Imperial College London
Department of Electrical and Electronic Engineering

Performance Modelling and Optimisation of Multi-hop Networks

Omer Hassan Omer Abdelrahman

January 2012

Submitted in part fulfilment of the requirements for the degree of
Doctor of Philosophy of Imperial College London

Abstract

A major challenge in the design of large-scale networks is to predict and optimise the total time and energy consumption required to deliver a packet from a source node to a destination node. Examples of such complex networks include wireless ad hoc and sensor networks which need to deal with the effects of node mobility, routing inaccuracies, higher packet loss rates, limited or time-varying effective bandwidth, energy constraints, and the computational limitations of the nodes. They also include more reliable communication environments, such as wired networks, that are susceptible to random failures, security threats and malicious behaviours which compromise their quality of service (QoS) guarantees. In such networks, packets traverse a number of hops that cannot be determined in advance and encounter non-homogeneous network conditions that have been largely ignored in the literature. This thesis examines analytical properties of packet travel in large networks and investigates the implications of some packet coding techniques on both QoS and resource utilisation.

Specifically, we use a mixed jump and diffusion model to represent packet traversal through large networks. The model accounts for network non-homogeneity regarding routing and the loss rate that a packet experiences as it passes successive segments of a source to destination route. A mixed analytical-numerical method is developed to compute the average packet travel time and the energy it consumes. The model is able to capture the effects of increased loss rate in areas remote from the source and destination, variable rate of advancement towards destination over the route, as well as of defending against malicious packets within a certain distance from the destination. We then consider sending multiple coded packets that follow independent paths to the destination node so as to mitigate the effects of losses and routing inaccuracies. We study a homogeneous medium and obtain the time-dependent properties of the packet's travel process, allowing us to compare the merits and limitations of coding, both in terms of delivery times and energy efficiency. Finally, we propose models that can assist in the analysis and optimisation of the performance of inter-flow network coding (NC). We analyse two queueing models for a router that carries out NC, in addition to its standard packet routing function. The approach is extended to the study of multiple hops, which leads to an optimisation problem that characterises the optimal time that packets should be held back in a router, waiting for coding opportunities to arise, so that the total packet end-to-end delay is minimised.

Acknowledgements

I would like to express my sincere appreciation to my supervisor, Professor Erol Gelenbe, for his invaluable academic and financial support without which this work would not have been possible. I also wish to thank my examiners, Professor Jon Crowcroft and Dr William Knottenbelt, for their thorough reading and helpful comments. I am also grateful to my colleagues at the Intelligent Systems and Networks Group, who provided a healthy working environment. Most of all, I want to thank my family for their encouragement and support in all my endeavours; I owe all my success to them.

Contents

List of Figures

List of Abbreviations

ARQ	Automatic repeat request
cdf	Cumulative distribution function
DPI	Deep packet inspection
DTMC	Discrete-time Markov chain
DTN	Delay-tolerant network
FCFS	First-come, first-served
FJQN	Fork-join queueing network
HOL	Head-of-line
iid	Independent and identically distributed
LT	Laplace transform
NC	Network coding
ONC	Opportunistic network coding
pdf	Probability density function
QNA	Queueing network analyser
QoS	Quality of service
RLNC	Random linear network coding
RR	Round-robin
SNC	Synchronous network coding
TTL	Time to live

List of Symbols

a. The diffusion model

$\bar{X}(s)$	LT of a function $X(t)$
$\tilde{X}(\xi, s)$	double LT of a function $X(x,t)$
X_0	X in the absence of packet losses and a time-out mechanism

Non-homogeneous media

r	time-out parameter
μ	retransmission delay parameter
D	distance between source and destination
$Y(t)$	distance of a packet to its destination at time $t \geq 0$
$b(z)$	mean of instantaneous rate of change of $Y(t)$ at distance z; $b(z) = b_k$ for segment k
$c(z)$	variance of instantaneous rate of change of $Y(t)$ at distance z; $c(z) = c_k$ for segment k
$\lambda(z)$	average loss rate at distance z; $\lambda(z) = \lambda_k$ for segment k
$f(z,t)$	pdf of $Y(t)$; $f(z,t) = f_k(z,t)$ for segment k
$I(z,t)$	rate of flow of probability, in the positive direction, across a point z at time t
$P(t)$	probability a packet has reached its destination at time t
$W(t)$	probability a packet's time-out has expired
$L(t)$	probability a packet is lost
T	total travel time of a packet
J	total energy consumption of a packet
Z_k	boundary between the k-th and $(k+1)$-th segments
S_k	size of the k-th segment
n	index of the discretisation segment that includes the source node
m	total number of segments

Time-dependent analysis

$T_{k,N}$	total delivery delay of any k-out-of-N packets
$g(t)$	pdf of T
$G(t)$	cdf of T
$g_{k,N}(t)$	pdf of $T_{k,N}$
$J(t)$	energy expended by a packet up to time t
$J_{k,N}$	total energy expended to deliver any k-out-of-N packets
$h(x)$	pdf of J
$h(x,t)$	pdf of $J(t)$
$h_d(x,t)$	pdf of energy expended by a packet which has arrived to destination on or before time t
$h_s(x,t)$	pdf of energy expended by a packet which is travelling at time t
$h_\iota(x,t)$	pdf of energy expended by a packet whose travel has been interrupted on or before time t
$h_{k,N}(x)$	pdf of $J_{k,N}$
$\phi(x,t)$	joint pdf of total energy consumption and travel time of a packet
$\psi(t)$	pdf of time interval between loss or time-out of a packet and retransmission of a new one
$\gamma_d(t)$	pdf of the duration of a successful search attempt
$\gamma_\iota(t)$	pdf of the duration of an interrupted search attempt due to either loss or time-out

b. Network coding

$\hat{X}$	forward recurrence time of a random variable X
$X(t)$	cdf of a random variable X
$Z(i)$	a subset that does not include i
A_i	packet inter-arrival time to queue i of encoding node
λ_i	average arrival rate of flow i
L	packet size
S	service time
φ	output packet rate from encoding node
φ_c	output rate of coded packets
η	coding gain

Opportunistic coding

$a_{i,n}$	arrival instant of the n-th packet of flow i to encoding node
$d_{i,n}$	departure instant of the n-th packet of flow i from encoding node
$S_{i,n}$	service demand of the n-th packet of flow i
$V_{i,n}$	vacation time experienced by the n-th packet of flow i
S_i	equivalent service time of queue i
V_i	equivalent vacation time of queue i
R_i	response time for flow i
q_i	probability that flow i does not participate in a coding operation
ζ	coding opportunities

Timer-based coding

Λ	total packet arrival rate to encoding node
ρ_i	probability a packet is of class i
r_i	time-out parameter for flow i
U_i	service time of coding queue i
μ_i	service rate of coding queue i
$\mathcal{Q}_i(t)$	number of packets waiting in coding queue i
p_n	stationary probability of n aggregate packets in coding stage
π_n	stationary probability of n aggregate packets waiting in coding stage at the end of a service time
T_i	synchronisation delay at coding queue i
D	aggregate delay cost function for all packet classes
$\mathcal{L}$	maximum of two packet lengths each of size L
FB	size of feedback packets
ε_c	energy cost of XOR-ing two bits
ε_{tr}	energy cost of transmitting a single bit
$\mathcal{E}_p$	energy cost of processing header of a packet
$\mathcal{P}_f$	power consumption of forwarding
$\mathcal{P}_c$	power consumption of coding
$c(u,v)$	capacity of edge (u,v)
$\lambda(u,v)$	average packet rate on edge (u,v)
$\mu(u,v)$	average service rate of edge (u,v)
$R(u,v)$	response time of edge (u,v)

1. Introduction

As computer and communication networks become more complex and increasingly interconnected, there is a growing demand for techniques that can assist in evaluating and optimising their performance. The aim of this thesis is to further our understanding of the interactions and trade-offs that govern the behaviour of large packet networks. To this end, we study whether a packet will ultimately succeed in reaching a given destination, how long this will take, and how much energy may be expended. We also investigate the implications of some packet coding techniques on both quality of service (QoS) and resource utilisation. Probability models of computer algorithms and networked systems have long been applied fruitfully, and this thesis takes a similar approach. The research hypothesis is that abstract mathematical models can help to optimise the performance of multi-hop networks, both at the macro and micro levels.

1.1. Research questions and motivations

In this thesis we examine three issues in multi-hop networks which have not been sufficiently addressed in previous work:

1) Modelling packet travel in large networks: Despite the large body of literature on characterising the performance of large-scale networks, most existing work has focused on spatially homogeneous environments. In many practical settings, however, routing accuracy and packet loss rate vary over the distance from a source to a destination. For example, some parts of a wireless network may be particularly faulty or degraded while the rest of the network is operating properly. Thus the network's operational quality may be quite good close to the source node, but it may become less reliable when the packet moves far away from it. Another example of a non-homogeneous medium occurs when a packet progresses more rapidly as it approaches its destination node, for instance because a directional routing being used may become more accurate. The converse is also possible if the packet is designed to carry some form of attack, such as a virus or a worm, on the destination node which is being protected from such packets by the intermediate nodes [1], so that as the packet approaches the destination node it is more likely to be dropped. Consequently, there is a need for analytical models to explicitly describe the travel process of a packet over networks that have such non-homogeneous properties. It would be particularly useful to have a model which is generic enough to be independent of network specifics such as topology, routing policy and physical medium, and to be able to represent different applications and scenarios. Many optimisation problems can then be formulated in such a generic framework.

2) Evaluating the impact of coding at source nodes: Wireless ad hoc and sensor networks, which are usually deployed in large numbers, need to deal with the effects of node mobility, routing inaccuracies, higher packet loss rates, limited or time-varying effective bandwidth, energy constraints, and the computational limitations of the nodes [2,3]. Such characteristics severely limit the ability of a network to guarantee a certain level of performance. Yet in some applications of sensor networks, such as forest fire detection and seismic activity monitoring, packets reporting irregularities in measurements have more stringent delay constraints that require the network to provide more reliable delivery. In such instances, it may be necessary for the source node to forward either duplicate or coded packets that follow independent paths to the destination node so as to improve reliable delivery and reduce effective travel times. This may, however, come at the price of higher energy consumption, which is also an important issue in wireless networks. Specifically, since energy utilisation per packet is proportional to the time spent travelling in the network, there is a trade-off between having a small number of packets which travel for a long time in the network and a large number of packets which may spend a shorter time. Hence the merits and limitations of sending duplicate or coded packets into large networks, both in terms of reliable delivery times and energy efficiency, need to be evaluated.

3) Evaluating the impact of inter-flow network coding (NC): With the advent of NC [4], it has been shown that network performance can be further improved by allowing not only source nodes but also intermediate nodes to combine received packets, for example by a bit-by-bit XOR operation of two packets, before forwarding them towards their destinations. This can provide path diversity for the information in such a way that an efficient communication paradigm can be deployed in resource-constrained environments [5]. However, the impact of inter-flow NC on packet delay is not fully understood. Indeed, although a lower traffic rate per link may reduce the link delay, and thus the overall delay that a given packet travelling through a network will experience, coding can also increase delay in several ways. The need for combining packets at nodes may force packets to wait for the arrival of other packets with which they will be combined, introducing a synchronisation delay. Furthermore, although individual link delays will be reduced, node delays may be affected adversely because in order to reconstitute the packet streams at output nodes, overall the network will have to carry the same amount of traffic if no information is to be lost. Finally, the need to decode packets at output nodes implies further synchronisation delays due to waiting for the "right" combination of packets to arrive before a given packet can be decoded and forwarded to the final receiver. Thus the resulting trade-offs in NC between throughput, delay and resource utilisation need to be investigated carefully, and effective mathematical tools are required so that correct design decisions can be made.

1.2. Methodology

In this thesis we apply two different mathematical techniques to achieve the research objectives. The first is based on Brownian motion [6–9] and allows us to analyse the travel process of a packet in large non-homogeneous environments and to evaluate the effect of sending redundant packets into a network. This *macroscopic* approach is motivated by the fact that it is difficult to obtain closed-form expressions for the performance of large-scale networks using microscopic techniques, such as queueing models, which are usually computationally expensive and may not always be scalable [10, 11]. The second approach we adopt is based on the analysis of queueing systems with specific service processes in order to capture the effect of inter-flow coding at the interior nodes in the network. Although the primary focus of our research is on large-scale networks, the choice of this *microscopic* method is justified by the facts that (a) inter-flow NC opportunities usually arise in certain basic structures which may nonetheless represent small fragments of larger arbitrary topologies, and (b) the level of detail provided by macroscopic techniques is not sufficient to capture the gains and limitations of NC.

1.3. Thesis contributions

In this thesis we consider a probability model for travel of a packet from a source node to a destination node in a large non-homogeneous multiple hop network with unreliable routing tables and packet losses. The randomness models the lack of precise routing information at each of the network hops [7, 12], and randomness in routing can also be used to model networks where one wishes to explore alternate paths [13] in order to discover the more reliable paths, or those that may have other desirable characteristics such as lower delay or lower packet loss. The packet's travel may also be impeded if certain routers on its path prove to be unreliable, or the packet may be dropped from a buffer or destroyed due to packet loss. The packet can also have a limited time-out that allows the source to retransmit a dropped or lost packet. Because the network itself may be extremely large, we consider packet travel in an infinitely large random non-homogeneous medium, with events that may interrupt, destroy or stop the packet from moving towards its destination. Generalising the work in [7,8] which focused on homogeneous environments, we obtain an exact expression for the average time and energy that it takes the packet to eventually find the destination node, based on a modified and non-homogeneous Brownian motion model.

While the expected performance can be useful in many cases of interest, it is not sufficient to provide worst-case guarantees that can only be inferred from the distribution. The latter, however, requires analysing the time-dependent behaviour of the packet travel process which is difficult to obtain but useful to know in order to evaluate the effect of uncertainties in the network, such as packet losses, inaccuracies or errors in routing, and possible energy limitations. It is also valuable if one wishes to evaluate different means for improving performance at the price of higher energy costs by sending out duplicate or coded packets. Thus we consider N multiple coupled Brownian motions to represent the travel of each packet and derive the distribution of total forwarding delay and energy consumption when any k-out-of-N packets can decode the entire group, where $k \geq 1$ and $N \geq k$. In order to make the analysis more tractable, we focus on homogeneous environments and assume that each of the transmitted packets travels independently of the others which is a reasonable assumption for very large networks [14]. Hence our work extends [9] where the average travel time and energy consumption of 1-out-of-N transmissions (i.e., duplication) are obtained.

Finally, we propose models that can assist in the analysis and optimisation of the performance of inter-flow NC. We first focus on modelling a single NC router in isolation, then we extend the analysis to multi-hop settings. We formulate and solve approximately an optimisation problem which provides the optimal time that an encoding node should wait before sending the information that it has un-coded, so that the average packet end-to-end delay including encoding, queueing and decoding at the output is minimised. We investigate the trade-offs in NC between delay and bandwidth or energy usage, and we evaluate the performance of the proposed coding schemes in comparison with a conventional store and forward network.

1.3.1. Summary of contributions

The contributions of this thesis can be divided into two categories: (a) theoretical developments for the diffusion model in [7–9], and (b) a novel queueing theoretic framework for NC of multiple stochastic flows.

a. Performance analysis of large-scale networks

i. For non-homogeneous networks, we have developed a numerical-analytical solution technique based on a finite but unbounded number of internally homogeneous segments, yielding the average packet travel time and the energy expended. We have illustrated the utility of the modelling approach with some applications to wireless networks and network security.

ii. For homogeneous networks, we have obtained the time-dependent solution of the density function of the distance of a packet to its destination, and derived the distribution of the total forwarding delay and energy consumption for k-out-of-N coding techniques.

b. Queueing performance of inter-flow NC

i. We have proposed and analysed approximately queueing models for two possible designs of a NC router: a single server queue for joint opportunistic coding and transmission, and a multistage model for a NC implementation in which packet coding and transmission are performed independently and a time-out is used to modify coding opportunities.

ii. We have incorporated the second model into a network setting and proposed a simple heuristic, based on fork-join synchronisation primitives, for choosing the time-out periods so that the average end-to-end packet delay (between encoding and decoding nodes) is minimised.

iii. We have validated the analytical solutions via discrete event simulations with ns-2 [15].

1.4. Thesis outline

The remainder of this thesis is organised as follows. In Chapter 2 we provide background information and review previous work related to our own in order to place our contributions in the proper context. The chapter is structured around the two main issues addressed by our research on performance analysis and optimisation of packet networks: modelling large networks, and evaluating the impact of packet coding techniques. In Chapter 3 we present our diffusion model for non-homogeneous networks, derive exact expressions for the average travel time and energy consumption, and illustrate the results with several examples. Chapter 4 is concerned with the time-dependent analysis of the packet's travel process in homogeneous environments, including the distribution of total delivery delay and energy expenditure of coding and replication at the source nodes. Chapter 5 presents queueing models and a heuristic for optimising the performance of inter-flow NC. Finally, the thesis is concluded in Chapter 6 by a summary of results and recommendations for future work.

1.5. Publications

This thesis is based in part on the following publications:

- O.H. Abdelrahman, E. Gelenbe, “Packet delay and energy consumption in non-homogeneous networks,” *The Computer Journal*, Special Issue on Security and Performance of Networks and Clouds, November 2011 [submitted].

- O.H. Abdelrahman, E. Gelenbe, “Search in non-homogenous random environments,” in *13th ACM SIGMETRICS Workshop on Mathematical Performance Modeling and Analysis (MAMA ’11)*, San Jose, CA, USA, June 2011, pp. 17-19. [extended abstract; revised version to appear in a special issue of *Performance Evaluation Review*].

