lambda} \quad (4.7)$$

### 4.1.2 Disruptive Links

In this section we consider an unreliable link that serves a queue. The link breaks down for various reasons including long transmission range, fading signals, disconnectivity, etc. We consider intermittent link that serves a node buffer which is modeled as an $M/M/1$ queue. The link alternates between connected and disconnected periods. The link is subject to so-called time-dependent breakdowns as opposed to operational dependent breakdowns which only occur when the link sever is processing a packet. A soon as the link comes back, processing resumes at the point where it was interrupted. We assume that the durations of a link being connected/disconnected are independent random variables, say $C_1, D_1, C_2, D_2, \ldots$. These random variables are also to be independent of queue size. The process is shown in Figure 4.2. The following analysis is based on time-dependent breakdowns.

![Figure 4.2: A representation of a sequence of OFF-ON periods.](image)

Since $C_i$ and $D_i$ are iid's, we assume that the sequence of random variables $\{X_i = \ldots\)$
$C_i + D_i \mid i = 1, 2, \cdots$ is mutually independent. This makes $\{X_i \mid i = 1, 2, \cdots\}$ a renewal process. The process is also known as the alternating renewal process as it models two states of a system [59]. The renewal process is illustrated in Figure 4.2. We assume that $C_1, C_2, \cdots$ and $D_1, D_2, \cdots$ are exponentially distributed with parameters $\alpha$ and $\beta$, respectively.

Packets arrive according to a Poisson stream with rate $\lambda$. The transmission times (service times) are exponentially distributed with mean $1/\mu$. While during connected periods, the system operates as an $M/M/1$ queue with mean arrival rate $\lambda$ and mean departure rate $\mu$. During disconnected periods, no packet departs, but packets can still arrive at Poisson rate $\lambda$. The link switches between ON (1) and OFF (0) as shown Figure 4.3. We model this system as a continuous time discrete state Markov chain.

\[
\begin{align*}
\text{OFF} & \quad \alpha \\
0 & \quad \beta \\
\text{ON} & \quad 1
\end{align*}
\]

Figure 4.3: States of link connectivity.

The transition matrix $P$ of the Markov chain and its generator $Q$ can be described, respectively as,

\[
P = \begin{bmatrix} 1 - 1/\alpha & 1/\alpha \\ 1/\beta & 1 - 1/\beta \end{bmatrix}, \quad Q = \begin{bmatrix} -\alpha & \alpha \\ \beta & -\beta \end{bmatrix}
\]  \hspace{1cm} (4.8)

The steady-state vector of the Markov chain is $\pi = [\pi_0, \pi_1]$ such that

\[
\pi Q = 0, \quad \pi_0 + \pi_1 = 1
\]  \hspace{1cm} (4.9)

where $\pi_0$ represents the long-term mean probability of being in state OFF(0) and
\( \pi_1 \) represents the long-term mean probability of being in state \( \text{ON}(1) \). After solving (4.9), we obtain:

\[
\pi = [\pi_0, \pi_1] = \left( \frac{\beta}{\alpha + \beta}, \frac{\alpha}{\alpha + \beta} \right) \tag{4.10}
\]

\( \pi_k \) is a fraction of time the link is in state \( i \in \{0, 1\} \). Now, consider transition rate diagram of an \( M/M/1 \) queue with output link intermittency shown in Figure 4.4.

The system can be described by two-dimensional Markov process with states \((i, k)\) where \( i \) indicates that state of the link. The link is up if \( i = 1 \) and it is down if \( i = 0 \), \( k = 0, 1, \ldots \). The necessary condition for stability requires that

\[
\frac{\lambda}{\mu} < \pi_1 = \frac{\alpha}{\alpha + \beta} \tag{4.11}
\]

where \( \pi_1 \) denote the fraction of time the link is up.

Let \( \pi(i, k) \) denotes the equilibrium probability of state \((i, k)\), \( i \in \{0, 1\}, k \geq 0 \). From the transition rate diagram in Figure 4.4 we can get the following balance equations. For the states \((0, 1)\) and \((1, 1)\),

\[
(\lambda + \alpha)\pi(0, 0) = \beta \pi(1, 0) \tag{4.12}
\]

\[
(\lambda + \beta)\pi(1, 0) = \mu \pi(1, 1) + \alpha \pi(0, 0) \tag{4.13}
\]

For all other states \((i, k)\), with \( k \geq 1 \),
\[(\lambda + \alpha)\pi(0, k) = \lambda\pi(0, k - 1) + \beta\pi(1, k) \quad (4.14)\]

\[(\lambda + \mu + \beta)\pi(1, k) = \lambda\pi(1, k - 1) + \alpha\pi(0, k) + \mu\pi(1, k + 1) \quad (4.15)\]

The corresponding transition rate matrix:

\[
Q = \begin{bmatrix}
  d_1 & \alpha & \lambda & \cdots & \\
  \beta & d_2 & \lambda & \cdots & \\
  & d_1 & \alpha & \lambda & \cdots & \\
  & \mu & \beta & d_3 & \lambda & \cdots & \\
  & & & & d_1 & \alpha & \cdots & \\
  & & & & \mu & \beta & d_3 & \cdots & \\
  \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots
\end{bmatrix} \quad (4.16)
\]

where \(d_1 = - (\lambda + \alpha)\), \(d_2 = - (\lambda + \beta)\), and \(d_3 = - (\lambda + \mu + \beta)\). We can re-write equations (4.14)-(4.16) as:

\[
\lambda\pi(0, k - 1) + \beta\pi(1, k) - (\lambda + \alpha)\pi(0, k) = 0 \quad (4.17)
\]

\[
\lambda\pi(1, k - 1) - (\lambda + \mu + \beta)\pi(1, k) + \alpha\pi(0, k) + \mu\pi(1, k + 1) = 0 \quad (4.18)
\]

\[k = 1, 2, \ldots\]
Let $\pi$ be partitioned conformally with $Q$, i.e.,

$$\pi = (\pi_0, \pi_1, \ldots, \pi_k, \ldots)$$  \hspace{1cm} (4.19)

where,

$$\pi_k = (\pi(0, k), \pi(1, k))$$  \hspace{1cm} (4.20)

Equations (4.12) and (4.13) and subsequently Equations (4.17) - (4.18) give the following equations

$$\pi_0 D_0 + \pi_1 B_1 = 0$$  \hspace{1cm} (4.21)
$$\pi_0 A_1 + \pi_1 D_1 + \pi_2 B_1 = 0$$
$$\pi_1 A_1 + \pi_2 D_1 + \pi_3 B_1 = 0$$
$$\vdots$$
$$\pi_{k-1} A_1 + \pi_k D_1 + \pi_{k+1} B_1 = 0 \quad k \geq 1$$  \hspace{1cm} (4.22)
$$\vdots$$

where,

$$D_0 = \begin{bmatrix} -\lambda - \alpha & \alpha \\ \beta & -\lambda - \beta \end{bmatrix}$$  \hspace{1cm} (4.23)
$$D_1 = \begin{bmatrix} -\lambda - \alpha & \alpha \\ \beta & -\lambda - \mu - \beta \end{bmatrix}$$  \hspace{1cm} (4.24)
$$A_1 = \begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix}, \quad B_1 = \begin{bmatrix} 0 & 0 \\ 0 & \mu \end{bmatrix}$$  \hspace{1cm} (4.25)

This belongs to *structured Markov Chain* whose transition matrices have a special
block structure. Each state can be written as \{(i, k), i \in 0,1, k \geq 0\}. States are grouped into stages according to their \(k\) value. This forms tridiagonal blocks in which transitions are taken place between blocks. Diagonal blocks form states of the same level. Super-diagonal blocks form states in the next highest level, and sub-diagonal blocks form states in the adjacent lower levels. Such a Markov chain is called \textit{Quasi-Birth-Death processes}.

\[
Q = \begin{bmatrix}
D_0 & A_1 & 0 & 0 & 0 & 0 & \cdots \\
B_1 & D_1 & A_1 & 0 & 0 & 0 & \cdots \\
0 & B_1 & D_1 & A_1 & 0 & 0 & \cdots \\
0 & 0 & B_1 & D_1 & A_1 & 0 & \cdots \\
0 & 0 & 0 & B_1 & D_1 & D_1 & \cdots \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots
\end{bmatrix}
\]

(4.26)

where,

\[
\pi Q = 0
\]

(4.27)

Clearly, if we can determine the equilibrium probabilities \(\pi(i, k)\), then we can also compute the average queue length, and subsequently, by Little’s law, we can compute the average queuing delay. We use the \textit{matrix-geometric method} [73]. We first simplify the equilibrium equations (4.22) by eliminating the vector \(\pi_{k+1}\). This can be done by equating the flow between level \(k\) and level \(k + 1\).

\[
\pi(0, k) \lambda + \pi(1, k) \lambda = \pi(1, k + 1) \mu
\]

(4.28)

With a matrix notation,

\[
\pi_k A_2 = \pi_{k+1} B_1, \quad \text{where} \quad A_2 = \begin{bmatrix} 0 & \lambda \\ 0 & \lambda \end{bmatrix}
\]

(4.29)

Substituting Equation (4.29) in Equation (4.22) yields,
\[ \pi_{k-1}A_1 + \pi_kD_1 + \pi_kA_2 = \pi_{k-1}A_1 + \pi_k(D_1 + A_2) = 0, \ k \geq 1 \quad (4.30) \]

or

\[ \pi_k = -\pi_{k-1}A_1(D_1 + A_2)^{-1} = \pi_{k-1}R, \quad k \geq 1 \quad (4.31) \]

where,

\[ R = -A_1(D_1 + A_2)^{-1} = \frac{\lambda}{\mu} \begin{bmatrix} (\mu + \beta)/(\lambda + \alpha) & 1 \\ \beta/(\lambda + \alpha) & 1 \end{bmatrix} \quad (4.32) \]

Iterating Equation (4.31) leads to the matrix-geometric solution

\[ \pi_k = \pi_0 R^k \quad k \geq 1 \quad (4.33) \]

The sub-vectors \( \pi_k \) are geometrically related to each other. This is very similar to the solution for the \( M/M/1 \) model where, \( p_k = p_0 \rho^k \) given by Equation (4.3). Given \( \pi_0 \) and \( R \), we can find all other \( \pi_k \).

**Derivation of \( \pi_0 \) and \( \pi_1 \)**

The first two equations of \( \pi Q = 0 \) in (4.27) are:

\[ \pi_0 D_0 + \pi_1 B_1 = 0 \quad (4.34) \]
\[ \pi_0 A_1 + \pi_1 D_1 + \pi_2 B_1 = 0 \quad (4.35) \]
Replacing $\pi_2$ with $\pi_1 R$

\[
\pi_0 D_0 + \pi_1 B_1 = 0 \tag{4.36}
\]

\[
\pi_0 A_1 + \pi_1 (D_1 + RB_1) = 0 \tag{4.37}
\]

\[
(\pi_0, \pi_1) \begin{bmatrix} D_0 & B_1 \\ A_1 & D_1 + RB_1 \end{bmatrix} = [0, 0] \tag{4.38}
\]

which can be solved for $\pi_0$ and $\pi_1$ with the condition that $\pi e = 1$, where $e$ is a column of 1’s.

\[
1 = \pi e = \pi_0 e + \pi_1 e + \sum_{k=2}^{\infty} \pi_k e
\]

\[
= \pi_0 e + \pi_1 e + \sum_{k=2}^{\infty} \pi_1 R^{k-1} e = \pi_0 e + \sum_{k=1}^{\infty} \pi_1 R^{k-1} e
\]

\[
= \pi_0 e + \sum_{k=0}^{\infty} \pi_1 R^k e = \pi_0 e + \pi_1 \left( \sum_{k=0}^{\infty} R^k \right) e \tag{4.39}
\]

The eigenvalues of $R$ lie inside the unit circle which means that $(I - R)$ is nonsingular and hence

\[
\sum_{k=0}^{\infty} R^i = (I - R)^{-1} \tag{4.40}
\]

Normalizing vectors $\pi_0$ and $\pi_1$ by computing

\[
1 = \pi_0 e + \pi_1 (I - R)^{-1} e = \pi_0 e + \pi_0 R (I - R)^{-1} e
\]

\[
= \pi_0 e + \pi_0 \begin{bmatrix} \lambda (\mu - \lambda + \beta)/d & \lambda (\lambda + \alpha)/d \\ \lambda \beta/d & \alpha \lambda/d \end{bmatrix} e \tag{4.41}
\]

65
where, \( d = \alpha \mu - \alpha \lambda - \lambda \beta \).

Given \( \pi(1,0) = \frac{(\lambda + \alpha)}{\beta} \pi(0,0) \), we have

\[
1 = \pi_0 e + \pi_1 (I - R)^{-1} e = \pi_0 e + \pi_0 R (I - R)^{-1} e
\]

\[
= \pi_0 e + \pi_0 \begin{bmatrix}
\lambda (\mu - \lambda + \beta)/d & \lambda (\lambda + \alpha)/d \\
\lambda \beta/d & \alpha \lambda/d
\end{bmatrix} e
\]

\[
= \pi(0,0) + \pi(1,0) + [\pi(0,0) + \pi(1,0)]
\]

\[
\begin{bmatrix}
\lambda (\mu - \lambda + \beta)/d & \lambda (\lambda + \alpha)/d \\
\lambda \beta/d & \alpha \lambda/d
\end{bmatrix} e
\]

\[
= \lambda (\pi(0,0)\mu + \pi(0,0)\beta + \pi(1,0)\beta + \pi(0,0)\alpha + \pi(1,0)\alpha)/d
\]

\[
= \pi(0,0) \frac{\mu (\alpha \beta + \alpha \lambda + \alpha^2 + \lambda \beta)}{\beta (\alpha \mu - \alpha \lambda - \lambda \beta)}
\]

(4.42)

Hence

\[
\pi_0 = \begin{cases}
\pi(0,0) = \frac{\beta (\alpha \mu - \alpha \lambda - \lambda \beta)}{\mu (\alpha \beta + \alpha \lambda + \alpha^2 + \lambda \beta)} \\
\pi(0,1) = \frac{(\alpha \mu - \alpha \lambda - \lambda \beta)}{\mu (\alpha + \beta)}
\end{cases}
\]

(4.43)

Now we can compute the average queue length by

\[
E(L) = \sum_{k=1}^{\infty} k \pi_k = \sum_{k=1}^{\infty} k \pi_1 R^{k-1} e = \pi_1 (I - R)^{-2} e
\]

(4.44)

where, \( \pi_1 = \pi_0 R \) and,

\[
\pi_1 = \begin{cases}
\pi(1,0) = \frac{\lambda \beta (\alpha \mu - \alpha \lambda - \lambda \beta)(\mu + \beta + \lambda + \alpha)}{\mu^2 (\lambda + \alpha)(\alpha \beta + \alpha \lambda + \alpha^2 + \lambda \beta)} \\
\pi(1,1) = \frac{\lambda (\alpha \mu - \alpha \lambda - \lambda \beta)(\beta + \lambda + \alpha)}{\mu^2 (\alpha \beta + \alpha \lambda + \alpha + \lambda \beta)}
\end{cases}
\]

(4.45)

After some algebraic manipulations,
\[ E(L) = \sum_{k=1}^{\infty} k\pi_k = \sum_{k=1}^{\infty} k\pi_1 R^{k-1}e = \pi_1 (I - R)^{-2}e \]
\[ = \frac{\lambda((\alpha + \beta)^2 + \beta\mu)}{(\alpha + \beta)(\alpha\mu - (\alpha + \beta)\lambda)}, \quad (4.46) \]

subject to the stability condition in (4.11). The process is stable if the drift to the left is greater than the drift to the right, i.e,

\[ (\pi_0 + \pi_1)\lambda < \pi_1\mu \quad (4.47) \]