- O.H. Abdelrahman, E. Gelenbe, “Packets travelling in non-homogeneous networks,” in *14th ACM International Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM ’11)*, Miami, FL, USA, October 2011.

- O.H. Abdelrahman, E. Gelenbe, “Performance trade-offs in a network coding router,” in *19th IEEE International Conference on Computer Communications and Networks (ICCCN ’10)*, Zurich, Switzerland, August 2010.

- O.H. Abdelrahman, E. Gelenbe, “Approximate analysis of a round robin scheduling scheme for network coding,” in *6th European Performance Engineering Workshop (EPEW ’09)*, LNCS, vol. 5652. London, UK: Springer, July 2009, pp. 212-217.

- O.H. Abdelrahman, E. Gelenbe, “Queueing performance under network coding,” in *IEEE Information Theory Workshop on Networking and Information Theory (ITW ’09)*, Volos, Greece, June 2009, pp. 135-139.

2. Background and related work

This chapter provides background information and a survey of previous work related to our own. The chapter is organised around the two main issues addressed by our research on performance analysis of packet networks: modelling large networks and evaluating the impact of packet coding.

In Section 2.1 we review existing analytical frameworks for characterising packet delivery time and energy consumption in large wireless ad hoc and sensor networks. The discussion covers queueing-theoretic and random walk techniques in addition to Brownian motion (diffusion) approximations. For each analytical approach, we highlight the main advantages, known limitations and possible application domains. Special emphasis is given to diffusion based methods which can simplify the analysis considerably due to their asymptotic nature. In Section 2.2 we discuss two well-known packet coding techniques that have been suggested in the literature to improve the reliability and delivery performance of packet networks. In the first scheme, known as erasure coding, the encoding and decoding operations are restricted to source and destination nodes, respectively, while other nodes in the network only store and forward packets. The second scheme is network coding (NC) which allows not only source nodes but also intermediate nodes to combine received packets instead of simply relaying them. We discuss the two main approaches to NC, namely intra-flow NC which restricts coding to packets of the same flow, and inter-flow coding in which packets from distinct flows can be combined together when they pass a common node. In addition, we survey the different queueing models that have been formulated for evaluating the performance of NC. Each of the two main sections in this chapter is concluded by summarising how our research builds on previous work.

2.1. Analytical frameworks for performance evaluation of large-scale networks

2.1.1. Queueing theory

Delay in store and forward packet networks is traditionally described as consisting of four components [16]: processing, queueing, transmission and propagation. Propagation delay depends on the physical characteristics of a link and is independent of the traffic carried by the link thus can be neglected. Processing delay is often ignored in the literature since the early days of modelling the ARPAnet [17]; the assumption is that computational power is not a limiting resource and as a result processing delay is considered to be independent of the amount of traffic handled by the node. Consequently, data networks are usually modelled as networks of transmission queues.

In recent work, queueing theory has been suggested as a means to evaluate packet delivery time and energy utilisation in multi-hop wireless ad hoc and sensor networks. The approach generally consists in (a) constructing a detailed model for an individual node which accounts for traffic generated by the node and interaction with neighbouring sensors including channel contention and reception of traffic; and (b) applying a decoupling approximation, whereby each node is assumed to be independent of the others, in order to analyse the entire network.

The work in [10] proposes a queueing model for delay analysis of random access wireless ad hoc networks with static nodes that are distributed uniformly and independently over a torus of unit area. In this network, the mean and the variance of the service time at each node are computed based on a simple protocol interference model [18] while routing probabilities are obtained in terms of the communication area of a node. A diffusion approximation is then applied to estimate the average node's delay assuming that each node generates constant size packets according to a Poisson process. The authors also derive an upper bound on the maximum achievable per-node throughput and show that it is of the same order as that of [18] obtained using an information theoretic approach.

A queueing network model for sensor networks with geographic random forwarding [19] is presented in [20] and [21] to evaluate the distributions of end-to-end delay and energy consumption, respectively. Specifically, the authors model each sensor as a discrete-time finite size queue with geometric inter-arrival time and Phase-type distributed service time, and they propose a decoupling approximation to analyse a network of these sensors. Experimental results, conducted on a small network testbed with different configurations, are also presented in [20, 21] to validate the analytical approach.

In [11], the performance of a sensor network whose nodes may enter a sleep mode is studied assuming that nodes are stationary and uniformly distributed over a disc of unit radius, and that all traffic is routed to a centrally located sink. In this approach, each node is represented by a discrete-time Markov chain (DTMC) which incorporates input parameters representing the node's sleep/active dynamics and packet generation rate, as well as estimated parameters describing traffic routing and channel contention. The latter are obtained using a fixed point approximation procedure which is computationally expensive and does not allow for large-scale networks to be represented. Nevertheless, the technique provides interesting results illustrating the trade-off between delivery delay and energy expenditure as the sensor dynamics in sleep/active mode vary; furthermore, the accuracy of the numerical technique is validated through comparison with simulations.

In order to deal with the aforementioned scalability issue in [11], the approach is modified in [22] using a fluid representation of all quantities that depend on the specific location within the network topology, including sensor, routing and traffic densities. Although the approach captures non-homogeneity in sensor deployment, it assumes a particular routing policy and channel contention model. Moreover, some of the results are numerical and not in closed-form.

The impact of node mobility on the performance of wireless sensor networks is studied in [23], where three types of queueing networks with immobile $GI/G/1$ queues are used to capture mobility: gated nodes which are probabilistically disconnected from the entire network; intermittent links that fail probabilistically; and intermittent servers which experience a vacation effect that allows them to receive but not transmit packets. Different performance measures of interest are obtained using the Queueing Network Analyser (QNA) tool [24]. A similar approach is followed in [25] but, differently from [23], the end-to-end delay is obtained using a Jackson network [26, 27] approximation.

Queueing theoretic techniques provide a microscopic view of network dynamics which requires intensive computations and may not always be scalable. Conversely, random walk and Brownian motion based approaches focus on characterising the route followed by a packet from its source to destination, without regard to queueing delays at intermediate nodes. This macroscopic view often leads to closed-form results which can be used in conjunction with approximate queueing models [28] in order to derive more accurate end-to-end performance measures. The relation between hop counts and network performance has been studied before (e.g., for multi-hop wireless networks [29]), and it is outside the scope of the present study.

2.1.2. Random walk

A random walk is a stochastic process describing the motion of a particle that takes random jumps at either discrete or continuous instants in time. The analysis of packet travel time in random multi-hop networks is related to first passage (hitting) times of random walks. In this area there are many results for special cases of network topologies.

In [30], the mean and variance of the hitting time are obtained for a torus-lattice network graph when the next node visited is selected at random among all neighbours leading to an unbiased walk. The analysis indicates that in such networks the probability distribution of packet delivery time is approximately geometrically distributed. In [31], the probability of an unbiased traveller visiting a particular node in a given step is derived for a 2D grid-based sensor network.

Random walks on random geometric graphs [32] are used in [33, 34] to model uniform wireless networks in which nodes with fixed transmission range are deployed uniformly at random over a given area. The results indicate that unbiased routing achieves poor performance because the walker may "orbit" around the target node for a long time before attaining it.

The first passage time for multiple independent and unbiased random walks on a connected network is considered in [35], and it is shown that the mean first passage time converges to the shortest path between the source and the destination as the number of walkers approaches infinity.

The average time to locate a node at the origin of a chain by multiple searchers switching probabilistically between random walks and long jumps is analysed in [36] showing how this alternate motions can be conducted to optimise the search. In the context of wireless environments [37], a random walk with jumps is used to represent a network in which an intermediate node may decide to increase its transmission power to reach a neighbour beyond its nominal transmission range in order to explore different regions of the network. The impact of the jump probability on the hitting time and energy consumption for a single packet search over a line, square grid and random geometric graph has been evaluated in [37].

There is a large body of literature on the passage time of random walks on *random graphs* (e.g., [38–41]); most of these studies, however, utilise tools from graph theory that are outside the scope of the present research. In general, random walk analysis is difficult when one studies non-homogeneous networks [42, 43] or if the distribution, rather than the expected value, of the hitting time is desired.

2.1.3. Brownian motion

Brownian motion is a scaling limit of the random walk obtained by letting the step size approach zero. The time evolution of the probability density function (pdf) of the position of a particle undergoing Brownian motion is described by a second-order partial differential equation known as the *diffusion equation*. It has been used traditionally to represent packet flow in communications systems and traffic flow in transportation systems [44–46] and more recently to model packet traversal through large-scale networks [7–9, 47, 48]. In the remainder of this section we survey the literature on using Brownian motion to approximate search problems in random environments, including the search by a packet for a destination node in a large multi-hop network. An outline of the derivation of the diffusion equation is provided in Appendix A.

The approach adopted in this research is based on the work of Gelenbe [7, 8] where the average travel time to a destination node in an infinitely large homogeneous multi-hop wireless network is analysed using a *mixed* discrete and Brownian motion model. It is shown that the travel time is finite on average even with inaccurate routing information and packet losses, provided that a time-out mechanism is inserted to destroy the ongoing packet after a predetermined time, and replace it with a new packet that starts at the same source and proceeds at random and independently of its predecessor. Since the network is infinite, the time-out also protects the packet from spending an unreasonably long time in remote areas from which it may never return. By using a randomly different travel path, the new packet takes a distinct path from its previous incarnation, increasing its chances of reaching the destination.

The diffusion model [7, 8] is generalised in [9] to multiple packets which are simultaneously but independently sent out in the quest for the same destination node. The analysis is based on the use of multiple Brownian motions coupled by the total rate of attraction exerted on each diffusion process by all other diffusions due to the fact that one of them may have reached the destination node. Closed-form expressions for the average travel time and energy consumption are obtained in terms of the distance between source and destination, the number of transmitted packets, the average time-out, the routing uncertainty, and the loss rate of packets.

The approach followed in [7–9] is based on transforming the transient process of travelling from the source to the destination just once to an ergodic process in which the packet goes from the source to the destination, stops there for a short time which is an iid positive random variable, and the travel restarts and is repeated indefinitely. This transformation facilitates the computation of the mean travel time and energy consumption from the steady-state solution of the synthetic ergodic process, but it does not allow for the distributions to be evaluated.

The time-dependent solution for the passage time using Brownian motion is considered in [48] where, differently from Gelenbe's model, losses and retransmissions are assumed to occur at specific distances from the destination that represent specific intermediate nodes. Thus Brownian motion is used as a model of packet propagation from one hop to another, whereas in [8, 9] it is used to represent the route followed by a packet from source to destination. By assuming absorbing barriers with jumps at intermediate nodes and applying previous results [49] for the distribution of hitting time of pure Brownian motion, the authors in [48] derive the Laplace transform (LT) of the distribution of the total delivery delay for finite but non-homogeneous node population.

Different Brownian motion based search strategies over a sensor network are considered in [47], and it is shown that a source and sink driven "sticky search", where both the source and destination send probes into the network that leave trails, can match the delivery success probability of a spatial periodic caching scheme without requiring much memory or infrastructure support. The analysis in [47] utilises results on intersection exponents for Brownian motion [50–52] where the probability of two groups of Brownian particles never intersecting over a time interval has been obtained. Scaling properties of hitting times for a circular target area located at the centre of a larger circle with reflecting boundaries are derived in [53]. The analysis suggests that when the source node is close to the target area, a Brownian motion scheme first analysed in [54] achieves mean hitting time on the same order as a random direction forwarding approach; otherwise, the latter achieves better performance.

The first passage time problem has been studied in physics, biology and ecology and some of the work done in these areas can be applied to communication networks. In [55] bounds and approximations are derived for the average travel time of a single searcher that alternates between local diffusive search and fast directed relocation (known as flights) in order to find any of Poisson distributed targets. It is assumed that the target can only be located during the diffusion phase, and that the searcher moves with a constant velocity during the ballistic phase. The authors show that the travel time can be minimised by appropriately choosing the waiting times in the slow and fast regimes. In [56], it is shown that an inverse square power-law distribution of flight lengths (i.e., Lévy flights) is optimal for searching sparsely and randomly distributed revisitable targets, and that Brownian movement is sufficiently efficient for locating abundant targets. Such intermittent search mechanisms, which avoid oversampling [57] in the sense that already visited sites are not revisited continuously, have been observed in animals' hunting patterns [58, 59] thus confirming the *Lévy flight foraging hypothesis* [60]. In other contexts, such as navigation in small-world networks [61, 62] and transport systems [63], studies have shown that efficient routing can be performed by links having a few long-range connections following Lévy distribution in addition to regular short-range connections.

Another related model [64] in biological physics investigates the first passage time distribution of a diffusing particle which may overshoot its destination, i.e. it diffuses away before being absorbed. The analysis suggests that there are two basic regimes: diffusion dominated and absorption dominated. In the former, most of the travel time is spent delivering the particle by diffusion to the destination, whereas in the latter the travel time is spent mostly wandering around the target and waiting for successful absorption. This model could be applicable to a wireless sensor network in which nodes are periodically put into a sleep mode, to reduce energy consumption, which causes the unavailability of the nodes and, in turn, the possibility of a packet overshooting its destination. This could capture, for instance, the case where a route discovery fails because the destination node is asleep.

2.1.4. Conclusions

We have reviewed previous research efforts on analysing packet delivery time and energy consumption in large wireless ad hoc and sensor networks. The discussion covered queueing, random walk and Brownian motion models. Although queueing theoretic approaches can accurately capture network dynamics, they are computationally expensive and may not always be scalable; furthermore most of their results are numerical and not in closed-form. On the other hand, random walk and Brownian motion provide a high-level abstraction of packet forwarding without regard to queueing at intermediate nodes, i.e., they yield result for the total number of hops travelled from source to destination. This macroscopic approach can yield simple analytical results that can be used in conjunction with approximate queueing models in order to obtain more accurate end-to-end performance measures. Random walk and Brownian motion have been applied in the literature to describe the behaviour of biological, chemical, transport and social networks, and we have highlighted the analogies between some of the work done in these areas and the problem under consideration. While random walk techniques need to make assumptions about the network structure and routing policy, Brownian motion methods represent these two aspects by a continuous diffusion process, characterised by a drift and a variance parameters, which simplifies the analysis considerably. Most previous work has focused on spatially homogeneous environments that may not represent many applications. Furthermore, the energy aspects of sending redundant packets in large networks have not been addressed in the literature. Thus our research will focus on these two issues, namely spatial non-homogeneity and packet redundancy, using a Brownian motion approach.

2.2. Coding techniques for packet networks

2.2.1. Erasure coding

In an error prone communication environment, it may be necessary for the source node to send redundant packets that follow different paths to the destination node in order to mitigate the effects of packet losses and uncertainties in routing information. This can be implemented through erasure coding whereby the source organises the data it transmits into successive blocks of k packets and encodes them into $N \geq k$ packets such that the original block can be reconstructed at the receiver from a k-subset of the N packets, which is possible with many existing coding algorithms. In the special case where $k = 1$, the source sends multiple identical copies of a single packet so that the packet is considered to be successfully delivered when at least one of the copies reaches the destination node. In [14], erasure coding has been shown to reduce delay variations in comparison to replication in delay-tolerant networks (DTNs) under the assumption that the travel time of each individual packet is an independent and identically distributed (iid) random variable with exponential or Pareto distribution. However, the impact of these packet redundancy schemes on energy consumption has not been considered in [14] and, furthermore, it is not clear if these results hold under more realistic assumptions.

2.2.2. Network coding

NC was first introduced in [4] showing that the multicast capacity of networks can be achieved by allowing intermediate nodes to combine received packets instead of simply relaying them. In contrast to traditional store and forward networks which aim at avoiding collision of traffic streams as much as possible, NC encourages mixing of information at intermediate nodes which has the advantages of increasing network throughput, saving bandwidth and providing load balancing and security to the network [5]. In particular, NC can reduce the maximum bandwidth requirements of certain links provided that redundant data is sent over alternate paths so that destination nodes may then reconstruct the original packet flows. Thus NC can reduce the peak traffic rates, but it will require traffic to be distributed on a larger number of paths [65]. Furthermore, with its ability to disguise the content of packets, NC provides a new form of system security beyond encryption by rendering the traffic streams traversing networks much more difficult to decipher, for instance if none of the flows taken singly can be decoded by itself [66–69]. NC can be based on encoding packets from the same flow (intra-flow coding), or from distinct flows (inter-flows coding).

Intra-flow network coding

Intra-flow NC generalises erasure coding by allowing not only source nodes but also intermediate nodes to combine packets within a single flow. Consequently, each packet can contain some information about other packets in the flow and a node need not keep track of packets which may have been missed by the next hop or the destination. The advantages of this method are therefore enhancing reliability by not relying on the reception of any particular packet and enabling multiple destinations of a single flow to share network resources.

Most existing work on intra-flow NC assumes a network model in which all participating nodes send out random linear combinations [70] of all previously received packets, which is known as random linear network coding (RLNC). An algebraic formulation of RLNC in the context of constant data rate sources and deterministic links capacities is introduced in [71], a fluid-flow analysis of the propagation of packets carrying innovative information in lossy networks with stochastic packet arrivals is presented in [72], and a distributed implementation is proposed in [73].

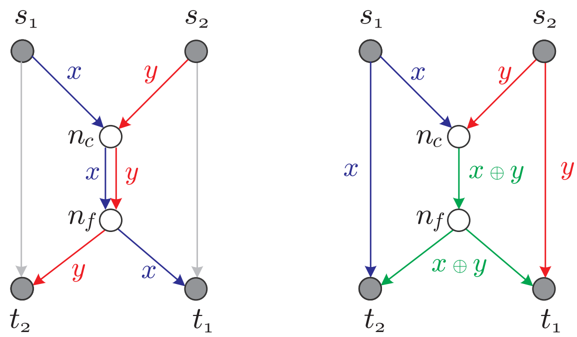
There is a large body of literature on the queueing performance of RLNC under different network models. A bulk-service queueing model for a single source-destination pair with lossy channel is presented in [74], and numerical results indicate that using a fixed coding block size increases delay in comparison to traditional retransmission schemes. In [75], the impact of limited buffer capacity on delay and loss performance of RLNC over a single-hop and a two-link tandem network is analysed. Simulation results are also provided to evaluate the effect of the coding field size showing that a relatively small one achieves comparable performance to an infinitely large field. Such a coding scheme, which does not require feedback, can outperform automatic repeat request (ARQ) when feedback is unreliable or too slow. A queue backlog analysis of single-hop multicast with RLNC is presented in [76] and it is shown that coding is order-optimal with respect to the number of receivers. An online coding and queue management algorithm based on acknowledging degrees of freedom rather than actual packets is proposed in [77, 78], and the analysis of the coding policy shows that the queue size grows more slowly with load than a standard acknowledgement strategy. This acknowledgement scheme has also been analysed in [79] for a wireless channel with delayed feedback.

In [80], the multicast delay and throughput trade-off with intra-flow coding is considered for a slotted-time collision based wireless network, showing that coding improves throughput and energy costs at the expense of higher packet delays as compared to plain routing. However, the results are obtained under the assumptions of one-bit packet lengths and saturated queues at source and relay nodes.

Inter-flow network coding

This type of NC allows packets belonging to distinct flows to be combined together when they pass through a common node, thus enabling higher throughput for the most common scenario where only unicast flows are present in the network. As an example, consider the directed butterfly network [4] depicted in Figure 2.1, which has two independent unicast packet flows x and y between the source-destination pairs (s_i, t_i), $i = 1, 2$. If the capacity of the link $n_c \to n_f$ is $c(n_c, n_f)$ and the traffic rate of x and y are λ_x and λ_y, then the shared link will saturate if $\lambda_x + \lambda_y \geq c(n_c, n_f)$. With inter-flow NC, however, node n_c can transmit the coded flow $x \oplus y$ along the bottleneck link to node n_f as long as $\max(\lambda_x, \lambda_y) < c(n_c, n_f)$, and n_f then forwards it to both destinations. Receivers t_1 and t_2 can then resolve coded information by receiving redundant flows through links $s_2 \to t_1$ and $s_1 \to t_2$, respectively. Obviously, if all links have the same capacity c then the achievable traffic is $\lambda_x + \lambda_y < c$ with routing, while with NC it will be $\lambda_x < c,\ \lambda_y < c$.

Despite its potential throughput gains, performing inter-flow NC is difficult [81], since intermediate nodes need to ensure that incoming flows can be decoded at all destinations



(a) routing (b) NC

Figure 2.1.: A directed network with two independent unicast sessions x and y between the source-destination pairs (s_i, t_i), $i = 1, 2$.

before combining flows together. This has motivated the development of protocols that restricts coding to packets that can be decoded at their respective next hops [82]. From a performance perspective, it is also difficult to analyse inter-flow NC due to the complex queueing behaviour resulting from joint servicing of multiple flows which leads to coupled queueing problems [83]. The impact of inter-flow NC on QoS introduces additional coding delays in nodes, synchronisation delays for jointly coded streams, and decoding delays due to processing and synchronisation at the network output. In the rest of this section, we review previous work on performance analysis of NC of multiple stochastic flows, and for brevity we refer to "inter-flow NC" as simply "NC".

The trade-off in NC between delay and transmission costs (bandwidth and energy) under stochastic packet arrivals has been studied [84–87] for the two-way relay example shown in Figure 2.2. In this network, nodes A and B exchange their packets a and b respectively, through a relay R. With NC, the relay can broadcast a single packet $a \oplus b$, instead of two successive packets, and each receiver decodes by XOR-ing the received packet with its own hence reducing the total number of transmissions from 4 to 3. This scenario has received much attention in the literature because of its simplicity and the fact that packets do not experience decoding delay at the output.

In [84], the energy delay trade-off in the two-way relay network is analysed assuming that the relay accumulates packets from one direction and sends them either after packets from the other direction arrive or the number of packets waiting exceeds the buffer capacity. Packet transmission is then assumed to occur instantaneously (i.e., zero transmission delay). The analysis indicates that in the case of even traffic load, the average delay must tend to infinity in order to achieve minimum energy consumption. A similar queueing analysis for slotted channel is presented in [85]; it is assumed that if the relay has packets from only one source and the buffer capacity is not exceeded, then it transmits

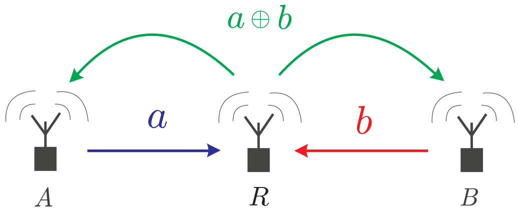


Figure 2.2.: The two-way relay network.

an un-coded packet with some probability. On the other hand, a packet arriving at a full buffer triggers an immediate transmission. Numerical results in [85], depicting the locus of delay and energy when varying the transmission probability of un-coded packets, resemble those reported in [84]. The delay and throughput performance of the two-way relay network under slotted ALOHA is considered in [86], showing that NC does not offer significant gains when the traffic at the relay node is unbalanced. Bounds on energy consumption based on queues with negative customers [88, 89] have been derived in [87]. In this model, coding of two packets is represented by a negative customer which, upon arrival at a non-empty queue, immediately removes another customer from the queue. All of the above results were obtained using simple assumptions such as zero [84,87] or slotted transmission time [85, 86], and they have not been extended to multiple hops.

The stability and energy consumption of NC in a wireless tandem network with slotted transmission are considered in [90], where it is assumed that intermediate nodes can either transmit self-generated packets or encode two relay flows received from neighbouring nodes. The authors consider both synchronous NC (SNC) and opportunistic NC (ONC) which represent two extreme cases with respect to delaying packets for NC. In the former, each node maximises its coding gain by forcing each packet in one flow to be encoded with a packet of *the other flow* that passes through that node with the encoding being carried in sequence for each flow. In the latter approach, a node attempts to minimise its response time by forwarding packets without coding if only one flow is present. The results obtained in [90] show that immediate transmission of first available packets yields higher throughput as compared to waiting for additional packets to arrive before coding, but this gain comes at the expense of reducing energy efficiency.

In SNC, which is the simplest and also most naive approach to NC, we can consider that the encoding process acts as a server which waits for the arrival of one packet from each flow before it can start encoding, and is idle when its input buffer does not contain at least one packet from each flow. A related model is studied in [91] for manufacturing systems, and it is shown that this queueing system is intrinsically unstable so that the queueing delay per flow tends to infinity when the input buffer is unlimited. In other words the waiting time process cannot converge in distribution to a non-defective limit. Most subsequent work which investigated assembly-like processes has avoided the instability problem by assuming limited capacity buffers [92–94] or by controlling the input flow of items [95]. The taxicab problem [96] where taxis and customers can only leave the system together is also closely related to this scheme. In the context of NC, the performance of SNC has been discussed in [90, 97, 98].

The delay performance of a wireless lossy butterfly network which employs opportunistic NC and dynamic buffer allocation at the relay node is evaluated in [99] using a DTMC model which is solved numerically. The authors show that NC provides significant delay gains in moderate to heavy traffic regimes, and they present a joint intra and inter-flow coding policy which further improves performance in low traffic conditions. In [100], the achievable rate regions under QoS constraints are computed for a butterfly network with and without NC. However, the analysis is based on a fluid flow model that does not capture the bursty nature of packet arrivals which is essential for understanding NC gains. End-to-end QoS bounds for both NC and plain forwarding have been derived in [101] using deterministic network calculus, and the results show that coding can improve the worst-case delays even in topologies where no throughput gains are expected. A similar approach for modelling NC using stochastic network calculus is proposed in [102]. Network calculus, however, can only provide bounds that may not be tight in practice.

The performance of NC in single-hop wireless erasure channels has been studied in [103, 104]. In [103], the delay benefits of NC for multiple file downloads from a single transmitter to distinct receivers are considered, and it is shown that coding packets within files improves delay and throughput performance whereas coding across files is not favourable. Throughput analysis of NC for multiple multicast transmissions is presented in [104], and the results indicate that coding across all the sessions can improve throughput under certain conditions which depend on the number of sessions, the number of receivers and the reliability of the channel.

A different line of research on NC of multiple stochastic packet flows develops throughput optimal control policies, based on differential backlog algorithms [105], that can stabilise the network traffic [81, 83]. The analysis of such algorithms, however, focuses on stability properties rather than delay performance and utilises tools from the Lyapunov Stability Theorem that take into account the coupling between queues due to NC.

2.2.3. Conclusions

In the following we summarise the main findings of our literature survey on coding in packet networks.

Coding and replication at source nodes have been shown to improve delivery time in large networks [14], but the cost in terms of energy utilisation has not been addressed before. Furthermore, previous results were obtained under simplified assumptions that do not include the effect of packet losses. The performance of intra-flow NC has been studied extensively in the literature, and it is well understood. It has been shown to be capacity-achieving for single unicast and single multicast connections over wireline and wireless networks [72]. More explicitly, if a network can support a unicast rate between a source and each destination node when no other destinations are sharing the network resources, then NC can support the maximum multicast rate to all destinations. Conversely, it is generally not possible to achieve this maximum rate if only routing is allowed at the interior nodes of the network. Intra-flow NC, however, does not provide additional gain when multiple unicast sessions are present in a network.

For inter-flow NC under stochastic packet arrivals, there are different strategies that can be employed. The simplest and also most naive strategy would require each packet in each flow be encoded with a packet of each of the other flows that pass through that node. This has been shown to incur very large delay and loss penalties particularly when the network is lightly loaded [90, 97, 98]. A more practical approach is to encode packets opportunistically [90] with the option of forwarding un-encoded packets if other packets are not available immediately for encoding. Also, time-outs can be employed by intermediate nodes in order to modify coding opportunities so that a compromise between latency and transmission costs can be achieved.

Despite the significant research efforts on characterising the performance of inter-flow NC, a complete understanding of the resulting trade-offs between delay, throughput and energy consumption is far from being reached. This may be partially attributed to the lack of accurate analytical models that incorporate the additional delays resulting from NC: synchronisation delays for jointly coded streams, higher transmission delays in links due to combining packets of different lengths, and decoding delays due to processing and synchronisation at the network output. While the effect of coding delays at intermediate nodes has been considered previously [84, 85, 90], the additional transmission costs due to coding packets of different sizes and sending remedy packets to assist in decoding, as well as the synchronisation constraints at the receivers have been largely ignored. Indeed, most theoretical studies focus on the two-way relay scenario [84–87, 90] in which no decoding delay is incurred, and they assume zero or slotted transmission times which may not capture the overhead of NC. On the other hand, the end-to-end performance of NC in multi-hop networks have been analysed based on fluid-flow models [100] which may not be appropriate for light traffic conditions; network calculus [101, 102] which only provides bounds that may not be tight in practice; or large Markov chains which are only solved numerically [99]. Consequently, there is a need to expand the classical theory of delay models for data networks [16, 17, 106] to address the requirements of NC.

3. A diffusion model for packet travel in non-homogeneous networks

In this chapter we consider the travel process of a packet from a given source to a destination which is at distance D from the packet, but whose whereabouts are unknown, or imprecisely known. Thus the packet may not have precise information about which direction it should pursue as it moves from one hop to the next, and we suppose that the destination node is recognised only when the packet gets close to it, typically one hop away.

Errors in routing information can be due to node failures, infrequent routing table updates which do not keep up with changes in the state of links and nodes, intermittent effects in wireless links that disable certain one-step connections and invalidate the routing tables, and node mobility which can easily invalidate previous routing information. In such circumstances, one can view routers as probabilistic entities [7], and specific schemes for discovering viable paths have been developed to deal with these circumstances [13]. Furthermore, when the network is very large, the packet may end up being dropped by its own finite time-out, and it is also more likely to be lost due to an error or failure in the communication layer or due to buffer overflows in some router.

We therefore study whether a packet ultimately succeeds in reaching its designated destination, how long this will take, and how much energy may be expended, in the context of a network with imperfect routing tables, and non-homogeneous network characteristics. We use a Brownian motion model, along the lines of previous work [8, 9], which accounts for network non-homogeneity regarding routing and packet loss rates.

This work is motivated by three interesting applications. The first case is related to defending a destination node against attacks taking the form of packets which may carry a virus or a worm that can be detected via deep packet inspection (DPI) at some intermediate nodes. Thus as a packet approaches the destination node it may be inspected by intermediate protecting nodes and dropped if it is viewed as a threat; however, the source of the packet will use a time-out to attempt sending the attacking packet forward again and it is interesting to see whether the attacker will eventually be successful. The second one relates to a wireless network where remote areas, away from where the source and destination nodes are located, perhaps have poor wireless coverage so that the packet losses become more frequent as the packet "unknowingly" (due to poor routing tables for instance) meanders away from the source and destination node. A third example of non-homogeneous wireless network occurs when the packet progresses faster as it approaches the destination, for instance when directional information such as a radio signature becomes stronger as the packet approaches the destination node.

In the sequel we will model a packet's motion towards a destination node in an infinite random non-homogeneous network, with packet drops that will stop the packet's progress resulting in a subsequent time-out retransmission of the packet from the source. Generalising the work in [9], we obtain in Section 3.1 an exact expression for the average time and energy that it takes the packet to eventually find the destination node, based on a non-homogeneous Brownian motion model enhanced with some useful point processes representing the relaunch of an aborted or interrupted search. We develop an analytical solution technique based on a finite but unbounded number of internally homogeneous segments, yielding the average travel time and the energy expended. Then, in Section 3.2, the results are applied to the three cases of interest that we have outlined. We also present some approximations which simplify the analytical results when the network has small routing errors. Furthermore, we show how the model can be applied to finite-size networks. Finally, we provide our concluding remarks in Section 3.3.

3.1. Modelling the travel process

Although traditionally most models in computer systems and networks are discrete [107], here we consider a continuous distance $Y(t)$ of the packet to its destination at time $t \geq 0$. The packet starts at distance $Y(0) = D$ and the travel process ends at some time T defined by:

$$T = \inf\{t : Y(t) = 0\},$$

We model the distance $\{Y(t) : t \geq 0\}$ as a diffusion process [6,108] which is a continuous-time Markov process with continuous state space in which small changes occur during small intervals of time, i.e., for small Δt and $Y(t) = z$ the process satisfies the condition:

$$\Pr[|Y(t + \Delta t) - Y(t)| > \epsilon | Y(t) = z] = o(\Delta t)$$

Furthermore, the increment $Y(t + \Delta t) - Y(t)$ is approximately normal with mean and variance:

$$E[Y(t + \Delta t) - Y(t) | Y(t) = z] = b(z, t)\Delta t + o(\Delta t)$$
$$E[\{Y(t + \Delta t) - Y(t)\}^2 - \{E[Y(t + \Delta t) - Y(t)]\}^2 | Y(t) = z] = c(z, t)\Delta t + o(\Delta t)$$

In other words, the limits of the infinitesimal mean and variance of the conditional increment of $Y(t)$ exist and are equal to $b(z,t)$ and $c(z,t)$, respectively. We assume that the process is time homogeneous so that

$$\lim_{\Delta t \to 0} \frac{E[Y(t+\Delta t) - Y(t)|Y(t) = z]}{\Delta t} \;=\; b(z) \;,$$
$$\lim_{\Delta t \to 0} \frac{E[\{Y(t+\Delta t) - Y(t)\}^2 - \{E[Y(t+\Delta t) - Y(t)]\}^2|Y(t) = z]}{\Delta t} \;=\; c(z)$$

When $b(z) < 0$, on average the packet gets closer over time to the destination node, but $b(z) \geq 0$ is also possible.

Let the random variable $s(t)$ represent the state of the packet at time $t \geq 0$; $s(t) \in \{\mathbf{S}, \mathbf{W}, \mathbf{L}, \mathbf{P}\}$ where:

- **S**: the travel is proceeding and the packet's distance from the destination is $Y(t) > 0$. The probability density function (pdf) of the distance $Y(t)$ is represented by $f(z,t)dz = \Pr[z < Y(t) \leq z + dz,\ s(t) = \mathbf{S}]$.

- **W**: the packet's life-span has ended, and so has its travel. This can happen because the packet was destroyed or became lost, and the source was informed via the time-out which is assumed to be exponential with parameter r. After an additional exponentially distributed delay of parameter μ, a new packet is placed at the source and a new travel immediately begins. We write $W(t) = \Pr[s(t) = \mathbf{W}]$.

- **L**: the packet is lost, and the travel is interrupted until a new packet can be sent out; for small Δt and $Y(t) = z > 0$, this happens with probability $\lambda(z)\Delta t + o(\Delta t)$, where $\lambda(z) \geq 0$ is the packet loss rate at distance z. Information about the loss of the packet will be available through the time-out effect; thus the time spent in this state is exponentially distributed with parameter r, after which the travel process enters state **W**. We denote $L(t) = \Pr[s(t) = \mathbf{L}]$.

- **P**: the packet has reached its destination and the travel process ends. However, as an artefact to construct an indefinitely repeating recurrent process in order to simplify the computation of $E[T]$, it is assumed that after one time unit the travel process restarts at the source and a new packet is sent out. We will use the notation $P(t) = \Pr[s(t) = \mathbf{P}]$.