Finally, Little’s law yield the average waiting time

\[ E(Q) = E(L)/\lambda = \frac{\lambda((\alpha + \beta)^2 + \beta\mu)}{(\alpha + \beta)(\alpha\mu - (\alpha + \beta)\lambda)}, \quad (4.48) \]

subject to the stability condition

\[ \rho = \frac{\lambda}{\mu} < \frac{\alpha}{\alpha + \beta}, \quad \text{or} \quad \alpha > \frac{\rho}{1 - \rho}\beta \quad (4.49) \]

When \( \beta = 0 \), Eqn. (4.48) will reduce to \( 1/(\mu - \lambda) \) which is the queuing delay of an \( M/M/1 \) queue.

\[ E(Q) = \frac{1}{\mu - \lambda} \quad (4.50) \]

Interestingly, \( \rho/(1 - \rho) \), in Equation (4.48) is the average number of packets in an \( M/M/1 \) queue. Hence, the average queue length at each node can not exceed \( \alpha/\beta \) for a given traffic intensity \( \rho \). This is necessary for a disruptive queue stability. Figure 4.5 illustrates the delay performance for various values of \( \alpha \) and \( \beta \) as well as maximum load for each case under stationary condition.
Figure 4.5: Delay performance for various values of $\alpha$ and $\beta$.

### 4.1.3 Delay Variance; Jitter

Similar to the above derivation of average waiting time for disruptive queue, the variance of the queuing length, and hence the variance of waiting time can be formulated accordingly.

\[
Var(L) = E(L_q^2) - E^2(L_q)
\]

\[
= \sum_{k=1}^{\infty} k^2 \pi_k - \left( \frac{\lambda(\alpha^2 + \beta \mu + \beta^2 + 2 \alpha \beta)}{(\alpha + \beta)(\alpha \mu - \alpha \lambda - \lambda \beta)} \right)^2
\]

\[
= \sum_{k=1}^{\infty} k^2 \pi_1 R^{k-1} e - \left( \frac{\lambda(\alpha^2 + \beta \mu + \beta^2 + 2 \alpha \beta)}{(\alpha + \beta)(\alpha \mu - \alpha \lambda - \lambda \beta)} \right)^2
\]

\[
= \pi_1 \frac{I + R}{(I - R)^3} e - \left( \frac{\lambda(\alpha^2 + \beta \mu + \beta^2 + 2 \alpha \beta)}{(\alpha + \beta)(\alpha \mu - \alpha \lambda - \lambda \beta)} \right)^2
\]

\[
= \frac{\alpha(\alpha + \beta)^3 + \beta \lambda(\alpha + \beta)^2}{(\alpha + \beta)^2(\alpha \mu - \alpha \lambda - \lambda \beta)^2} \mu \lambda
\]

\[
+ \frac{[\beta(\lambda \mu + \alpha \lambda - 2 \lambda^2)(\alpha + \beta) + \alpha \beta \mu \lambda]}{(\alpha + \beta)^2(\alpha \mu - \alpha \lambda - \lambda \beta)^2} \mu \lambda
\]

When $\beta = 0$, Equation (4.51) becomes Equation (4.52) and that verifies with the
variance of the queuing length in an $M/M/1$ queue when there is no link failure.

$$Var(L) = \frac{\lambda \mu}{(\mu - \lambda)^2} = \frac{\rho}{(1 - \rho)^2} \quad (4.52)$$

### 4.2 Network of Disruptive Queues

We model the network as an interconnection of tandem open queuing systems in which packets move from queue $i$ to queue $j$ with probability $r_{ij}$ as discussed in section 2.4. The queue capacity at each node is assumed to be infinite, so there is no packet dropping.

To illustrate how the average end-to-end delay can be calculated, we considered the NASA’s space communication networks project, Edison Demonstration of Smallsat Networks (EDSN) [5], that was originally designed to act as an 8-node identical spacecraft with proper scheduling in which one satellite acts as a Captain and the rest act as Lieutenants. Figure 4.6 illustrates an 8-node network configuration in which node 8 acts as a Captain and others act as Lieutenants. Lieutenants were intended to communicate with the Captain, and the Captain was responsible for downlinking to an earth station. While the satellites were lost in the failure of the launch vehicle on November 3, 2015, subsequent efforts are underway in a follow-on

![Figure 4.6: EDSN nano satellites network [5].](image-url)
mission called Nodes. The Nodes satellites were developed by the EDSN project team and have a similar design.

Let the state of each link in Figure 4.6 be described by Equation (4.10) while satisfying the stability condition in Equation (4.11). The aggregate traffic at each node destined to the sink node 8 can be obtained from Equation (2.3) as,

$$\begin{bmatrix}
\gamma_1 \\
1/3 \gamma_1 + \gamma_2 \\
1/2 \gamma_1 + 1/2 \gamma_2 + \gamma_3 \\
1/2 \gamma_1 + 1/6 \gamma_2 + 1/3 \gamma_3 + \gamma_4 \\
1/4 \gamma_1 + 1/12 \gamma_2 + 1/6 \gamma_3 + 1/2 \gamma_4 + \gamma_5 \\
1/3 \gamma_1 + 2/3 \gamma_2 + 1/3 \gamma_3 + \gamma_6 \\
5/6 \gamma_1 + 2/3 \gamma_2 + 5/6 \gamma_3 + \gamma_4 + \gamma_5 + 1/2 \gamma_6 + \gamma_7 \\
\gamma_1 + \gamma_2 + \gamma_3 + \gamma_4 + \gamma_5 + \gamma_6 + \gamma_7 + \gamma_8
\end{bmatrix}$$

(4.53)

Without loss of generality, assume that the routing matrix $P$ generates the probabilistic alternate routing paths depicted in Figure 4.6 and calculated in Equation (4.54) as,

$$\begin{bmatrix}
q_1 + 1/3 q_2 + 1/2 q_3 + 1/2 q_4 + 1/4 q_5 + 5/6 q_7 + 1/3 q_6 \\
q_2 + 1/2 q_3 + 1/6 q_4 + 1/12 q_5 + 2/3 q_7 + 2/3 q_6 \\
q_3 + 1/3 q_4 + 1/6 q_5 + 5/6 q_7 + 1/3 q_6 \\
q_4 + 1/2 q_5 + q_7 \\
q_5 + q_7 \\
q_6 + 1/2 q_7 \\
q_7 \\
0
\end{bmatrix}$$

(4.54)

where, $q_i = \frac{\lambda_i ((\alpha_i + \beta_i)^2 + \beta_i \mu_i)}{(\alpha_i + \beta_i) (\alpha_i \mu_i - (\alpha_i + \beta_i) \lambda_i)} \quad 1 \leq i \leq n$, 

$$\text{subject to: } \alpha_i > \frac{\rho_i}{1 - \rho_i} \beta_i$$

(4.55)
\( q_8 = 0 \) since it is a sink node. For the sake of mathematical derivation, we assume \( \alpha_i = \alpha \) and \( \beta_i = \beta \). Substituting \( q_i \) from Equation (4.55) in Equation (4.54) gives the end-to-end delay vector.

\[
\vec{D} = \begin{bmatrix}
(\alpha+\beta)^2 + \beta \mu_1 \\
(\alpha+\beta)(\alpha \mu_1 - (\alpha+\beta) \lambda_1) \\
(\alpha+\beta)^2 + \beta \mu_2 \\
(\alpha+\beta)(\alpha \mu_2 - (\alpha+\beta) \lambda_2) \\
(\alpha+\beta)^2 + \beta \mu_3 \\
(\alpha+\beta)(\alpha \mu_3 - (\alpha+\beta) \lambda_3) \\
(\alpha+\beta)^2 + \beta \mu_4 \\
(\alpha+\beta)(\alpha \mu_4 - (\alpha+\beta) \lambda_4) \\
(\alpha+\beta)^2 + \beta \mu_5 \\
(\alpha+\beta)(\alpha \mu_5 - (\alpha+\beta) \lambda_5) \\
(\alpha+\beta)^2 + \beta \mu_6 \\
(\alpha+\beta)(\alpha \mu_6 - (\alpha+\beta) \lambda_6) \\
(\alpha+\beta)^2 + \beta \mu_7 \\
(\alpha+\beta)(\alpha \mu_7 - (\alpha+\beta) \lambda_7) \\
0
\end{bmatrix}
\] (4.57)