Figure 3.1 shows a high level diagram of the model illustrating the different states that a packet can be in during its travel from a source to a destination. In this abstract representation, the diffusion parameters $b(z)$ and $c(z)$ capture quality of routing as well as packet losses that can be repaired by intermediate network nodes (e.g., due to interference). On the other hand, the loss parameter $\lambda(z)$ represents packet losses that require

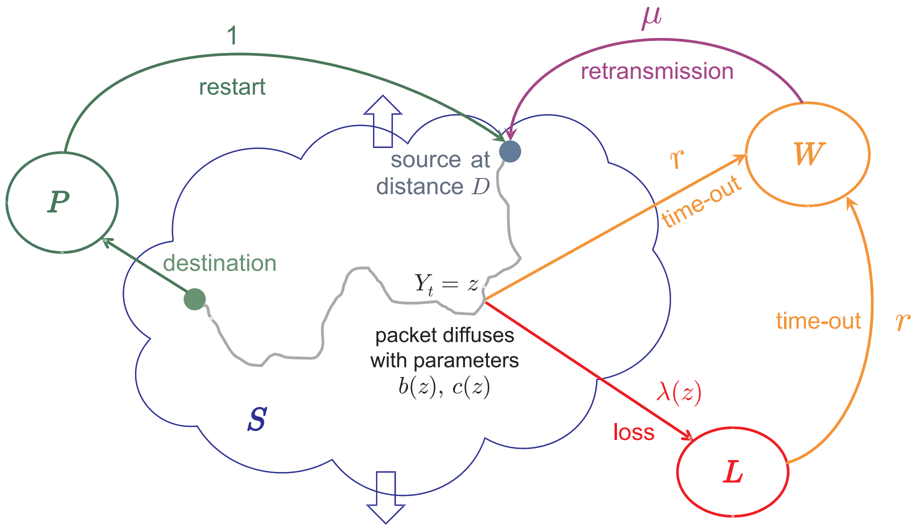


Figure 3.1.: A schematic representation of the diffusion model.

retransmission by the source node, for instance due to node failure, buffer overflow or malicious packet dropping by a relay node.

Notice that **P** is a fictitious state that we use to create a recurrent random process which indefinitely repeats itself. The average total travel time $E[T]$ is the average time that it takes from any successive start of the travel until the first instance when state **P** is reached again. Let $P = \lim_{t\to\infty} P(t)$, then:

$$P = \frac{1}{1 + E[T]}, \quad E[T] = P^{-1} - 1 \tag{3.1}$$

Thus we can compute $E[T]$ by solving the model for P, then applying the above equation.

3.1.1. Equations in the non-homogeneous medium

As discussed earlier, we model the packet's movement in a non-homogeneous network by the pdf $f(z, t)$ that represents the distance of the packet at time $t \geq 0$, and assume that it satisfies a modified version of the distance dependent *diffusion equation* (A.4) in order to take account of the discrete probabilities. We write the equations that the probability density function $f(z, t)$, $z > 0$, and the probability masses $L(t)$, $W(t)$ and $P(t)$, $t \geq 0$ will satisfy:

$$
\begin{aligned}
\frac{\partial f(z,t)}{\partial t} &= \frac{1}{2}\frac{\partial^2[c(z)f(z,t)]}{\partial z^2} - \frac{\partial[b(z)f(z,t)]}{\partial z} - (\lambda(z)+r)f(z,t) \\
&\quad + [P(t)+\mu W(t)]\delta(z-D) \\
\frac{dL(t)}{dt} &= -rL(t) + \int_0^\infty \lambda(z)f(z,t)dz \\
\frac{dW(t)}{dt} &= -\mu W(t) + r[L(t) + \int_0^\infty f(z,t)dz] \\
\frac{dP(t)}{dt} &= -P(t) + \lim_{z\to 0^+}\big[\frac{1}{2}\frac{\partial[c(z)f(z,t)]}{\partial z} - b(z)f(z,t)\big] \\
1 &= P(t)+W(t)+L(t) + \int_0^\infty f(z,t)dz
\end{aligned}
\tag{3.2}
$$

where the distance dependent behaviour of the packet is captured in the drift $b(z)$, instantaneous variance $c(z)$ as well as loss parameter $\lambda(z)$. In effect, this is equivalent to also letting the time-out parameter r be distance dependent because its distance dependent part could be included in $\lambda(z)$. On the other hand, if the time-out has operated then the delay (of average value $1/\mu$) before the search is started again is independent of the distance where the time-out occurred. Note that in practice the time-out is incorporated in the packet itself so that intermediate nodes can discard the packet if the time-out has elapsed, and similarly the source will know the time-out value and eventually it will retransmit a packet whose time-out has elapsed.

3.1.2. Piece-wise approximation for non-homogeneity

We simplify the model of a non-homogeneous medium by considering a finite but unbounded number of "segments" that have different parameters for the Brownian motion describing the packet's movement as a function of its distance to the destination node, while within each segment the parameters are the same. The first segment is in the immediate proximity of the destination node, starting at distance $z = 0$. Each segment may have a different size, and we assume that there are a total of $m < \infty$ segments. By choosing as many segments as we wish, and letting each segment be as small as we wish (all segments need not be of the same length), we can approximate as closely as needed any physical situation that arises where the packet's motion characteristics vary over the distance of the packet to the destination node. We also show that this discrete representation leads to a neat algebraic "product form" representation of the average travel time, and that it thus provides a useful analytic form that offers a more intuitive representation of the analytical results.

We denote by $0 \leq Z_k < \infty$ the boundary between the k-th and $(k+1)$-th segments

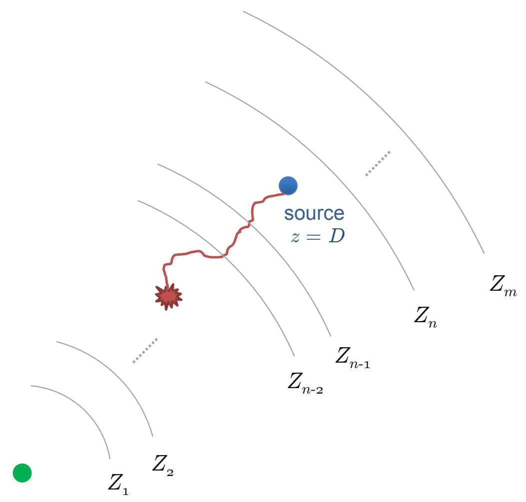


Figure 3.2.: Illustration of the piece-wise approximation for a non-homogeneous packet travel.

with $Z_0 = 0$. The last segment goes from Z_{m-1} to $+\infty$, and we assume that both m and Z_{m-1} are finite but unbounded. Thus for greater accuracy in representing the medium we can take as many segments as we wish, and they may be as small as needed, but they are all finite except the last segment. Thus for $0 \leq k \leq m$, the k-th segment represents the range of distances $Z_{k-1} \leq z < Z_k$, and let $S_k = Z_k - Z_{k-1}$ denote its size. We use n to denote the segment number in which the source node is located, i.e. $Z_{n-1} < D \leq Z_n$. The piece-wise approximation is illustrated in Figure 3.2.

If we use the following notation:

$$\{f(z,t), b(z), c(z), \lambda(z)\} = \{f_k(z,t), b_k, c_k, \lambda_k\} \ , \quad Z_{k-1} < z \leq Z_k$$

Then the differential equation representing the stationary solution of the location dependent diffusion equation for any segment $k \neq n$ is given by:

$$0 = \frac{c_k}{2}\frac{d^2 f_k(z)}{dz^2} - b_k \frac{df_k(z)}{dz} - (\lambda_k + r) f_k(z) \tag{3.3}$$

while the equation for the segment where the source is located is:

$$-[P + \mu W]\delta(z - D) = \frac{c_n}{2}\frac{d^2 f_n(z)}{dz^2} - b_n \frac{df_n(z)}{dz} - (\lambda_n + r) f_n(z) \tag{3.4}$$

We will also have:

$$rL = \sum_{k=1}^{m} \lambda_k \int_{Z_{k-1}}^{Z_k} f_k(z)dz \tag{3.5}$$

$$\mu W = r[L + \sum_{k=1}^{m} \int_{Z_{k-1}}^{Z_k} f_k(z)dz] \tag{3.6}$$

$$P = \lim_{z \to 0^+} [\frac{c_1}{2} \frac{df_1(z)}{dz} - b_1 f_1(z)] \tag{3.7}$$

and the normalisation condition:

$$1 = P + W + L + \sum_{k=1}^{m} \int_{Z_{k-1}}^{Z_k} f_k(z)dz \tag{3.8}$$

3.1.3. Computing average travel time

Result 3.1 *The total average travel time, which is obtained by solving for P so that $E[T] = P^{-1} - 1$, is given by:*

$$E[T] = \Big(\frac{1}{r} + \frac{1}{\mu}\Big)\Big[\sqrt{\frac{b_n^2 + 2c_n(\lambda_n + r)}{b_1^2 + 2c_1(\lambda_1 + r)}} \frac{\mathcal{A}_n^- \mathcal{B}_n^+ e^{u_n S_n} - \mathcal{B}_n^- \mathcal{A}_n^+ e^{v_n S_n}}{\mathcal{B}_n^+ e^{u_n(Z_n - D)} + \mathcal{A}_n^+ e^{v_n(Z_n - D)}} - 1\Big] \tag{3.9}$$

where u_k, v_k are, respectively, the positive and negative real roots of the characteristic polynomial of the stationary differential equation for the k-th segment (3.3)*:*

$$u_k, v_k = \frac{b_k \pm \sqrt{b_k^2 + 2c_k(\lambda_k + r)}}{c_k} \tag{3.10}$$

The remaining parameters in (3.9) *are computed as follows. Define:*

$$\begin{aligned} \alpha_k^- &= \frac{c_k u_k - c_{k-1} v_{k-1}}{c_k(u_k - v_k)}, \quad \beta_k^- = \frac{c_k u_k - c_{k-1} u_{k-1}}{c_k(u_k - v_k)} \\ \alpha_k^+ &= \frac{c_k u_k - c_{k+1} v_{k+1}}{c_k(u_k - v_k)}, \quad \beta_k^+ = \frac{c_k u_k - c_{k+1} u_{k+1}}{c_k(u_k - v_k)} \end{aligned} \tag{3.11}$$

Then set $\mathcal{A}_1^- = 1$ and $\mathcal{B}_1^- = -1$ and for $2 \le k \le n$ compute:

$$\begin{bmatrix} \mathcal{A}_k^- \\ \mathcal{B}_k^- \end{bmatrix} = \begin{bmatrix} \alpha_k^- & \beta_k^- \\ 1 - \alpha_k^- & 1 - \beta_k^- \end{bmatrix} \begin{bmatrix} e^{u_{k-1} S_{k-1}} & 0 \\ 0 & e^{v_{k-1} S_{k-1}} \end{bmatrix} \begin{bmatrix} \mathcal{A}_{k-1}^- \\ \mathcal{B}_{k-1}^- \end{bmatrix} \tag{3.12}$$

Then set $\mathcal{A}_m^+ = 0$ and $\mathcal{B}_m^+ = e^{v_m Z_m}$, and start another computation at $k = m-1$ for $n \leq k \leq m-1$ with:

$$\begin{bmatrix} \mathcal{A}_k^+ \\ \mathcal{B}_k^+ \end{bmatrix} = \begin{bmatrix} \alpha_k^+ & \beta_k^+ \\ 1-\alpha_k^+ & 1-\beta_k^+ \end{bmatrix} \begin{bmatrix} e^{-u_{k+1}S_{k+1}} & 0 \\ 0 & e^{-v_{k+1}S_{k+1}} \end{bmatrix} \begin{bmatrix} \mathcal{A}_{k+1}^+ \\ \mathcal{B}_{k+1}^+ \end{bmatrix} \tag{3.13}$$

This completes the definition of all the terms in $E[T]$ and the proof of (3.9) is provided in the derivation given below. But first let us point to some useful properties of the formula that we have derived.

Remark With n being the index of the discretisation segment that includes the source node at D, it is interesting to see that $E[T]$ only depends on a set of parameters that are computed for values of $k = 1$, $k = n$, and on two sets of algebraic iterations between $k = 1$ and $k = n$ and $k = m$ down to $k = n$.

Remark When the source node is located in the last segment we have $m = n$, and the average travel time takes the much simpler form:

$$E[T] = \frac{r+\mu}{r\mu} \, [\sqrt{\frac{b_n^2 + 2c_n(\lambda_n + r)}{b_1^2 + 2c_1(\lambda_1 + r)}} \mathcal{A}_n^- e^{u_n(D-Z_{n-1})} - 1] \tag{3.14}$$

Furthermore, if we have a homogeneous network with $m = n = 1$ we end up with:

$$E[T] = \frac{r+\mu}{r\mu} \, [e^{u_1 D} - 1] \tag{3.15}$$

as we would expect from [9].

Proof The general solution has the form:

$$f_k(z) = \begin{cases} A_k^- e^{u_k z} + B_k^- e^{v_k z}, & Z_{k-1} \leq z \leq \min(D, Z_k) \\ A_k^+ e^{u_k z} + B_k^+ e^{v_k z}, & \max(D, Z_{k-1}) \leq z \leq Z_k \end{cases}$$

Thus there are $2m+2$ constants to be determined from (a) the boundary conditions at 0 and $+\infty$, (b) the continuity condition of the probability density function at D and at the boundaries between segments, and (c) conditions obtained by integrating the defining differential equation around D and the boundaries between segments. First consider the case $Z_{k-1} \leq z \leq \min(D, Z_k)$; to ensure continuity of the probability density function at $z = Z_{k-1}$ we have:

$$f_k(Z_{k-1}) = f_{k-1}(Z_{k-1}) \tag{3.16}$$

which leads to

$$A_k^- e^{u_k Z_{k-1}} + B_k^- e^{v_k Z_{k-1}} = A_{k-1}^- e^{u_{k-1} Z_{k-1}} + B_{k-1}^- e^{v_{k-1} Z_{k-1}}$$

Furthermore, integrating the differential equation (3.3) from $z = Z_{k-1} - \epsilon$ to $z = Z_{k-1} + \epsilon$ and taking the limit as ϵ tends to 0 yields:

$$\frac{c_k}{2}\frac{df_k(Z_{k-1})}{dz} - \frac{c_{k-1}}{2}\frac{df_{k-1}(Z_{k-1})}{dz} = [b_k - b_{k-1}]f_k(Z_{k-1}) \tag{3.17}$$

or equivalently

$$\begin{aligned} A_k^- u_k e^{u_k Z_{k-1}} + B_k^- v_k e^{v_k Z_{k-1}} = & \\ & \frac{2b_k - c_{k-1}v_{k-1}}{c_k} A_{k-1}^- e^{u_{k-1}Z_{k-1}} + \frac{2b_k - c_{k-1}u_{k-1}}{c_k} B_{k-1}^- e^{v_{k-1}Z_{k-1}} \end{aligned}$$

Solving (3.16) and (3.17), we can write A_k^- and B_k^- in terms of A_{k-1}^- and B_{k-1}^- as:

$$\begin{aligned} A_k^- e^{u_k Z_{k-1}} &= \alpha_k^- A_{k-1}^- e^{u_{k-1}Z_{k-1}} + \beta_k^- B_{k-1}^- e^{v_{k-1}Z_{k-1}} \\ B_k^- e^{v_k Z_{k-1}} &= [1-\alpha_k^-] A_{k-1}^- e^{u_{k-1}Z_{k-1}} + [1-\beta_k^-] B_{k-1}^- e^{v_{k-1}Z_{k-1}} \end{aligned}$$

where α_k^- and β_k^- are defined in (3.11). The above linear equations can be written in matrix form as:

$$\begin{bmatrix} A_k^- e^{u_k Z_{k-1}} \\ B_k^- e^{v_k Z_{k-1}} \end{bmatrix} = \begin{bmatrix} \alpha_k^- & \beta_k^- \\ 1-\alpha_k^- & 1-\beta_k^- \end{bmatrix} \begin{bmatrix} e^{u_{k-1}S_{k-1}} & 0 \\ 0 & e^{v_{k-1}S_{k-1}} \end{bmatrix} \begin{bmatrix} A_{k-1}^- e^{u_{k-1}Z_{k-2}} \\ B_{k-1}^- e^{v_{k-1}Z_{k-2}} \end{bmatrix}$$

From the boundary condition $\lim_{z\to 0^+} f_1(z) = 0$ we have $B_1^- = -A_1^-$. Thus, if we define $A_1^- \mathcal{A}_k^- \triangleq A_k^- e^{u_k Z_{k-1}}$ and $A_1^- \mathcal{B}_k^- \triangleq B_k^- e^{v_k Z_{k-1}}$, then we can compute $\mathcal{A}_k^-$ and $\mathcal{B}_k^-$ recursively using the matrix multiplication in (3.12). Furthermore, the stationary solution of the differential equation for $Z_{k-1} \le z \le \min(D, Z_k)$ becomes:

$$f_k(z) = A_1^-[\mathcal{A}_k^- e^{u_k(z-Z_{k-1})} + \mathcal{B}_k^- e^{v_k(z-Z_{k-1})}] \tag{3.18}$$

where the constant A_1^- is yet to be determined. Next consider a segment k where $z \ge D$, and write the constants A_k^+ and B_k^+ in terms of A_{k+1}^+ and B_{k+1}^+ by solving boundary conditions similar to (3.16) and (3.17) at $z = Z_k$:

$$\begin{aligned} A_k^+ e^{u_k Z_k} &= \alpha_k^+ A_{k+1}^+ e^{u_{k+1}Z_k} + \beta_k^+ B_{k+1}^+ e^{v_{k+1}Z_k} \\ B_k^+ e^{v_k Z_k} &= [1-\alpha_k^+] A_{k+1}^+ e^{u_{k+1}Z_k} + [1-\beta_k^+] B_{k+1}^+ e^{v_{k+1}Z_k} \end{aligned}$$

or equivalently

$$\begin{bmatrix} A_k^+ e^{u_k Z_k} \\ B_k^+ e^{v_k Z_k} \end{bmatrix} = \begin{bmatrix} \alpha_k^+ & \beta_k^+ \\ 1-\alpha_k^+ & 1-\beta_k^+ \end{bmatrix} \begin{bmatrix} e^{-u_{k+1}S_{k+1}} & 0 \\ 0 & e^{-v_{k+1}S_{k+1}} \end{bmatrix} \begin{bmatrix} A_{k+1}^+ e^{u_{k+1}Z_{k+1}} \\ B_{k+1}^+ e^{v_{k+1}Z_{k+1}} \end{bmatrix}$$

Since $f(z)$ is a probability density function we must have $\lim_{z\to\infty} f_m(z) = 0$ which implies that $A_m^+ = 0$, hence the solution for $\max(D, Z_{k-1}) \leq z \leq Z_k$ can be expressed as follows:

$$f_k(z) = B_m^+[\mathcal{A}_k^+ e^{-u_k(Z_k-z)} + \mathcal{B}_k^+ e^{-v_k(Z_k-z)}] \tag{3.19}$$

where $B_m^+\mathcal{A}_k^+ \triangleq A_k^+ e^{u_k Z_k}$ and $B_m^+\mathcal{B}_k^+ \triangleq B_k^+ e^{v_k Z_k}$ leading to the matrix multiplication in (3.13). Note that the initialisation $\mathcal{B}_m^+ = e^{v_m Z_m}$, with $Z_m \to +\infty$, yields the desired solution for the last segment, that is $f_m(z) = B_m^+\mathcal{B}_m^+ e^{-v_m(Z_m-z)} = B_m^+ e^{v_m z}$.

In order to determine A_1^- and B_m^+, consider the n-th segment and apply the continuity condition of $f_n(z)$ at $z = D$ so that:

$$B_m^+[\mathcal{A}_n^+ e^{-u_n(Z_n-D)} + \mathcal{B}_n^+ e^{-v_n(Z_n-D)}] = A_1^-[\mathcal{A}_n^- e^{u_n(D-Z_{n-1})} + \mathcal{B}_n^- e^{v_n(D-Z_{n-1})}] \tag{3.20}$$

Also, integrating the differential equation (3.4) from $z = D - \epsilon$ to $z = D + \epsilon$ and taking the limit as ϵ tends to 0 yields:

$$\begin{aligned}\frac{2[P+\mu W]}{-c_n} &= B_m^+[\mathcal{A}_n^+ u_n e^{-u_n(Z_n-D)} + \mathcal{B}_n^+ v_n e^{-v_n(Z_n-D)}] \\ &\quad - A_1^-[\mathcal{A}_n^- u_n e^{u_n(D-Z_{n-1})} + \mathcal{B}_n^- v_n e^{v_n(D-Z_{n-1})}]\end{aligned} \tag{3.21}$$

From (3.7), the probability P is given by:

$$P = \frac{c_1}{2}(u_1 - v_1)A_1^- = \sqrt{b_1^2 + 2c_1(\lambda_1 + r)}\ A_1^- \tag{3.22}$$

Substituting (3.6) into (3.8) yields:

$$P + \mu W\left(\frac{1}{r} + \frac{1}{\mu}\right) = 1 \tag{3.23}$$

Now solving the system of linear equations (3.20)−(3.23) we can determine A_1^- and B_m^+:

$$\begin{aligned}A_1^- &= C\left[\mathcal{B}_n^+ e^{u_n(Z_n-D)} + \mathcal{A}_n^+ e^{v_n(Z_n-D)}\right] \\ B_m^+ &= C\left[\mathcal{A}_n^- e^{u_n S_n} e^{v_n(Z_n-D)} + \mathcal{B}_n^- e^{v_n S_n} e^{u_n(Z_n-D)}\right]\end{aligned} \tag{3.24}$$

where

$$C = \frac{r\mu/(r+\mu)}{\sqrt{b_n^2 + 2c_n(\lambda_n + r)}} \Bigg\{ \mathcal{A}_n^- \mathcal{B}_n^+ e^{u_n S_n} - \mathcal{B}_n^- \mathcal{A}_n^+ e^{v_n S_n}$$
$$- [1 - \frac{r\mu}{r+\mu}] \sqrt{\frac{b_1^2 + 2c_1(\lambda_1 + r)}{b_n^2 + 2c_n(\lambda_n + r)}} [\mathcal{B}_n^+ e^{u_n(Z_n - D)} + \mathcal{A}_n^+ e^{v_n(Z_n - D)}] \Bigg\}^{-1}$$

Substituting A_1^- in (3.22) yields P from which the average travel time follows directly. ∎

Result 3.2 *In the special case without packet losses and without a time-out ($\lambda_k = 0$, $r = 0$) the average travel time for $b_m < 0$ is:*

$$E[T] = \sum_{k=1}^{n-1} \frac{S_k}{-b_k} + \frac{D - Z_{k-1}}{-b_n}$$
$$+ \sum_{k=1}^{n} \frac{c_k}{2b_k} \mathcal{F}_k \, [e^{\frac{2b_k}{c_k} \min(D, Z_k)} - e^{\frac{2b_k}{c_k} Z_{k-1}}] + \sum_{k=n}^{m} \frac{c_k}{2b_k} \mathcal{G}_k \, [e^{\frac{2b_k}{c_k} Z_k} - e^{\frac{2b_k}{c_k} \max(D, Z_{k-1})}]$$

where

$$\mathcal{F}_1 = \frac{1}{b_1}, \quad \mathcal{F}_k = \Big[\frac{1}{b_k} - \frac{1}{b_{k-1}} + \mathcal{F}_{k-1} e^{2\frac{b_{k-1}}{c_{k-1}} Z_{k-1}}\Big] \; e^{-2\frac{b_k}{c_k} Z_{k-1}}$$
$$\mathcal{G}_k = \Big[\mathcal{F}_n - \frac{e^{-2\frac{b_n}{c_n} D}}{b_n}\Big] \; e^{-2\sum_{j=n+1}^{k} (\frac{b_j}{c_j} - \frac{b_{j-1}}{c_{j-1}}) Z_{j-1}}$$

Proof The proof is similar to that of Result 3.1 and we omit it.

3.1.4. Energy consumption

Note that both delay and energy consumption include the effect of packet loss, time-outs and retransmission. Delay includes all the (possibly multiple) waiting times for time-outs to operate. However we assume that energy is only consumed by a packet while it is actually being forwarded through the network, so that during wait times for retransmissions the packet (which remains stored at the source until final successful delivery) will consume a negligible amount of energy. Thus the average energy consumption $E[J]$ until the packet reaches its destination is [9]:

$$E[J] = (1 + E[T]) \sum_{k=1}^{m} \int_{Z_{k-1}}^{Z_k} f_k(z) dz \tag{3.25}$$

There may be circumstances where packet storage plays a significant role in energy consumption, in which case the total energy consumption can be estimated as being proportional to the total delivery time for the packet. This case will not be considered in the present work.

3.2. Applications

In this section we present some applications of the proposed model which may arise in different physically meaningful environments.

3.2.1. A network with small routing errors

Suppose that the routing tables are reliable except for some errors that occur infrequently. Thus, most of the time the packet moves towards its destination at the top speed allowed via the shortest path. We represent this case as follows:

$$b_k = -1 + \delta_b \ , \quad c_k = \delta_c \quad \text{for} \ \ 1 \leq k \leq m$$

where δ_b and δ_c are small non-negative numbers. Using first order approximation for Taylor expansion we can write

$$\sqrt{b_k^2 + 2c_k(\lambda_k + r)} \simeq 1 - \delta_b + \frac{\delta_b^2}{2} + \delta_c(\lambda_k + r)$$

and from (3.10) we have

$$u_k = \frac{b_k + \sqrt{b_k^2 + 2c_k(\lambda_k + r)}}{c_k} \simeq \frac{\delta_b^2}{2\delta_c} + \lambda_k + r$$

Furthermore if we multiply both the numerator and denominator of v_k by u_k we obtain

$$v_k = \frac{-2(\lambda_k + r)}{b_k + \sqrt{b_k^2 + 2c_k(\lambda_k + r)}} \simeq \frac{-2(\lambda_k + r)}{\frac{\delta_b^2}{2} + \delta_c(\lambda_k + r)} << 0$$

or $v_k \simeq -\infty$ which yields

$$\begin{aligned}
\begin{bmatrix} \mathcal{A}_k^- \\ \mathcal{B}_k^- \end{bmatrix} &\simeq \begin{bmatrix} \alpha_k^- e^{u_{k-1}S_{k-1}} \mathcal{A}_{k-1}^- \\ (1-\alpha_k^-) e^{u_{k-1}S_{k-1}} \mathcal{A}_{k-1}^- \end{bmatrix} \\
&= \begin{bmatrix} \prod_{i=2}^{k} \alpha_i^- e^{u_{i-1}S_{i-1}} \\ (1-\alpha_k^-) e^{u_{k-1}S_{k-1}} \prod_{i=2}^{k-1} \alpha_i^- e^{u_{i-1}S_{i-1}} \end{bmatrix}, \quad 2 \leq k \leq n \\
\begin{bmatrix} \mathcal{A}_k^+ \\ \mathcal{B}_k^+ \end{bmatrix} &\simeq \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \quad n \leq k \leq m
\end{aligned}$$

Thus the total average travel time can be approximated as follows

$$\begin{aligned}
E[T] &\simeq \frac{r+\mu}{r\mu} [\sqrt{\frac{b_n^2 + 2c_n(\lambda_n + r)}{b_1^2 + 2c_1(\lambda_1 + r)}} \mathcal{A}_n^- e^{u_n S_n} e^{-u_n(Z_n - D)} - 1] \\
&= \frac{r+\mu}{r\mu} [\sqrt{\frac{b_n^2 + 2c_n(\lambda_n + r)}{b_1^2 + 2c_1(\lambda_1 + r)}} \prod_{k=2}^{n} \alpha_k^- e^{u_n D} \; e^{\sum_{k=1}^{n-1}(u_k - u_n) S_k} - 1] \\
&\simeq \frac{r+\mu}{r\mu} [K \; e^{(\lambda_n + r)D} \; e^{\sum_{k=1}^{n-1}(\lambda_k - \lambda_n) S_k} - 1] \qquad (3.26)
\end{aligned}$$

where

$$K = e^{\frac{\delta_b^2}{2\delta_c} D} [1 + \frac{\delta_c(\lambda_n - \lambda_1)}{(1-\delta_b)^2 + 2\delta_c(\lambda_1 + r)}] \prod_{k=2}^{n} [1 - \frac{\delta_c(\lambda_k - \lambda_{k-1})}{1 + (1-\delta_b)^2 + 2\delta_c(\lambda_k + r)}]$$

Note that the accuracy of the above approximation improves as the errors in routing (i.e., δ_b and δ_c) tend to zero. If there is no uncertainty in routing, then $K = 1$ and the above expression is exact.

3.2.2. Retarding an attacking packet

An example of practical interest occurs when the packet that we are modelling contains some form of attack on the destination node, such as a virus or a worm. Also, we suppose that the network protects this particular node by introducing a capability at intermediate nodes to detect the contents of the packet and to drop it. This procedure usually involves deep packet inspection (DPI) where intermediate nodes have to perform additional operations, such as reading and checking the content of the packet during routing. The specific mechanisms for malicious packet detection are out of scope of this research; we do however realise that detection may not be perfect and this is reflected in the model. If the packet is dropped by the defense mechanism or lost due to imprecise routing, the sender will send the attacking packet again after a time-out. The question is then whether it is possible to block the attack indefinitely or whether to the contrary the attacking packet will eventually reach the destination node that is being defended.

We first examine this problem in the context of a wired network that uses shortest path routing. Thus if the distance D refers to the number of hops from source to destination, and if the routers are operating properly, we will have a drift $b = -1$ and a variance $c = 0$ throughout the network. In other words, each transmission will send the packet one hop closer to the destination node. More generally if there is no uncertainty in routing $c_k = 0$ and $b_k < 0$, and it can be shown that the total average travel time does not depend on the network's parameters for $z > D$:

$$E[T] = \frac{r+\mu}{r\mu}\ [e^{\frac{\lambda_n + r}{|b_n|}D}\ e^{\sum_{k=1}^{n-1}\left(\frac{\lambda_k+r}{|b_k|} - \frac{\lambda_n+r}{|b_n|}\right)S_k} - 1] \qquad (3.27)$$

Furthermore if the routers are perfect and always provide shortest distance routing we have $b_k = -1$ and the approximation in (3.26) is exact and reduces to:

$$E[T] = \frac{r+\mu}{r\mu}\ [e^{(\lambda_n + r)D}\ e^{\sum_{k=1}^{n-1}(\lambda_k - \lambda_n)S_k} - 1] \qquad (3.28)$$

Now let us introduce a non-homogeneous packet drop effect by choosing an integer n to create an acceleration in the packet drop effect and let $S_k = D/(n-1)$ so that:

$$E[T] = \frac{r+\mu}{r\mu}\ [e^{(r + \frac{\sum_{k=1}^{n-1}\lambda_k}{n-1})D} - 1] \qquad (3.29)$$

which yields the following result.

Result 3.3 *If* $\lim_{n\to\infty} \frac{\sum_{k=1}^{n-1}\lambda_k}{n-1} = +\infty$ *then the packet will never reach the destination node. Otherwise it will reach it in a time which is finite on average, and with probability one.*

Figure 3.3 illustrates Result 3.3 by showing that even with a small excess, represented by $\theta > 1$, above the $O(n)$ rate of increase for the loss rate λ_k the attacking packet's progress will be indefinitely impeded by the drops, despite the subsequent time-outs.

A phase transition effect

The destruction of the packet and the time-out will both relaunch the search for the destination node allowing the attacker to improve its chances to find it. Figure 3.4 shows that if the node is heavily defended when the attacking packet gets very close to it, then the attack may never take place. Specifically, if the packet loss rate is $\log \lambda_k = \frac{1}{k\rho}$, then as ρ becomes very small $E[T]$ and the energy consumed tend to infinity despite the fact that near the origin the search speed is greater $b_k = -0.25 + 0.5(k-1)/(m-1)$ and its randomness is smaller $c_k = 0.5 + 0.5(k-1)/(m-1)$.

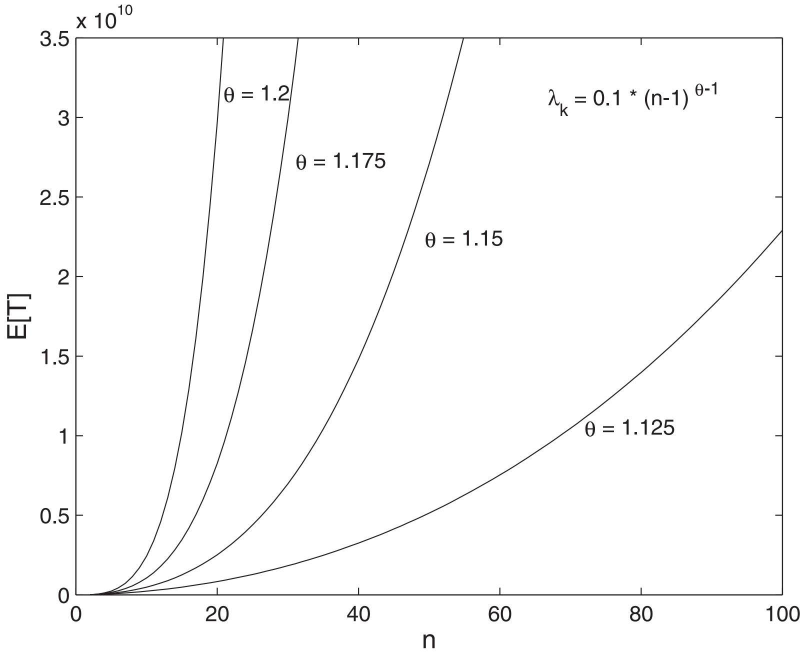


Figure 3.3.: $E[T]$ versus the source segment n when the segment size $S_k = D/(n-1)$ and the loss rate $\lambda_k = 0.1(n-1)^{\theta-1}$ for different values of θ; $b_k = -1$, $c_k = 0$, $\mu = 0.1$, $r = 0.02$, and $D = 100$.

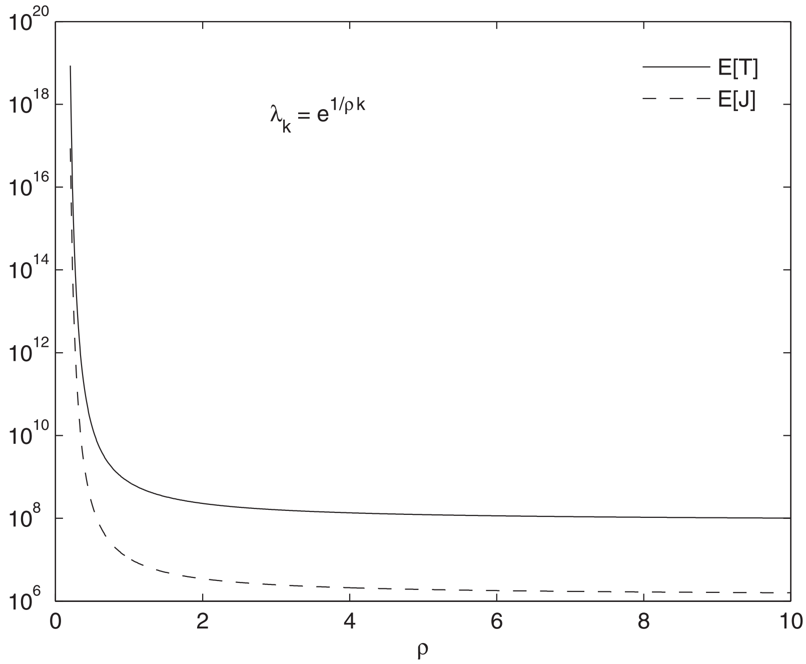


Figure 3.4.: $E[T]$ and average energy consumption $E[J]$ (logarithmic scale) versus ρ when the loss rate $\lambda_k = e^{\frac{1}{k\rho}}$, $r = 0.05$, $D = 10$, $\mu = 0.025$ and $S_k = 1, k < m = 20$.