4.3 Numerical and Simulation Results

To better understand the effectiveness of the model, we have simulated several network topologies including trees, linear networks, and mesh networks. In this section we present the theoretical derivation for the Smallsat Networks (EDSN) of NASA [5] and two widely used networks, trees and meshes. We also compared the delay performance for each model under various traffic load and link disruption distribution parameters. In terms of routing, the mesh network and EDSN network offer alternate paths between any two nodes, while the tree provides a unique path between any two nodes. For each network scenario we developed an end-to-end delay formulation along with their corresponding simulation models to confirm the accuracy of the approximation model. Various values for \( \alpha, \beta, \lambda, \mu \), and packet size have been tested subject to the stationary condition described earlier.
4.3.1 EDSN Network Performance

Figure 4.7 illustrates the theoretical and simulation comparison of four different scenarios for EDSN network. Figure 4.7(a) shows the delay performance in theory and simulation when there is no link failure ($\beta = 0$). The delay becomes independent of $\alpha$ when $\beta = 0$ (see Equation (4.50)). Figure 4.7(b) shows the impact of increasing $\beta$. The steady state requires that $\alpha > \beta$. Figure 4.7(c) shows that increasing $\alpha$ reduces delay equally in both theory and simulation. Finally, increasing $\beta$ in Figure 4.7(d) increases the delay equally in both theory and simulation. In terms of simulation parameters, for the purpose of consistency, we chose packet lengths to be geometrically distributed with mean 250 bytes. The channel speed varied from 250KB to 2.5MB in various experiments, not all shown here.
4.3.2 Tree Network

A widely used hierarchical network topology is a tree network similar to the network in Figure 4.8. Consider a network of disruptive queues that its long-term probabilistic connectivity form a tree with exogenous traffic arrival rate $\gamma_i$ and aggregate traffic $\lambda_i$ at each node. Let the state of each link in Figure 4.8 be described by Equation (4.10)

![Figure 4.8: A sink tree network with disruptive links.](image)

and satisfies the stability condition in Equation (4.11). The aggregate traffic vector
can be obtained from Equation (2.3) as,

\[
\overrightarrow{\lambda} = \begin{bmatrix}
\frac{(\gamma_8 + \gamma_9 + \gamma_{10}) \alpha^2}{(\alpha + \beta)^2} + \frac{(\gamma_4 + \gamma_5 + \gamma_7) \alpha^2}{(\alpha + \beta)^2} + \frac{(\gamma_2 + \gamma_3) \alpha}{\alpha + \beta} + \gamma_1 \\
\frac{(\gamma_4 + \gamma_5) \alpha^2}{(\alpha + \beta)^2} + \gamma_2 \\
\frac{(\gamma_6 + \gamma_7) \alpha^2}{(\alpha + \beta)^2} + \gamma_3 \\
\frac{(\gamma_8 + \gamma_9) \alpha}{\alpha + \beta} + \gamma_4 \\
\gamma_5 \\
\gamma_6 + \frac{\gamma_{10} \alpha}{\alpha + \beta} \\
\gamma_7 \\
\gamma_8 \\
\gamma_9 \\
\gamma_{10}
\end{bmatrix}
\] (4.58)

Based on the aggregate traffic at each node, from Equation (4.48), the expected queuing delay can be calculated as,
\[
\bar{q} = \begin{bmatrix}
0 \\
\frac{\alpha^2 + \beta \mu_2 + \beta^2 + 2 \alpha \beta}{(\alpha + \beta)(\alpha \mu_2 - \alpha \Gamma_2 - \Gamma_2 \beta)} \\
\frac{\alpha^2 + \beta \mu_3 + \beta^2 + 2 \alpha \beta}{(\alpha + \beta)(\alpha \mu_3 - \alpha \Gamma_3 - \Gamma_3 \beta)} \\
\frac{\alpha^2 + \beta \mu_4 + \beta^2 + 2 \alpha \beta}{(\alpha + \beta)(\alpha \mu_4 - \alpha \Gamma_4 - \Gamma_4 \beta)} \\
\frac{\alpha^2 + \beta \mu_5 + \beta^2 + 2 \alpha \beta}{(\alpha + \beta)(\alpha \mu_5 - \alpha \gamma_5 - \gamma_5 \beta)} \\
\frac{\alpha^2 + \beta \mu_6 + \beta^2 + 2 \alpha \beta}{(\alpha + \beta)(\alpha \mu_6 - \alpha \Gamma_6 - \Gamma_6 \beta)} \\
\frac{\alpha^2 + \beta \mu_7 + \beta^2 + 2 \alpha \beta}{(\alpha + \beta)(\alpha \mu_7 - \alpha \gamma_7 - \gamma_7 \beta)} \\
\frac{\alpha^2 + \beta \mu_8 + \beta^2 + 2 \alpha \beta}{(\alpha + \beta)(\alpha \mu_8 - \alpha \gamma_8 - \gamma_8 \beta)} \\
\frac{\alpha^2 + \beta \mu_9 + \beta^2 + 2 \alpha \beta}{(\alpha + \beta)(\alpha \mu_9 - \alpha \gamma_9 - \gamma_9 \beta)} \\
\frac{\alpha^2 + \beta \mu_{10} + \beta^2 + 2 \alpha \beta}{(\alpha + \beta)(\alpha \mu_{10} - \gamma_{10} \alpha - \gamma_{10} \beta)}
\end{bmatrix}
\]

where, \(\Gamma_1 = \sum_{i=1}^{10} \gamma_i\), \(\Gamma_2 = \gamma_2 + \gamma_4 + \gamma_5 + \gamma_8 + \gamma_9\), \(\Gamma_3 = \gamma_3 + \gamma_6 + \gamma_7 + \gamma_{10}\), \(\Gamma_4 = \gamma_4 + \gamma_8 + \gamma_9\), \(\Gamma_6 = \gamma_6 + \gamma_{10}\), and \(q_1 = 0\) since it is a sink node. Now, the average end-to-end delay
for each path can be easily computed as,

$$\overrightarrow{p} = \begin{bmatrix} 0 \\ q_2 \\ q_3 \\ q_2 + q_4 \\ q_2 + q_5 \\ q_3 + q_6 \\ q_3 + q_7 \\ q_2 + q_4 + q_8 \\ q_2 + q_4 + q_9 \\ q_3 + q_6 + q_{10} \end{bmatrix}$$

(4.60)

where, $q_i = \frac{\lambda_i ((\alpha + \beta)^2 + \beta \mu_i)}{(\alpha + \beta)(\alpha \mu_i - (\alpha + \beta) \lambda_i)}$, $1 \leq i \leq n$.  

(4.61)

subject to: $\alpha > \frac{\rho_i}{1 - \rho_i} \beta$,  

(4.62)

assuming $\alpha_i = \alpha$ and $\beta_i = \beta$. $q_8 = 0$ since it is a sink node. One can calculate the average end-to-end delay by substituting entries of Equation (4.59) in Equation (4.60). Figure 4.9 illustrates the theoretical and simulation comparison of four scenarios for the tree network in Figure 4.8, without ($\beta = 0$) and with ($\beta = \{1, 2\}$) link failure for $\alpha = \{2, 3\}$, respectively. Again, the delay in theory and simulation are very close, within the conventional bounds of statistical significance. As network load ($\rho$) increases, the gap between the theoretical model and simulation model slightly widens. This is due to the exponentially generated random variables for inter-arrival times and packet size that the theory provides asymptotically. We have noticed that
in several experiments smaller packets and higher channel speeds shrink these gaps. The theoretical model is based on fine-grained packet forwarding and that keeps the curve slightly below the simulation model. If an outgoing link is unavailable, then a packet stays in the queue until a route becomes available.

### 4.3.3 Meshes

Mesh networks are widely used as non-heirarchical networks for various applications ranging from smart highways to deep space communication and remote sensing. A typical linear mesh topology is shown in Figure 4.10.

Figure 4.10 has two new characteristics that two previous networks don’t have; (i) a larger diameter, and (ii) it offers relatively more end-to-end path diversity that would expose significant end-to-end delays under various link intermittency. To study the worst case analysis, we assume all nodes send their traffic to a
sink node (9). This allows us to pinpoint the effect of disruptive tandem queues on an end-to-end average delay. $\vec{\lambda}$ in Equation (4.63) shows the aggregate traffic vector that corresponds to the aggregate traffic each node receives from various nodes. We uniformly distributed the arriving packets on the outgoing links provided that the outgoing link is in connection state. We did not have this condition for the tree due to its unique path property. If an outgoing link is unavailable, then a packet stays in the queue until a route becomes available.

$$\vec{\lambda} = \begin{bmatrix} \gamma_1 \\ 1/2 \gamma_1 + \gamma_2 \\ 3/4 \gamma_1 + 1/2 \gamma_2 + \gamma_3 \\ 5/8 \gamma_1 + 3/4 \gamma_2 + 1/2 \gamma_3 + \gamma_4 \\ \frac{11}{16} \gamma_1 + 5/8 \gamma_2 + 3/4 \gamma_3 + 1/2 \gamma_4 + \gamma_5 \\ \frac{21}{32} \gamma_1 + \frac{11}{16} \gamma_2 + 5/8 \gamma_3 + 3/4 \gamma_4 + 1/2 \gamma_5 + \gamma_6 \\ \frac{11}{16} \gamma_1 + \frac{21}{32} \gamma_2 + 11/16 \gamma_3 + 5/8 \gamma_4 + 3/4 \gamma_5 + 1/2 \gamma_6 + \gamma_7 \\ \frac{85}{128} \gamma_1 + \frac{43}{64} \gamma_2 + \frac{21}{32} \gamma_3 + \frac{11}{16} \gamma_4 + 5/8 \gamma_5 + 3/4 \gamma_6 + 1/2 \gamma_7 + \gamma_8 \\ \gamma_1 + \gamma_2 + \gamma_3 + \gamma_4 + \gamma_5 + \gamma_6 + \gamma_7 + \gamma_8 + \gamma_9 \end{bmatrix}$$  

$\vec{p}$ in Equation (4.64) gives a vector of probabilistic routes towards the destination when alternate routing is considered. $q_i$ corresponds to the participation of node $i$.

Figure 4.10: A mesh network.
along a path from a source (node $i = 1, 2, \cdots, 8$) to the destination (node 9).

\[
\bar{p} = \begin{bmatrix}
q_1 + 1/2 q_2 + 3/4 q_3 + 5/8 q_4 + 11/16 q_5 + 21/32 q_6 + 43/64 q_7 + 85/128 q_8 \\
q_2 + 1/2 q_3 + 3/4 q_4 + 5/8 q_5 + 11/16 q_6 + 21/32 q_7 + 43/64 q_8 \\
q_3 + 1/2 q_4 + 3/4 q_5 + 5/8 q_6 + 11/16 q_7 + 21/32 q_8 \\
q_4 + 1/2 q_5 + 3/4 q_6 + 5/8 q_7 + 11/16 q_8 \\
q_5 + 1/2 q_6 + 3/4 q_7 + 5/8 q_8 \\
q_6 + 1/2 q_7 + 3/4 q_8 \\
q_7 + 1/2 q_8 \\
q_8 \\
0
\end{bmatrix}
\] (4.64)

Now, we can apply Equation (4.48) to each $q_i$ in Equation (4.64) to get the average end-to-end delay. Figure 4.11 illustrates the theoretical and simulation comparison of four scenarios for the mesh network in Figure 4.10, without ($\beta = 0$) and with ($\beta = \{1, 2\}$) link failure for $\alpha = \{2, 3\}$. Again, the delay in theory and simulation are very close, within the conventional bounds of statistical significance. The gaps are closer than those in the tree due to alternate paths availability. As before, the steady state requires that $\alpha > \beta$.

In terms of simulation parameters, for the purpose of consistency, we chose packet lengths to be geometrically distributed with mean 250 bytes. The channel speed varied from 250KB to 2.5MB in various experiments, not all shown here. Considering the stationary condition in Equation (4.48), in which $\alpha > \beta$ and $\alpha/\beta > 1/(1 - \rho)$, we run a number of experiments for $\alpha = \{1, 2, \cdots\}$ and $\beta = \{0, 1, \cdots\}$, not all were shown in this section.
Figure 4.11: Delay comparison for the mesh network in Figure 4.10 for various values of $\alpha$ and $\beta$.

4.4 Summary

Networks with disruptive links have various applications including sensor networks, battle-field networks, satellite networks, and mobile ad-hoc networks. Traditional open queuing models fail to describe the average end-to-end delay performance in these networks. This chapter describes how a disruptive link can be modeled in these networks and how the average end-to-end delay and its variance can be formulated in closed forms given a long-term probabilistic routing matrix. While the model is theoretically sound, simulation results for various network topologies confirm with the theoretical result within the the bounds of statistical significance. While Jackson tandem queuing network [57] along with exponentially distributed contact time and inter-contact time may impose some limitations to generalize the model, studies [9] have shown that for random way-point [10] and the random direction [11], mobility models exhibit an exponentially distributed inter-contact times with small
scale mobility. For large scale mobility, experiments have shown that the tail of the inter-contact time distribution follows a power law decay in some finite range, but exhibits an exponential decay afterward [12]. The simulation models developed in this study strongly confirm with the Kleinrock independence approximation [60] that merging several packet streams on a transmission line has an effect akin to restoring the independence of inter-arrival times and packet lengths, thus an $M/M/1$ queuing model can be used. The limitation of the model is that one has to have the asymptotic routing matrix to obtain the long-term end-to-end approximated average delay. Normally, this is available in network systems with no or limited mobility and in networks with deterministic (periodic) link disconnectivity. It would be interesting to see how the model can be extended to cover a general traffic distribution with general contact and inter-contact times.
Chapter 5

End to End Delay Analysis based on Mean First Contact Time

In this chapter, we have developed a stochastic model that is based on an open queuing system that approximates the E2E delay in closed form for DTN. To calculate E2E delay the model incorporates the impact of traffic load, arrival rate, service rate, link connectivity and the underlying routing algorithm. The model is generic enough to incorporate any DTN routing (unicast or multicast or anycast) algorithm. In this chapter first, we deploy a traffic distribution model for DTN that incorporates dissemination of packets under various DTN routing algorithms. Second, we employ a strong, yet simple, open queuing system based on Jackson’s theorem discussed in section 2.4 that facilitate the computation of E2E delay performance under various DTN network conditions. The rest of the chapter is organized as follows. The open tandem queuing model that describes packet migration along a path towards a destination along with the effect of aggregate traffic on each hop is described in chapter 2 section 2.4. Supporting mobility and closed form solutions are discussed in Section 5.1. Section 5.2 incorporates expected connection time in closed form solutions. Section 5.3 describes numerical results along with simulation of a network scenario.
Conclusions and remarks are given in Section 5.4.

5.1 Supporting Mobility

Mobility is an important aspect of DTN and MANET. Mobility models can be classified into (i) deterministic models (e.g., urban traffic models), (ii) semi-deterministic models (e.g., Column model, Pursue models), and (iii) random models (e.g., Brownian Motion and Waypoint models) [30].