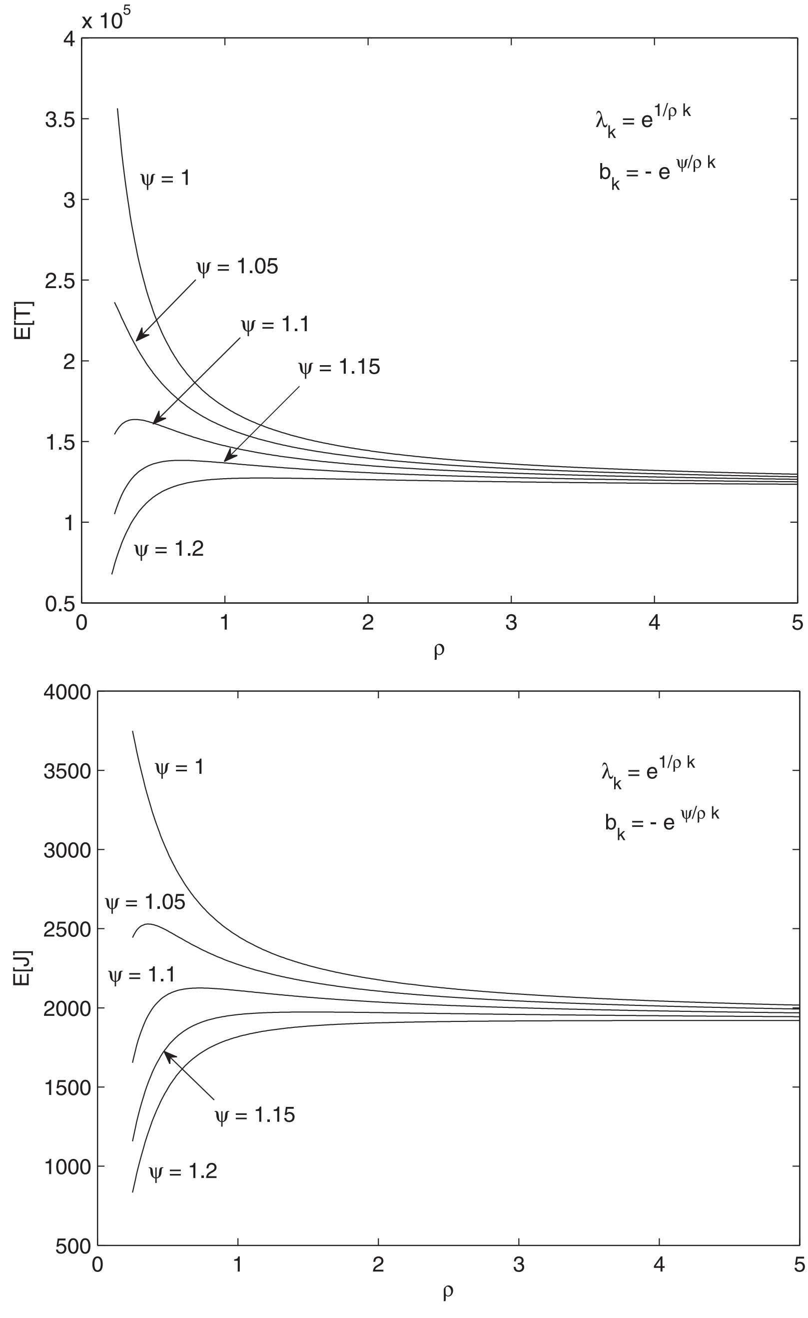


Figure 3.5.: Phase transition effect for average (a) travel time and (b) energy consumption versus ρ when varying the value of ψ in $\lambda_k = e^{1/(\rho k)}$ and $b_k = -e^{\psi/(\rho k)}$; $c_k = 1$, $D = 10$, $r = 0.05$, $\mu = 0.025$ and $S_k = 1,\ k < m = 20$.

However it is interesting to see that if the packet's speed of approach to the destination grows faster than the rate at which the packet may be destroyed, then both $E[T]$ and $E[J]$ remain finite and may tend to zero, while in the opposite case they will tend to infinity, as shown in Figure 3.5, presenting a form of phase transition.

3.2.3. A neighbourhood with traps

Suppose that routers in the neighbourhood of the destination node within a distance S contains "traps" that can identify the attacking packet and drop it. Thus we take $m = n = 2$, so that $E[T]$ is obtained from (3.14) with $\lambda_2 = 0$ and $\lambda_1 > 0$:

$$E[T] = \frac{r+\mu}{r\mu}[\sqrt{\frac{b_2^2 + 2c_2 r}{b_1^2 + 2c_1(\lambda_1 + r)}}\mathcal{A}_2^- e^{u_2(D-S)} - 1]$$

Figure 3.6 shows the manner in which $E[T]$ sharply increases with λ_1, for S ranging between 10 and 15, $D = 100$, $b_2 = b_1 = 0.25$ and $c_1 = c_2 = 1$. Also $\mu = 1/10$ and r is set to the value that minimises $E[T]$ when $\lambda_1 = 0$ and $S = 10$. Figure 3.7 shows how $E[T]$ varies with S, with the same parameters and different values of λ_1. The logarithmic scale shows markedly how the average time it takes to reach the object being sought increases by orders of magnitude as S and λ_1 are increased. Figures 3.8 and 3.9, with $S = 10$, $b_2 = 0.25$ and the same set of parameters, show that even small increases (more negative) in average speed at which the packet approaches its objective can reduce average travel time by an order of magnitude, yet $E[T]$ is still very large.

Figures 3.10 and 3.11 raise the question about how to select S and λ_1 together in order to maximise the protection offered to the destination node. If we keep the same set of parameters as previously but take λ_1 to be inversely proportional to S in Figure 3.10 so that the average number of sources of protection, placed at rate λ_1, remains constant in proportion to the protection space of size S. The mapping of time rate to spatial rate will remain constant for any fixed value of b_1 which is the speed of approach inside the protected neighbourhood. In this context, we examine whether there is a size S^* of the protected neighbourhood which maximises protection, i.e., that *maximises* the average time to locate the destination node. Figure 3.10 shows that there is indeed an optimum size of protection space $S = S^*$ that maximises the delay before the attacking packet can reach the destination node, and that it varies with the speed b_1 of the packet inside the protected neighbourhood. As the speed increases, the optimum size of the neighbourhood gets smaller. This follows from the fact that we have taken $\lambda_1 \approx 1/S$: a smaller size implies a higher "rate of protection" and hence more frequently occurring destructions of the packet which compensate for the higher speed of the packet. However, the corresponding maximum values of $E[T]$ do become smaller as the packet's speed increases. In Figure 3.11

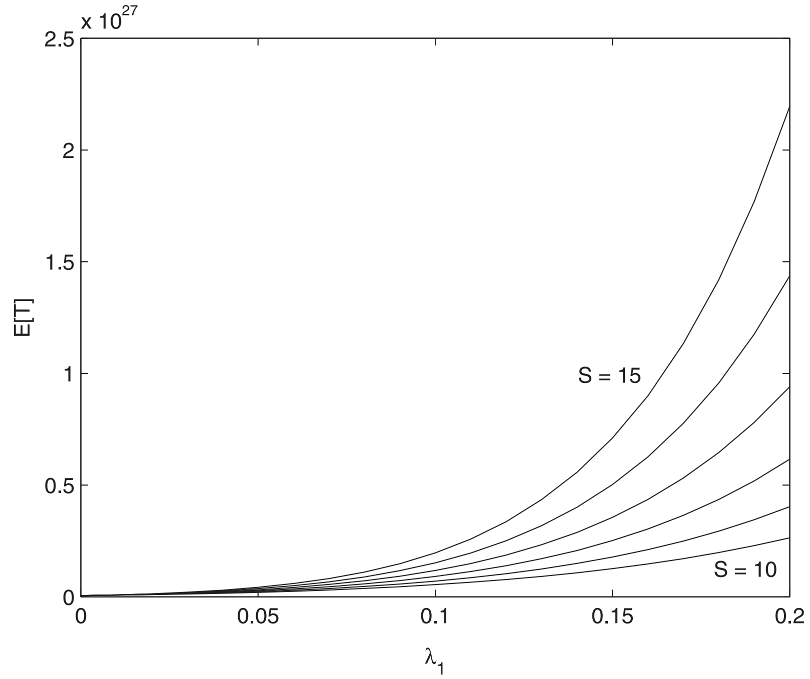


Figure 3.6.: Average travel time $E[T]$ versus the loss rate λ_1 in the protected neighbourhood S for S between 10 and 15 with a step size of 1.

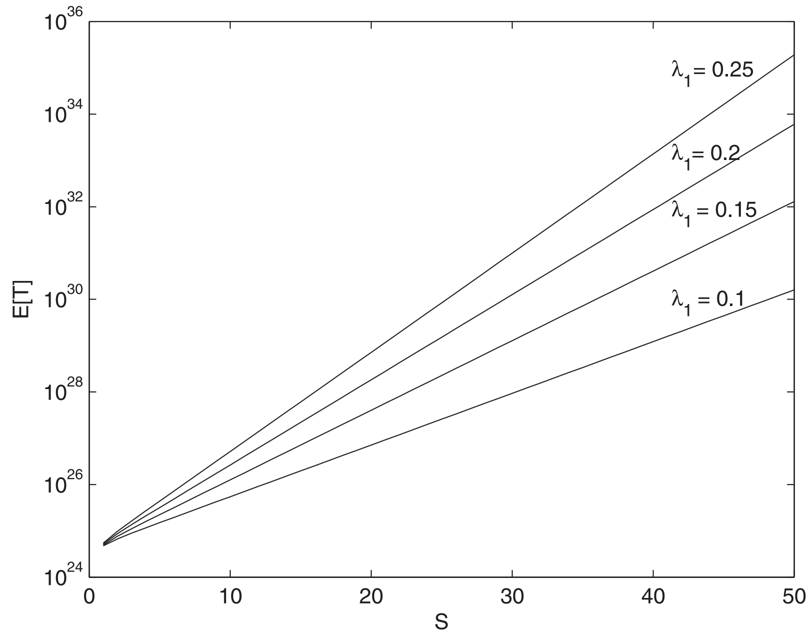


Figure 3.7.: $E[T]$ (logarithmic scale) versus size of the protected neighbourhood S for different values of loss rate λ_1.

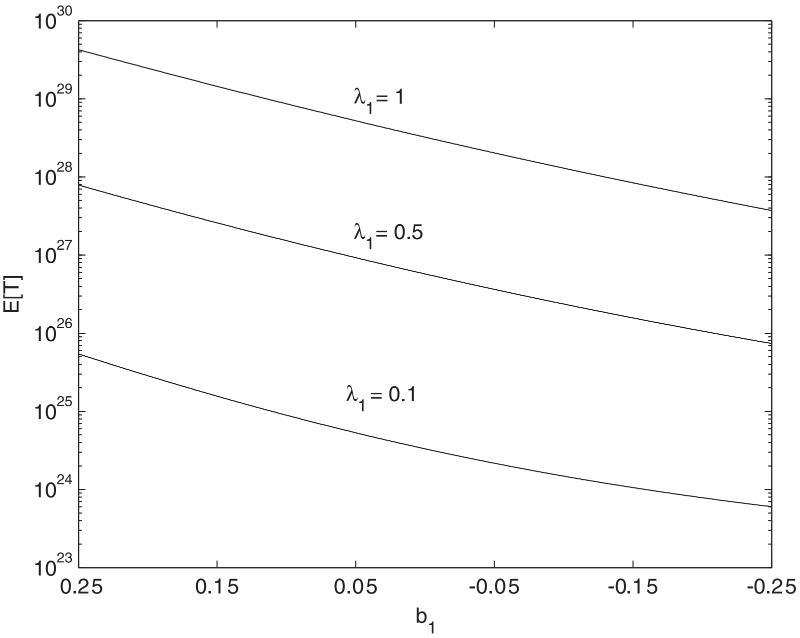


Figure 3.8.: $E[T]$ (logarithmic scale) versus travel speed b_1 inside the protected neighbourhood for different values of loss rate λ_1.

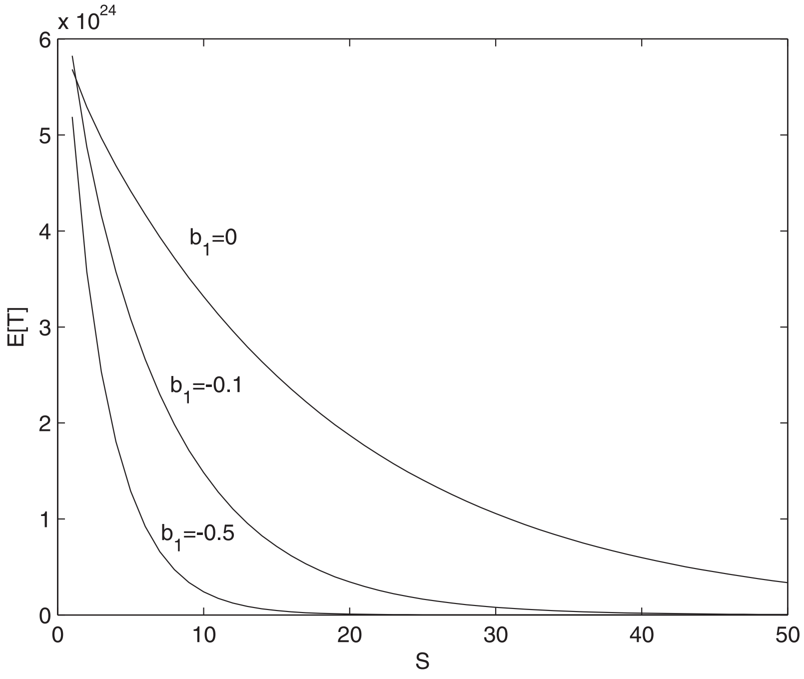


Figure 3.9.: $E[T]$ versus size of the protected neighbourhood S for $D = 100$ and different values of travel speed b_1.

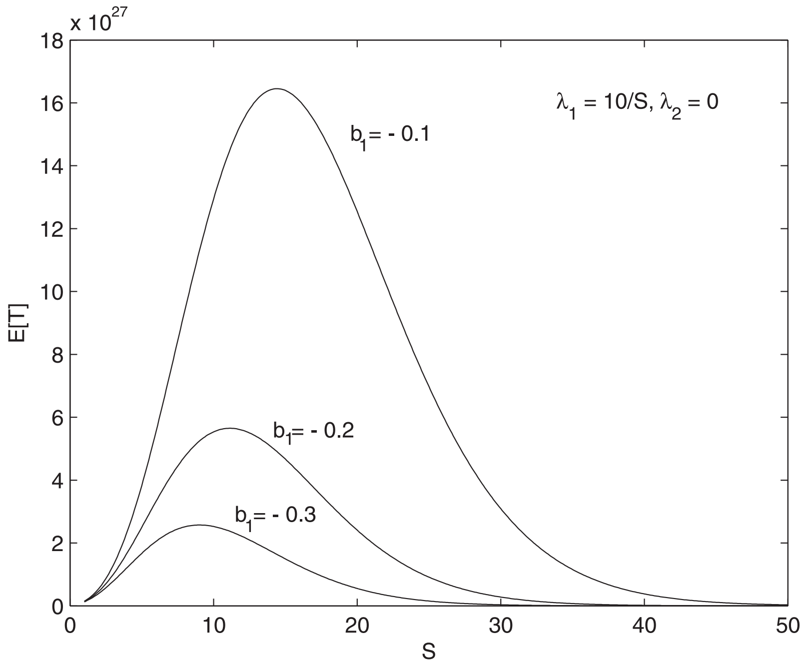


Figure 3.10.: $E[T]$ versus size of protected neighbourhood S when loss rate $\lambda_1 = 10/S$ for different values of travel speed b_1. The optimum protection size needed becomes smaller so that λ_1 increases when the travel speed increases.

we set $b_1 = b_2 = 0.25$ and Λ is varied in $\lambda_1 = \Lambda/S^2$. The results are similar to the previous ones.

For the examples of Figures 3.10 and 3.11 the average energy expenditure is closely proportional to $E[T]$ because $\sum_{k=1}^{m} \int_{Z_{k-1}}^{Z_k} f_k(z)dz \simeq 1$ so that we omit showing the numerical results for the energy.