Mobility can change the state of the routing matrix $R$ deterministically or probabilistically as defined by one of the above mobility models. $R$ contains two pieces of information: adjacency (connectivity) matrix and weighted links that represent the proportion of traffic routed through each link. Let $N = \{v_1, v_2, \cdots , v_n\}$ be the set of nodes and $L = \{\ell_{ij}; 1 \leq i, j \leq n, i < j\}$ the set of links. We define the adjacency matrix $A = [a_{ij}]$ from the routing matrix $R = [r_{ij}]$ as,

$$a_{ij} = \begin{cases} 1 & \text{if } r_{ij} \geq 0 \\ 0 & \text{otherwise} \end{cases} \quad 1 \leq i, j \leq n \quad (5.1)$$

We assume undirected graphs that represent symmetric connectivity in term of bitrate between two connected nodes. Therefore, $|L| = n(n-1)/2$, represents the size of transition states. Hence, there are potentially $2^{|L|}$ states the network can go through. Let $S = \{0, 1, 2, \cdots , 2^{|L|}-1\}$ be the set of possible states the network can go through as the result of any mobility pattern, where each state can be represented by a binary number $[s_0s_2\cdots s_{|L|-1}]$, $s_i \in \{0, 1\}$, $s_i = 1$ corresponds to link $i$ being in ON state, and $s_i = 0$ corresponds to link $i$ being in OFF state, $0 \leq i \leq |L| - 1$. Therefore, $[00\cdots 0]$ represents an edge-less graph and $[11\cdots 1]$ represents a fully-connected graph. The transition from OFF state to ON state on link $i$ occurs with probability $\beta_i$ and the transition from ON state to OFF state on link $i$ occurs with probability $\alpha_i$. In DTN
terminology, ON and OFF periods are called contact time and inter-contact time, respectively. In discrete time, OFF and ON periods are geometrically distributed with the mean $1/\alpha_i$ and $1/\beta_i$, respectively. In continuous time, OFF and ON periods are exponentially distributed with the mean $1/\alpha_i$ and $1/\beta_i$, respectively. Figure 5.1 represents the transition probabilities. The transition probability matrix and the transition rate matrix for the state transition in Figure 5.1 are $P_i$ and $Q_i$, respectively.

$$
P_i = \begin{bmatrix}
1 - \alpha_i & \alpha_i \\
\beta_i & 1 - \beta_i
\end{bmatrix}, \quad Q_i = \begin{bmatrix}
-1/\alpha_i & 1/\alpha_i \\
1/\beta_i & -1/\beta_i
\end{bmatrix}
$$

(5.2)

Now, consider the network in Figure 5.2 with 16 possible states of which under mobility 5 states preserve network connectivity and 11 states cause network disconnectivity. The set of possible states are divided into two groups; connected states $S_c$, in which the underlying graph is connected, and disconnected states, $S_d$, in which the underlying graph is not connected. The latter cause intermittency in DTNs. When a link disruption does result in network disconnectivity, such as those in states

$$
S_c = \{7, 11, 13, 14, 15\}, \text{ then the average E2E delay can be computed by Equation (2.7). However, for the rest of the states (} S_d, \text{ the E2E delay should be computed}
$$

Figure 5.1: State transition of a link.

Figure 5.2: (a) and (b) are connected states and (c) is a disconnected state.
differently. This is discussed in Section 5.2. Figure 5.3 illustrates the state transitions for the network in Figure 5.2(a). Yellow nodes correspond to disconnected states and blue nodes correspond to connected states. A link goes off with probability $\alpha_i = \alpha$ and comes back on with probability $\beta_i = \beta$, according to the state transition diagram in Figure 5.1. Let $\pi_i$ be the probability that the network is in state $i$, $0 \leq i < 2^{|L|}$.

<table>
<thead>
<tr>
<th>Connected states ($S_c$)</th>
<th>Disconnected states ($S_d$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>7 [ 0 1 1 1 ]</td>
<td>0 [ 0 0 0 0 ]</td>
</tr>
<tr>
<td>11 [ 1 0 1 1 ]</td>
<td>1 [ 0 0 0 1 ]</td>
</tr>
<tr>
<td>13 [ 1 1 0 1 ]</td>
<td>2 [ 0 0 1 0 ]</td>
</tr>
<tr>
<td>14 [ 1 1 1 0 ]</td>
<td>3 [ 0 0 1 1 ]</td>
</tr>
<tr>
<td>15 [ 1 1 1 1 ]</td>
<td>4 [ 0 1 0 0 ]</td>
</tr>
<tr>
<td>5 [ 0 1 0 1 ]</td>
<td>6 [ 0 1 1 0 ]</td>
</tr>
<tr>
<td>8 [ 0 1 1 0 ]</td>
<td>9 [ 1 0 0 1 ]</td>
</tr>
<tr>
<td>10 [ 1 0 1 0 ]</td>
<td>12 [ 1 1 0 0 ]</td>
</tr>
</tbody>
</table>

Table 5.1: Connected and disrupted states

where, $\sum_{i=0}^{2^{|L|}-1} \pi_i = 1$. The state diagram in Figure 5.3 gives the balance equations A.1. In steady state, the set of equations in (A.1) can be solved symbolically by a

Figure 5.3: State transition digram.
linear solver, and the steady state probabilities can be computed as,

\[
\begin{align*}
\pi_0 &= \beta^4/d \\
\pi_1 &= \beta^3\alpha/d \\
\pi_2 &= \beta^3\alpha/d \\
\pi_3 &= \beta^2\alpha^2/d \\
\pi_4 &= \beta^3\alpha/d \\
\pi_5 &= \beta^2\alpha^2/d \\
\pi_6 &= \beta^2\alpha^2/d \\
\pi_7 &= \beta\alpha^3/d \\
\pi_8 &= \beta^3\alpha/d \\
\pi_9 &= \beta^2\alpha^2/d \\
\pi_{10} &= \beta\alpha^2/d \\
\pi_{11} &= \beta\alpha^3/d \\
\pi_{12} &= \beta^2\alpha^2/d \\
\pi_{13} &= \beta\alpha^3/d \\
\pi_{14} &= \beta\alpha^3/d \\
\pi_{15} &= \alpha^4/d
\end{align*}
\] (5.3)

where, \( d = (\beta + \alpha)^4 \).

### 5.2 Expected Connection Time

While the expected delay for connected states \( (S_c = \{7, 11, 13, 14, 15\}) \) can be directly computed from Equation (2.7), the expected delay for disconnected states \( (S_d = \{0, 1, 2, 3, 4, 5, 6, 8, 9, 10, 12\}) \) include the expected connection time to reach a connected state. Before calculating expected connection time we will describe some formal terms and properties related to Markov Chains. Markov chain is a process in which the outcome of a given experiment can affect the outcome of the next experiment.

A Markov chain can be described as follows, we have a set of discrete states, \( S = \{i, j, \cdots\} \) for example state space discussed in Section 5.1. The chain starts in one of these states and moves successively from one state to another. Each move is called a step. If the chain is currently in state \( i \), then it moves to state \( j \) at the next step with a probability denoted by \( p_{ij} \), and this probability does not depend upon which states the chain was in before the current state. The probability \( p_{ij} \) is called transition probability. The process can remain in the state it is in, and this occurs with probability \( p_{ii} \). An initial probability distribution, defined on \( S \), specifies the
starting state. Usually this is done by specifying a particular state as the starting state. For example transition matrix shown in Equation 5.2, shows given that the link is in ON state, transitional probability of going from ON state to OFF state is given by probability $\alpha_i$ and staying in ON state is given by probability $1 - \alpha_i$. One of the important property of transition matrix is, sum of all elements in a row adds up to 1.

**Irreducible** or **Ergodic** Markov Chains are special types of Markov chains where we can reach any state $j$ from a current state $i$ i.e. all states are transient. Counter to these chains are **Absorbing Markov Chains** in which, a state $i$ of a Markov chain is called absorbing if it is impossible to leave it (i.e., $p_{ii} = 1$). A Markov chain is absorbing if it has at least one absorbing state, and if from every state it is possible to go to an absorbing state (not necessarily in one step). In an absorbing Markov chain, a state which is not absorbing is called transient. We build our model based on **irreducible** Markov chains.

State $i$ leads to state $j$ ($ij$) if there exist $t \geq 1$ such that $P_{ij}^{(t)} > 0$, where $P^{(t)}$ is the $t$-step transition probability matrix, i.e., $P^{(t)} = P^t$. State $i$ communicates with state $j$ ($ij$) if $i$ leads to $j$ and $j$ leads to $i$. If state $i$ and state $j$ communicate, then there exists $t \geq 1$ and $r \geq 1$ such that $P_{ij}^{(t)} > 0$ and $P_{ji}^{(r)} > 0$.

We refer to **irreducible** discrete-time Markov chain $X_0, X_1, X_2, \cdots$ which has the state space $S$, and that could be finite set. Let $P = [p_{ij}]$ be the transition matrix representing the state transition diagram in Figure 5.3. We have to calculate average number of steps to reach from a disconnected state $i$, where $i \in S_d$, to a connected state $j$, where $j \in S_c$. First, we define a few properties for the above discrete-time Markov chain [74].

**property 1:** Let $P^n$ represent power of the matrix. When $n$ becomes very large, i.e. $n \to \infty$, we get limiting matrix as,

$$\Pi = \lim_{n \to \infty} P^n$$  \hfill (5.4)
the rows of Π are identical. This property is known as large time (long term) behavior of Markov chain and it means after large number of steps the matrix converges.

**Property 2:** Since states represented by P are transient, then \( P^n \to 0 \), where \( n \) is power of \( P \). Hence, \( I-P \) is an invertible matrix where \( I \) is the identity matrix. So, we can have a matrix, \( M = (I-P)^{-1} \).

Let \( j \) be a transient state and consider \( Y_j \), the total number of visits to \( j \),

\[
Y_j = \sum_{n=0}^{\infty} X_n = j
\]  

(5.5)

Since, \( j \) is transient, \( Y_j < \infty \) with probability 1. Suppose \( X_0 = i \), where \( i \) is another transient state. Then,

\[
E(Y_j | X_0 = i) = E\left[ \sum_{n=0}^{\infty} X_n = j | X_0 = i \right] = \sum_{n=0}^{\infty} PX_n = j | X_0 = i.
\]

(5.6)

In other words, \( E(Y_j | X_0 = i) \) is the \((i, j)\) entry of the matrix, \( I+P+P^2+\cdots \).

With simple calculations we can show that,

\[
(I + P + P^2 + \cdots)(I - P) = I
\]

(5.7)

\[\Rightarrow (I + P + P^2 + \cdots) = (I - P)^{-1} = M.\]

(5.8)

We have just shown that the expected number of visits to \( j \) starting at \( i \) is given by \( M_{ij} \), the \((i, j)\) entry of \( M \).

Assuming above properties and derived expressions we define first connection time and mean first connection time as follows. The first connection time \( F_{ij} \) is the random variable representing the smallest number of timesteps for reaching state \( j \) for the first time, given that the system
was initially in state $i$. That is:

$$F_{ij} = \min\{t \geq 1 : X_t = j \mid X_0 = i\} \quad i, j \in S$$

The mean first connection time from state $i$ to state $j$ is the expected number of time-steps for reaching state $j$ for the first time, given that initially the chain was in state $i$. The mean first connection times, $m_{ij}$, can be defined as,

$$m_{ij} = E(F_{ij}) = \sum_{k=1}^{\infty} k \Pr[F_{ij} = k] \quad (5.9)$$

Matrix $M = [m_{ij}]$ is called the mean first connection time matrix of the chain. The mean first connection time gives us information about the short range behavior of the chain, i.e., how long one can expect to get to the state of connectivity, given that the current state is a disconnected state.

The mean connection time, $[m_i]$, for the state transition diagram in Figure 5.3 has been calculated accordingly in Table 5.2, where, $i \in S_d = \{0, 1, 2, 3, 4, 5, 6, 8, 9, 10, 12\}$ and $j \in S_c = \{7, 11, 13, 14, 15\}$. Note that $\overline{m_i}$ is averaged over column $j$, and that represents the average connection time from state $i$ to any state $j$.

Now, the average E2E delay can be computed by,

$$\overline{D} = \sum_{j \in S_c} \pi_j \overline{D}_j + \sum_{i \in S_d} \pi_i (\overline{D}_i + \overline{m_i}) \quad (5.10)$$

where, $\overline{D}_i$ is obtained from Equation (2.7) and $m_i$ from Equation (5.9).
Table 5.2: Mean Connection Time ($m_i$)

<table>
<thead>
<tr>
<th>$S_d$</th>
<th>$S_{c{7, 11, 13, 14, 15}}$</th>
<th>$\alpha = 0.5$</th>
<th>$\alpha = 0.7$</th>
<th>$\alpha = 0.6$</th>
<th>$\alpha = 0.8$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>3.13</td>
<td>5.19</td>
<td>7.54</td>
<td>2.40</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2.66</td>
<td>4.58</td>
<td>6.68</td>
<td>2.04</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>2.63</td>
<td>4.59</td>
<td>6.66</td>
<td>2.04</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1.83</td>
<td>3.36</td>
<td>5.00</td>
<td>1.41</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>2.63</td>
<td>4.58</td>
<td>6.67</td>
<td>2.04</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>1.81</td>
<td>3.37</td>
<td>4.98</td>
<td>1.40</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>1.80</td>
<td>3.37</td>
<td>4.97</td>
<td>1.41</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>2.66</td>
<td>4.57</td>
<td>6.66</td>
<td>2.04</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>1.83</td>
<td>3.35</td>
<td>4.98</td>
<td>1.41</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>1.84</td>
<td>3.36</td>
<td>4.98</td>
<td>1.41</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>1.83</td>
<td>3.35</td>
<td>5.00</td>
<td>1.42</td>
<td></td>
</tr>
<tr>
<td>Mean</td>
<td>2.26</td>
<td>3.95</td>
<td>5.83</td>
<td>1.71</td>
<td></td>
</tr>
</tbody>
</table>

5.3 Simulation Results

We have tested the model for various network topologies using both the model described in this chapter as well simulation models. Specifically, we used DTN simulator $ONE$ [65] to compare theoretical results with simulation results. In these experiments, we have maintained an unsaturated state for each node (queue) in the system and maintained the stability condition for each node under composite traffic aggregation by conforming with Equation (2.4). We used $First Contact$ routing algorithm [48] which forwards a packet along an edge chosen randomly among all the current contacts. Anti-cycling and packet duplication elimination have been provisioned. The routing protocol is a single copy protocol in which at any given time $t$, there exists only one copy of the message in the network. We set up the simulation parameters with channel-speed (250 KB/s), packet size (250 KB), service time ($\mu = 1$ s), traffic load ($0.1 \leq \rho \leq 0.9$), ON/OFF probabilities ($\alpha = \beta = 0.5$). We have run experiments long enough, to meet
steady state requirement of the model.

The results obtained from the model in Sections 5.1 and 5.2, have been compared with the simulation results obtained under various traffic load conditions. We have compared the results for connected states ($S_c$) and all states ($S_c \cup S_d$), illustrated in Figure 5.4 (a) and (b), respectively.

Figure 5.4: E2E delay analysis of four node topology.

We also tested our model with some real life topologies (Figure 5.5 and 5.6) and results are shown in Figure 5.7. The reason the model gives a slightly better performance than the simulation experiments in Figure 5.4 and 5.7, is the result of finer timing used in model ($\Delta t \to 0$) and exponentially generated pseudorandom numbers generated by the simulation. This effect is slightly greater when we increase number of nodes and for higher loads, the analytical results show better performance than simulation. Still E2E delay approximation is very close to simulation results.