3.2.4. Wireless networks

Another interesting case arises when a packet that moves far away from its initial point and from the destination node, has a greater chance of being lost or destroyed. This could represent a multi-hop wireless network deployed in a very large area; as the packet moves to remote areas far from the region where the source and destination are located, the nodes that the packet might visit are less likely to handle it and more likely to just discard it. This can also represent a network where there are fewer nodes in remote areas and inter-node communications in such areas are less reliable. As an example consider 100 segments with $S = 1$ and a loss rate that increases with distance: $\lambda_k = k\ell,\ \ell > 0,\ 1 \le k \le 99$, $\lambda_{100} = 100\ell$. If average speed of the packet's motion and its second moment remain constant with $b_k = 0$ and $c_k = 1$ for $1 \le k \le 100$, the results with $D = 10$ in Figure 3.12 show that a relatively short time-out is needed to optimise the average travel time, but that

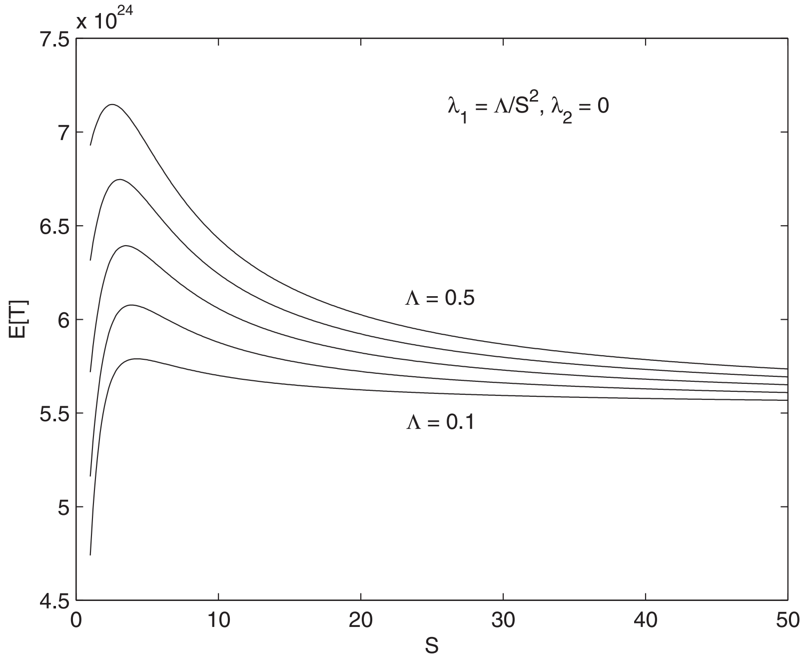


Figure 3.11.: $E[T]$ versus size of protected neighbourhood S when loss rate $\lambda_1 = \Lambda/S^2$ for $\Lambda = 0.1$ to 0.5. The protection area needed to maximise the search time decreases as Λ increases.

the resulting optimum is nevertheless very large. Similar results are shown in Figure 3.13 where the loss rate varies as $\lambda_k = \theta^k - 1, \theta > 1$ with $S_k = 2.5,\ k < m = 15$, but the packet's average speed of approach towards the destination and its instantaneous variance improve as it gets closer to the destination node with $b_k = -0.25 + 0.5(k-1)/(m-1)$ and $c_k = 0.5 + 0.5(k-1)/(m-1)$.

Figures 3.14 and 3.15 show the locus of $E[T]$ with $E[J]$ when the average time-out $1/r$ is varied. The effect of varying the distance D between the source and destination nodes is illustrated in Figure 3.14. It is interesting to note that when the time-out is small, energy consumption decreases because the packet spends less time travelling, while delay increases because many potentially successful search attempts will be interrupted by the time-out. On the other hand, large time-outs increase both delay and energy consumption since the source node will spend significant time before realising that a search attempt is unsuccessful. These results indicate that both delay and energy consumption are strongly influenced by the time-out, since a smaller value of D does not necessarily guarantee better performance. In Figure 3.15, the loss rate increases with distance according to $\lambda_k = 10^{-4}k/S$ with $S_k = S, k < m$ and $Z_{m-1} = 100$; on the other hand the speed and the uncertainty in motion improve as the packet approaches the destination node with $b_k = -0.5 + (k-1)/(m-1)$ and $c_k = 0.75 + 0.25(k-1)/(m-1)$; when S decreases,

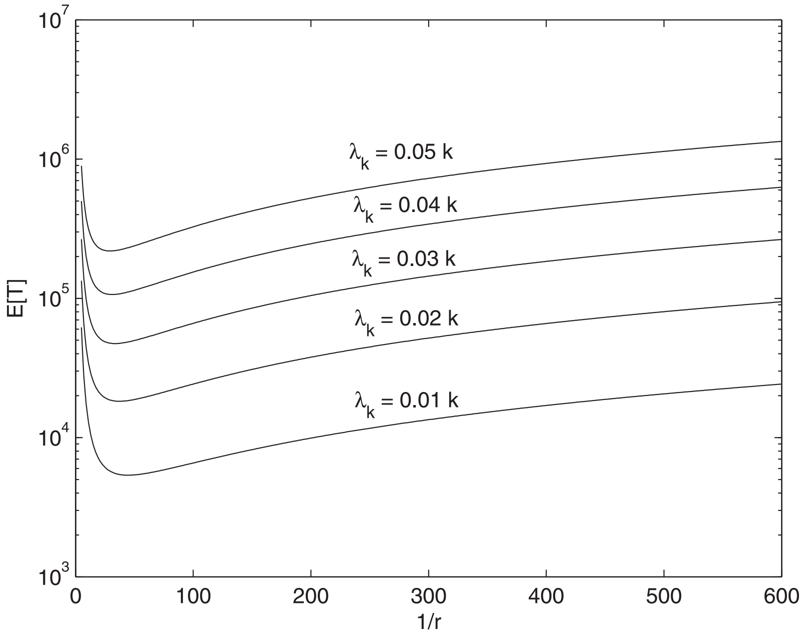


Figure 3.12.: $E[T]$ versus the average time out $1/r$ when loss rates increases linearly and the number of segments $m = 100$.

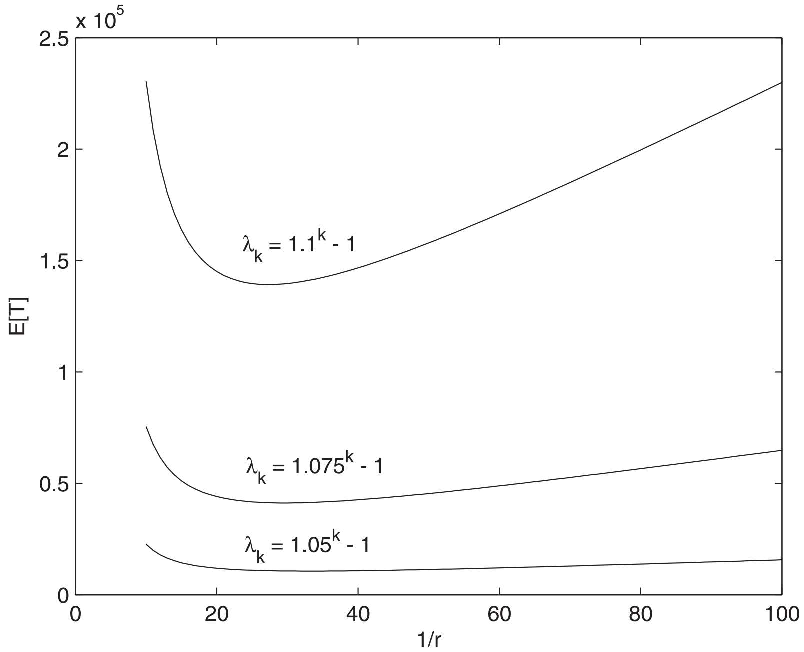


Figure 3.13.: $E[T]$ versus the average time-out $1/r$ when loss rates increase geometrically and the number of segments $m = 15$.

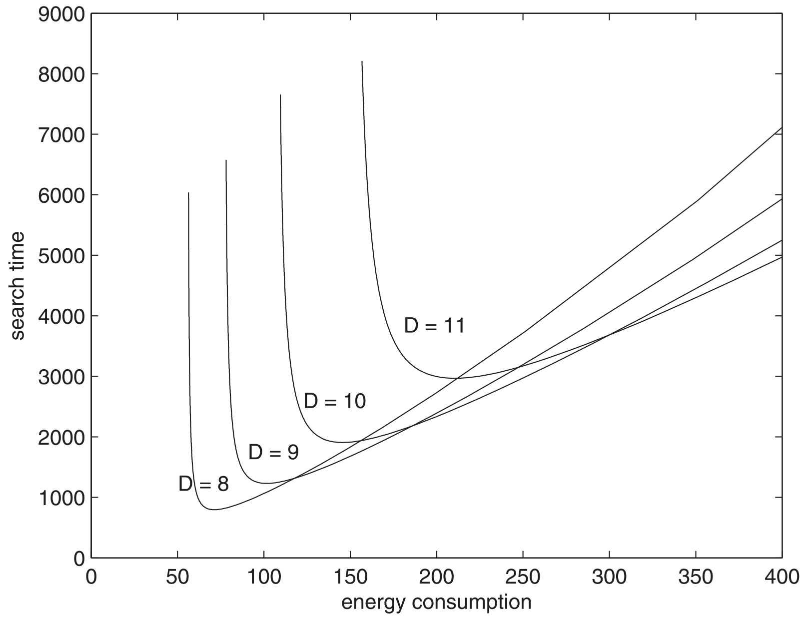


Figure 3.14.: The locus of $E[T]$ and $E[J]$ when the time-out $1/r$ is varied for $D = 8, 9, 10$ and 11. Loss rates increase with distance according to $\lambda(z) = \frac{1}{4}[1 - e^{-z/10}]$ which is segmented with $m = 42$; $b_k = -0.25, c_k = 1$ and $\mu = 0.025$.

increasing the loss rates but also improving the speed at which the packet reaches the destination, energy decreases because less time is spent in actual motion, while delay increases because in proportion more time is spent waiting and then restarting after the packet is lost. Note that with high enough loss rates ($S = 0.15$) the minimum travel time does not coincide with minimum energy consumption.

Next we examine how routing information should be distributed in a large network, that is whether it is more useful to concentrate routing information around the destination node or distribute them on a larger area at the cost of accuracy. To this end, we construct networks that use "equivalent" resources by choosing $b_1 = -b_m$ and $b_k = b_1 + (b_m - b_1)(k-1)/(m-1)$ so that $\sum_{k=1}^{m} b_m = 0$ for different values of $b_1 \leq 0$. Figure 3.16(a) shows an illustration of the compared networks where darker colours indicate areas with more accurate routing information. The results in Figure 3.16(b) are obtained for a network with high loss rates around the destination node defined by a continuous function $\lambda(z) = 0.01 + ze^{-0.35z}$ which is segmented with $m = 74$; also, we set $c_k = 1$, $\mu = 0.025$ and $D = 10$ (which lies in the 53rd segment and close to the last one in the discretised function). The figure shows that the higher the speed of the packet near the destination node (i.e., more negative b_1), the better the performance despite the fact that routing tables become less reliable in remote areas where the source node is located. These results, however, are specific to the above example and do not generalise to other networks.

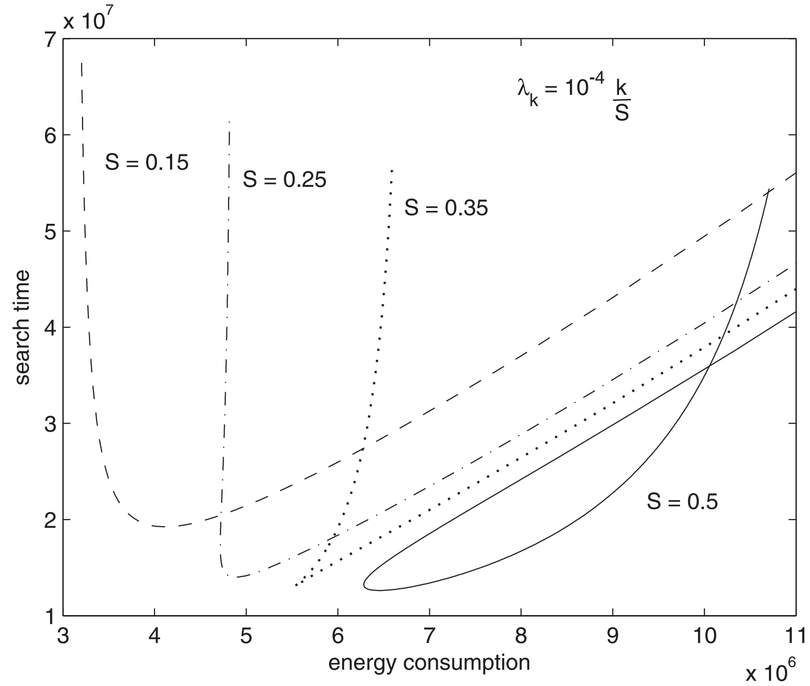


Figure 3.15.: The locus of $E[T]$ and $E[J]$ resulting from varying the time-out $1/r$ when loss rates and travel speeds depend on the segment size $S_k = S$; $D = 10$ and $\mu = 0.05$.

3.2.5. Search in a bounded environment

As a final example, consider a wireless sensor network in which nodes are distributed over a finite area of radius $R \geq D$, and all packets are routed to a centrally located sink [11,22]. If the network is sufficiently dense, then we can approximate packet routing by a continuous diffusion process. To further simplify matters, assume a homogeneous medium so that the average travel time can be obtained by first substituting $n = 1$, $m = 2$ and $S_1 = R$ into (3.9) yielding:

$$E[T] = \frac{r+\mu}{r\mu}\ [\frac{e^{u_1 R} + \frac{\beta_1^+}{1-\beta_1^+} e^{v_1 R}}{e^{u_1 (R-D)} + \frac{\beta_1^+}{1-\beta_1^+} e^{v_1 (R-D)}} - 1]$$

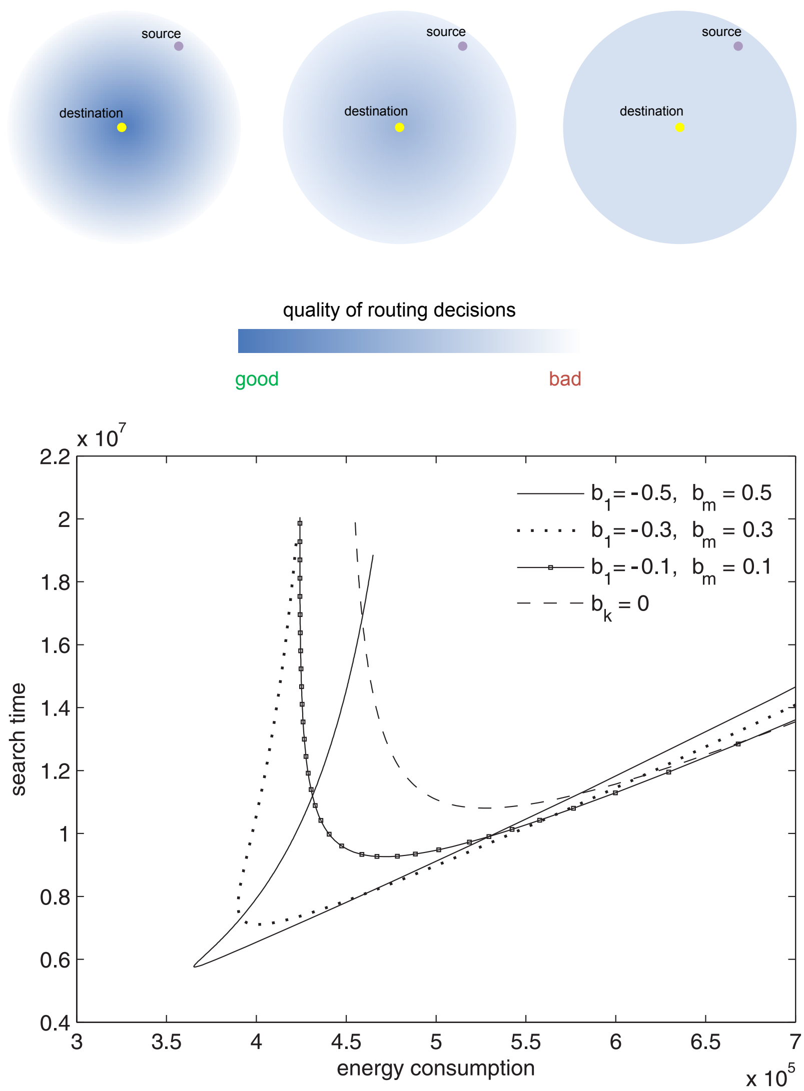


Figure 3.16.: (a) Illustration of networks with "equivalent" routing resources, where darker colours indicate more accurate routing decisions, (b) locus of $E[T]$ and $E[J]$ when the average time out $1/r$ is varied for different distributions of routing information.

Next we need to account for the edge effect. If the boundary is reflecting, then we have $c_2 = 0$ and $b_2 < 0$ yielding $\beta_1^+/(1-\beta_1^+) = -u_1/v_1$ or:

$$\begin{aligned} E[T] &= \frac{r+\mu}{r\mu}\,[\frac{u_1 e^{v_1 R} - v_1 e^{u_1 R}}{u_1 e^{v_1(R-D)} - v_1 e^{u_1(R-D)}} - 1] \\ &= \frac{D}{-b_1} - \frac{c_1}{2b_1^2} e^{\frac{2b_1 R}{c_1}} \big[e^{\frac{-2b_1 D}{c_1}} - 1\big], \quad \lambda_1, r = 0,\ b_1 \neq 0 \\ &= \frac{D}{c_1}[2R - D], \qquad\qquad \lambda_1, r, b_1 = 0 \end{aligned} \tag{3.30}$$

On the other hand, if packets are discarded at the boundary then we can place an absorbing barrier by taking $\lim_{\lambda_2\to\infty} \beta_1^+/(1-\beta_1^+) = -1$ so that:

$$E[T] = \frac{r+\mu}{r\mu}\,[\frac{e^{u_1 R} - e^{v_1 R}}{e^{u_1(R-D)} - e^{v_1(R-D)}} - 1] \tag{3.31}$$

3.3. Summary

We have constructed a Brownian motion model to represent a packet's travel to a destination node in a very large non-homogeneous network. A mixed analytical-numerical method has been developed to compute the average packet travel time and the energy it consumes. We observe that the degree of non-homogeneity of the network will significantly affect the average travel time and energy consumed. The role of time-outs to optimise these quantities has been exhibited, and several examples have been detailed. We considered wireless networks where packet losses (for instance due to insufficient wireless network coverage) increase as the packet reaches areas which are remote from the source and destination nodes. By varying the value of the time-out and studying the locus of the energy expended versus the time taken by the search, we have noticed desirable operating areas where both of these parameters of interest are minimised. We also modelled an attacking packet which may be detected and destroyed as it approaches the destination node, but in turn the attacking packet may progress more rapidly as it approaches the destination node, for instance because a directional routing being used may become more accurate. Comparing the increasing speed of approach of the packet with the possible steeper defenses of the destination node, we observe that there may be conditions whereby despite the use of time-outs the attacking packet may never make it to the destination node, while in other circumstances the attack will be successful. Finally, we illustrated how the model can be used to capture specific environments such as bounded search spaces and networks with small routing errors.