5.4 Summary

In this chapter, we have developed a new model that can be used to approximate the average E2E delay in DTNs. The model is based on an open queuing system with Markovian process using Jackson’s theorem [57]. We have augmented the model to incorporate various types of mobility mod-
els. The model is quite general in the sense that it can accommodate various routing algorithms as long as the algorithms generates a routing matrix which is irreducible on each time interval. The limitation of the model relates to the scenarios when mobility pattern is highly random and causes infinite disconnectivity. The model efficiently calculates the transition time between various graph disconnectivity states and those which are in the states of connected graphs. This model can be further extended to explore relationship between mobility and E2E delay approximation in

Figure 5.5: Multiple Spacecrafts as in [4]

Figure 5.6: Spacecraft Topology similar to [6]

Figure 5.7: E2E delay analysis of Nasa spacecraft topologies
various scenarios.
Chapter 6

Conclusion and Future Work

*G-Network*, Networks with disruptive links have various applications including sensor networks, battle-field networks, satellite networks, and mobile ad-hoc networks. Traditional open queuing models fail to describe the average end-to-end delay performance in these network. This research work describes three different approaches that how a disruptive link can be modeled in these networks and how the average end-to-end delay and its variance can be formulated in closed forms given a long-term probabilistic routing matrix. Chapter 4 proposes a disruptive link model for individual queue and then for network of queues, simulation results for various network topologies confirm with the theoretical result within the the bounds of statistical significance. Simulation results show that the model proposed in Chapter 4 gives better approximation than model in chapter 3. But the model in later is simpler than previous. Whereas, chapter 5 builds a model to incorporate any mobility model and routing algorithm by exhausting all possible states the network can go through which may cause state space explosion problem. While Jackson tandem queuing network [57] along with exponentially distributed contact time and inter-contact time may impose
some limitations to generalize the model, studies [9] have shown that for
random way-point [10] and the random direction [11], mobility models
exhibit an exponentially distributed inter-contact times with small scale
mobility. For large scale mobility, experiments have shown that the tail of
the inter-contact time distribution follows a power law decay in some finite
range, but exhibits an exponential decay afterward [12]. The simulation
models developed in our models strongly confirm with the Kleinrock in-
dependence approximation [60] that merging several packet streams on a
transmission line has an effect akin to restoring the independence of inter-
arrival times and packet lengths, thus an $M/M/1$ queuing model can be
used. The limitation of our models is that one has to have the asymptotic
routing matrix to obtain the long-term end-to-end approximated average
delay. Normally, this is available in network systems with no or limited
mobility and in networks with deterministic (periodic) link disconnectiv-
ity. It would be interesting to see, if possible, how these model can be
extended to cover a general traffic distribution with general contact and
inter-contact times.

6.1 G-network

An alternate approach to model DTN using open queuing network could
be $G$-network but traffics have different characteristics. A $G$-network
(generalized queueing network or Gelenbe network) is an open network of
$G$-queues. It is a model for queuing systems with specific control func-
tions, such as traffic re-routing or traffic destruction. In this network
traffic types are as follows: i) positive customers, which arrive from other
queues or arrive externally as Poisson arrivals, and obey standard service
and routing disciplines as in conventional network models.
ii) negative customers, which arrive from another queue, or which arrive externally as Poisson arrivals, and remove customers in a non-empty queue, representing the need to remove traffic when the network is congested, including the removal of batches of customers.

iii) triggers, which arrive from other queues or from outside the network, and which displace customers and move them to other queues [75]. The customers in DTN context are packets or bundles.

Currently, DTN doesn’t incorporate traffic with such characteristics and it is mostly modeled assuming traffic nature as positive customers only. A powerful property of \textit{G-networks} is that they are universal approximators for continuous and bounded functions. The approximation of a function $f(X)$ of an input vector $X$ by some other function $F(w, x)$ having fixed number of parameters denoted by the vector $w$. The parameters $w$ are chosen so as to achieve the best possible approximation of the function $f$, so that they can be used to approximate quite general input-output behaviors [75].

6.2 Queues with breakdowns and customer discouragement

In theory models for queue breakdowns and customer discouragement exist where a customer may get discouraged when a break down happens and leaves the system with a constant probability, independently of other customers. The system alternates between working and repair periods [38]. From DTN perspective this model can be implemented for deciding packet drop policy and evaluate E2E delay in DTN. Because the criteria for deciding packet drop policy in DTN is an open research problem. It is
relevant in context of research problems those involve improving network throughput, control congestion and improve performance of DTN routing algorithms.
Bibliography


102


Appendix

DTN details

Balanced equation for the state diagram in Figure 5.3.

\[
\begin{array}{l}
(4\beta)\pi_0 = \alpha\pi_1 + \alpha\pi_2 + \alpha\pi_4 + \alpha\pi_8 \\
(3\beta + \alpha)\pi_1 = \beta\pi_0 + \alpha\pi_3 + \alpha\pi_5 + \alpha\pi_9 \\
(3\beta + \alpha)\pi_2 = \beta\pi_0 + \alpha\pi_3 + \alpha\pi_6 + \alpha\pi_{10} \\
(2\beta + 2\alpha)\pi_3 = \beta\pi_1 + \beta\pi_2 + \alpha\pi_7 + \alpha\pi_{11} \\
(3\beta + \alpha)\pi_4 = \beta\pi_0 + \alpha\pi_5 + \alpha\pi_6 + \alpha\pi_{12} \\
(2\beta + 2\alpha)\pi_5 = \beta\pi_1 + \beta\pi_4 + \alpha\pi_7 + \alpha\pi_{13} \\
(2\beta + 2\alpha)\pi_6 = \beta\pi_2 + \beta\pi_4 + \alpha\pi_7 + \alpha\pi_{14} \\
(\beta + 3\alpha)\pi_7 = \beta\pi_3 + \beta\pi_5 + \beta\pi_6 + \alpha\pi_{15} \\
(3\beta + \alpha)\pi_8 = \beta\pi_0 + \alpha\pi_9 + \alpha\pi_{10} + \alpha\pi_{12} \\
(2\beta + 2\alpha)\pi_9 = \beta\pi_1 + \beta\pi_8 + \alpha\pi_{11} + \alpha\pi_{13} \\
(2\beta + 2\alpha)\pi_{10} = \beta\pi_2 + \beta\pi_8 + \alpha\pi_{11} + \alpha\pi_{14} \\
(\beta + 3\alpha)\pi_{11} = \beta\pi_3 + \beta\pi_9 + \beta\pi_{10} + \alpha\pi_{15} \\
(2\beta + 2\alpha)\pi_{12} = \beta\pi_4 + \beta\pi_8 + \alpha\pi_{13} + \alpha\pi_{14} \\
(\beta + 3\alpha)\pi_{13} = \beta\pi_5 + \beta\pi_9 + \beta\pi_{12} + \alpha\pi_{15} \\
(\beta + 3\alpha)\pi_{14} = \beta\pi_6 + \beta\pi_{10} + \beta\pi_{12} + \alpha\pi_{15} \\
(4\alpha)\pi_{15} = \beta\pi_7 + \beta\pi_{11} + \beta\pi_{13} + \beta\pi_{14} \\
\end{array}
\]

(A.1)
Acronyms

ALA Average Link Availability 5, 21

DTN Delay and Disruption Tolerant Network iii–v, ix, 1–13, 15, 17, 19–21, 23, 25, 30, 59–61, 65, 66, 76

E2E end-to-end vii, 1–4, 6, 7, 10, 12, 13, 18, 19, 59, 61, 64, 66, 67

FIFO Frist In First Out 10, 16

FTP File Transfer Protocol 1

HTTP Hypertext Transfer Protocol 1

IP Internet Protocol 1

LAN Local Area Network 1

LRAW Localized Random Walk 10

MANET Mobile Ad Hoc Network iii, 1–3, 6, 60

MTTF Mean Time To Failure 21

MTTR Mean Time To Recovery 21

NASA National Aeronautics and Space Administration 5

PDTN Predictable Delay Tolerant Network 21
PQ  Priority Queue 10

SFTP  Secure File Transfer Protocol 1

SMTP  Simple Mail Transfer Protocol 1

TCP  Transport Control Protocol 1