In the next chapter, we derive the time-dependent properties of the packet travel process in a homogeneous environment and use the analysis to obtain the distributions of the delivery time and energy consumption when multiple coded packets are transmitted into the network. We also validate the accuracy of the diffusion model through a simulation study of a grid topology.

4. Time-dependent analysis of coded transmission in homogeneous networks

In Chapter 3, a mixed analytical-numerical technique was developed to evaluate the *average* packet forwarding delay and the energy consumption for a single packet travelling in a non-homogeneous network. While the expected performance can be useful in many cases of interest, it is not sufficient to provide worst-case guarantees that can only be inferred from the distribution. The latter, however, requires analysing the time-dependent behaviour of the packet travel process which is difficult to obtain but useful to know in order to evaluate the effect of uncertainties in the network, such as packet losses, inaccuracies or errors in routing, and possible energy limitations. It is also valuable if one wishes to evaluate different means for improving performance at the price of higher energy costs by sending out multiple duplicate packets or using erasure coding techniques.

Thus in this chapter we consider the probability distribution of both the forwarding delay and energy consumption for a single as well as multiple simultaneously transmitted packets. Characterising the energy aspects of sending multiple packets is particularly challenging, since it requires knowledge of the distribution of energy expended by each individual packet at every instant of time. As a result, we will focus on homogeneous networks in order to make the analysis more tractable.

The remainder of the chapter is organised as follows. In Section 4.1 we obtain the time-dependent solution of the density function of the distance of a packet to its destination, and derive the Laplace transform (LT) of the probability density of total packet forwarding delay, total energy consumption and the energy expended up to time t by a single packet. The analysis is extended in Section 4.2 to the case where the source sends either duplicate or coded packets that follow independent paths to the destination node so as to improve reliable delivery and reduce effective travel times. The results are illustrated by several examples in Section 4.3, where we evaluate the performance of a scheme which imposes a limit on the total energy consumption per packet; furthermore we investigate the resulting gain in performance, and loss in increased energy consumption, due to coding at the source node. Section 4.4 summarises the main outcomes of the chapter.

4.1. Time-dependent solution of the diffusion model

We will first compute the distribution of packet travel time and energy consumption by modifying the recurrent approach in Chapter 3 where we assumed that when the packet reaches the destination node at $z = 0$ it remains there for one time unit then jumps to $z = D$ and diffuses anew. The aim was to construct a synthetic ergodic process which simplifies the computation of expected performance. In contrast, since we are concerned with the time-dependent solution, we place an absorbing barrier at the destination. Because the network is assumed to be homogeneous, we have $b(z) = b$, $c(z) = c$ and $\lambda(z) = \lambda$. From the above assumptions, the equations governing the pdf $f(z, t)$ and the probability masses $L(t)$ and $W(t)$ become:

$$\begin{aligned}
\frac{\partial f(z,t)}{\partial t} &= \frac{c}{2}\frac{\partial^2 f(z,t)}{\partial z^2} - b\frac{\partial f(z,t)}{\partial z} - (\lambda + r)f(z,t) + \mu W(t)\delta(z - D) \\
\frac{dL(t)}{dt} &= -rL(t) + \lambda \int_{0^+}^{\infty} f(z,t)dz \\
\frac{dW(t)}{dt} &= -\mu W(t) + r[L(t) + \int_{0^+}^{\infty} f(z,t)dz]
\end{aligned}$$

Moreover, the probability that the packet has not reached the destination by time t is:

$$\Pr[T > t] = L(t) + W(t) + \int_{0^+}^{\infty} f(z,t)dz$$

The cumulative distribution function (cdf) of the packet travel time $G(t) = \Pr[T \le t]$ can be deduced from the other quantities we defined since the probabilities sum to one:

$$1 = G(t) + L(t) + W(t) + \int_{0^+}^{\infty} f(z,t)dz$$

and as a direct consequence we obtain the pdf of the travel time:

$$g(t) = \frac{dG(t)}{dt} = \lim_{z \to 0^+} [-bf(z,t) + \frac{c}{2}\frac{\partial f(z,t)}{\partial z}] \tag{4.1}$$

Note that from the definition of the probability current $I(z, t)$ in (A.5), $g(t)$ is given by the flow of probability mass from the interval $(0, \infty)$ to the origin, namely

$$g(t) = -\lim_{z \to 0^+} I(z,t)$$

The initial condition of the diffusion process is

$$f(z, 0) = \delta(z - D)$$

while the boundary conditions are

$$f(0,t)=0, \qquad \lim_{z\to+\infty} f(z,t)=0$$

If the LT of a function $a(t)$ is written as:

$$\bar{a}(s)=\int_0^\infty a(t)e^{-st}dt$$

where s is a complex number, then the LT of the equations describing the travel process are:

$$\begin{aligned}
-[1+\mu\bar{W}(s)]\delta(z-D) &= \frac{c}{2}\frac{\partial^2\bar{f}(z,s)}{\partial z^2}-b\frac{\partial\bar{f}(z,s)}{\partial z}-(s+\lambda+r)\bar{f}(z,s) && (4.2)\\
\bar{L}(s) &= \frac{\lambda}{s+r}\int_{0^+}^\infty \bar{f}(z,s)dz && (4.3)\\
\bar{W}(s) &= \frac{r}{s+\mu}\,[\bar{L}(s)+\int_{0^+}^\infty \bar{f}(z,s)dz] && (4.4)\\
\bar{g}(s) &= \frac{c}{2}\lim_{z\to0^+}\frac{\partial\bar{f}(z,t)}{\partial z} && (4.5)
\end{aligned}$$

and the normalisation condition regarding the sum of the probabilities gives:

$$\frac{1}{s}=\bar{L}(s)+\bar{W}(s)+\frac{\bar{g}(s)}{s}+\int_{0^+}^\infty \bar{f}(z,s)dz \tag{4.6}$$

We now summarise our first result.

4.1.1. Travel time

Result 4.1 *The LT of the pdf of the travel time is given by*

$$\bar{g}(s)=\frac{(s+\mu)(s+r)}{\mu r+s(s+\mu+r)e^{\omega_1 D}} \tag{4.7}$$

Furthermore, for $z\geq 0$ we have:

$$\bar{f}(z,s)=\frac{\bar{g}(s)}{\sqrt{b^2+2c(s+\lambda+r)}}\left[e^{\frac{b}{c}z}e^{\frac{\sqrt{b^2+2c(s+\lambda+r)}}{c}(D-|z-D|)}-e^{\omega_2 z}\right] \tag{4.8}$$

where ω_1,ω_2 are the roots of the characteristic polynomial $\frac{c}{2}\omega^2-b\omega-(s+\lambda+r)=0$

$$\omega_1,\omega_2=\frac{b\pm\sqrt{b^2+2c(s+\lambda+r)}}{c}$$

Proof The solution of the differential equation (4.2) has the form:

$$\bar{f}(z,s) = \begin{cases} Ae^{\omega_1 z} + Be^{\omega_2 z}, & z < D \\ Ce^{\omega_1 z} + Ee^{\omega_2 z}, & z > D \end{cases}$$

The boundary condition $\bar{f}(0,s) = 0$ implies that $B = -A$, and the condition $\bar{f}(\infty, s) = 0$ requires that $C = 0$. To ensure continuity of $\bar{f}(z,s)$ at $z = D$ we have $A(e^{\omega_1 D} - e^{\omega_2 D}) = Ee^{\omega_2 D}$ so that $E = A(e^{\omega_1 D} - e^{\omega_2 D})e^{-\omega_2 D}$ and consequently

$$\bar{f}(z,s) = \begin{cases} A[e^{\omega_1 z} - e^{\omega_2 z}] & , \quad z \leq D \\ A[e^{(\omega_1 - \omega_2)D} - 1]\ e^{\omega_2 z} & , \quad z \geq D \end{cases}$$

which can be written concisely as:

$$\bar{f}(z,s) = A\Big[e^{\frac{b}{c}z} e^{\frac{\sqrt{b^2+2c(s+\lambda+r)}}{c}(D-|z-D|)} - e^{\omega_2 z}\Big] \tag{4.9}$$

Applying the above equation to (4.5) we get:

$$\bar{g}(s) = \frac{c}{2}A(\omega_1 - \omega_2), \quad \text{or} \quad A = \frac{\bar{g}(s)}{\sqrt{b^2 + 2c(s + \lambda + r)}}$$

substituting A above in (4.9) yields the expression for $\bar{f}(z,s)$ in (4.8). Now it remains to determine $\bar{g}(s)$; integrating (4.2) in the interval $D \pm \epsilon$ and taking the limit as $\epsilon \to 0$ yields:

$$1 + \mu \bar{W}(s) = \frac{c}{2}A(\omega_1 - \omega_2)e^{\omega_1 D} = \bar{g}(s)e^{\omega_1 D} \tag{4.10}$$

We can also substitute (4.4) into (4.6) to obtain:

$$1 = s\bar{W}(s)\big[1 + \frac{s + \mu}{r}\big] + \bar{g}(s) \tag{4.11}$$

and the result for $\bar{g}(s)$ follows by solving (4.10) and (4.11). ∎

Note that the average packet travel time can be found from $E[T] = -\lim_{s\to 0} \frac{d\bar{g}(s)}{ds}$.

Remark In the special case without packet losses and without a time-out ($\lambda = 0$, $r = 0$) our analysis agrees with previous results [48, 49, 109]:

$$\bar{g}_0(s) = e^{-\frac{D}{c}\ (b+\sqrt{b^2+2cs})}$$

$$\bar{f}_0(z,s) = \frac{e^{-\frac{b}{c}(D-z)}}{\sqrt{b^2 + 2cs}}\big[e^{\frac{-\sqrt{b^2+2cs}}{c}|z-D|} - e^{\frac{-\sqrt{b^2+2cs}}{c}(z+D)}\big] \tag{4.12}$$

that can be easily inverted to obtain:

$$g_0(t) = \frac{D}{\sqrt{2\pi ct^3}}\, e^{-\frac{(D+bt)^2}{2ct}}$$

$$f_0(z,t) = \frac{e^{-\frac{b^2 t}{2c}}\, e^{-\frac{b}{c}(D-z)}}{\sqrt{2\pi ct}}\big[e^{-\frac{(z-D)^2}{2ct}} - e^{-\frac{(z+D)^2}{2ct}}\big] \tag{4.13}$$

Here the subscript 0 is used to indicate that the quantity is relevant to pure Brownian motion. The cdf of the travel time $G_0(t) = \Pr[T_0 \le t]$ can also be derived using other standard results [110]:

$$G_0(t) = \frac{1}{2}\left[\operatorname{erfc}\left(\frac{D+bt}{\sqrt{2ct}}\right) + e^{-2\frac{b}{c}D}\operatorname{erfc}\left(\frac{D-bt}{\sqrt{2ct}}\right)\right] \tag{4.14}$$

where erfc is the complementary error function, $\operatorname{erfc}(x) = \frac{2}{\sqrt{\pi}}\int_x^\infty e^{-y^2}dy$. In this case the condition $b < 0$ is necessary in order to have a finite average travel time $E[T_0] = -D/b$. Moreover if $b > 0$ then there is a non-zero probability that the packet will never reach the destination node; more precisely we have:

$$G_0(\infty) = \begin{cases} 1 & , \quad b \le 0 \\ e^{-2\frac{b}{c}D} & , \quad b > 0 \end{cases}$$

Remark We can also express the density $f(z,t)$ in terms of $f_0(z,t)$ as noted in [111]:

$$f(z,t) = f_0(z,t)e^{-(\lambda+r)t} + \int_0^t \mu W(\tau)e^{-(\lambda+r)(t-\tau)}f_0(z,t-\tau)d\tau \tag{4.15}$$

where the first term is the probability that the packet reaches distance z in time t without being lost or interrupted by a time-out, while the second term is the probability that the travel is restarted at some time $\tau \in [0,t]$ and that distance z is reached in a time interval $t-\tau$ without interruption. While (4.15) is easier to interpret in the time domain, the proof is straightforward if we take the LT of the right hand side yielding:

$$[1+\mu\bar{W}(s)]\bar{f}_0(z,s+\lambda+r) = \bar{g}(s)e^{\omega_1 D}\bar{f}_0(z,s+\lambda+r) = \bar{f}(z,s)$$

4.1.2. Energy consumption

Result 4.2 *The LT of the pdf of the packet's total energy expenditure J is given by:*

$$\bar{h}(s) = \frac{s+\lambda+r}{se^{\omega_1 D}+\lambda+r} \tag{4.16}$$

which does not depend on $1/\mu$ the average delay for packet retransmission, after the time-out elapses.

Proof During any travel epoch, the packet's travel can be interrupted by its loss occurring according to a Poisson process of rate λ, and by the time-out which occurs independently and is exponentially distributed with parameter r. Let τ_λ and τ_r be mutually independent and exponentially distributed random variables representing the time to the first loss and the time to the first time-out, respectively. Let $\gamma_\iota(t)$ be the pdf of the duration of a packet travel time until its first interruption. From previous analysis, T_0 is the random variable representing the total travel time of a packet from source to destination if it were not interrupted and its pdf is given by (4.13). Thus we have:

$$\gamma_\iota(t)dt = \Pr[t \leq \min(\tau_\lambda, \tau_r) \leq t + dt,\ T_0 > t] \tag{4.17}$$

or

$$\gamma_\iota(t) = (\lambda + r)e^{-(\lambda+r)t}[1 - G_0(t)], \quad \bar{\gamma}_\iota(s) = \frac{\lambda + r}{s + \lambda + r}\,[1 - \bar{g}_0(s + \lambda + r)] \tag{4.18}$$

A packet's travel may be interrupted several times in this manner, and after each interruption it will (after some time) be sent out again. On the other hand, its last and hence successful attempt at reaching its destination will have a duration whose pdf is defined as:

$$\gamma_d(t)dt = \Pr[t \leq T_0 \leq t + dt,\ \ \min(\tau_\lambda, \tau_r) > t]$$

or

$$\gamma_d(t) = g_0(t)e^{-(\lambda+r)t}, \quad \bar{\gamma}_d(s) = \bar{g}_0(s + \lambda + r) \tag{4.19}$$

Since the total energy expenditure is proportional to the time spent travelling, it can be obtained by accounting for the possibilities of reaching the destination in $1, 2, \ldots$ attempts without including the time spent in the lost and wait-for-retransmission states. Since each attempt is independent of its predecessors, the LT of the pdf of J is simply:

$$\bar{h}(s) = \frac{\bar{\gamma}_d(s)}{1 - \bar{\gamma}_\iota(s)} \tag{4.20}$$

and the result follows by substituting (4.18) and (4.19) into (4.20). ∎

4.1.3. Energy expended up to time t

In this section we generalise the results presented in the previous section. We first derive the *joint* distribution of the total energy consumption J and travel time T. This will then allow us to analyse the distribution of the random variable $J(t)$ representing the energy consumed by a packet up to time $t \geq 0$, with the total energy expenditure of the travel process being $J(\infty) \equiv J$. These two results will finally be applied to obtain the distribution of energy consumption of erasure coding and duplication.

Lemma 4.3 *Let $\phi(x,t)$ denote the joint pdf of the total energy consumption and travel time, i.e., $\phi(x,t)dxdt = \Pr[x \leq J \leq x+dx,\ t \leq T \leq t+dt]$ and*

$$\int_0^\infty \int_x^\infty \phi(x,t)dt\ dx = 1$$

Then its two-dimensional LT, with complex variables ξ and s, is given by:

$$\tilde{\phi}(\xi,s) = \int_0^\infty \int_x^\infty \phi(x,t)e^{-\xi x - st}dt\ dx = \frac{\bar{\gamma}_d(s+\xi)}{1-\bar{\psi}(s)\bar{\gamma}_\iota(s+\xi)} \tag{4.21}$$

where $\psi(t)$ denotes the pdf of the time interval between the loss or time-out of a packet and the retransmission of a new one:

$$\psi(t) = \frac{r}{\lambda + r}\ \mu e^{-\mu t} + \frac{\lambda}{\lambda + r}\int_0^t re^{-ry}\mu e^{-\mu(t-y)}dy$$

whose LT takes the form

$$\bar{\psi}(s) = \frac{\mu r}{\lambda + r}\ \frac{s+\lambda+r}{(s+\mu)(s+r)}$$

Proof If the packet is successful in reaching the destination node in its first attempt then $J = T$. On the other hand, if at least one retransmission takes place then T will exceed J by the amount of time spent in the lost or wait-for-retransmission states. Therefore:

$$\begin{aligned}
\phi(x,t) &= \gamma_d(t)\delta(t-x) + \int_0^x \gamma_\iota(y)\psi(t-x)\gamma_d(x-y)dy \\
&+ \int_0^x \int_0^{x-y_1} \int_0^{t-x} \gamma_\iota(y_1)\gamma_\iota(y_2)\gamma_d(x-y_1-y_2)\psi(t-x-y_3)\psi(y_3)dy_3dy_2dy_1 + \cdots
\end{aligned}$$

If we take the LT with respect to the time variable t we get:

$$\begin{aligned}
\bar{\phi}(x,s) = {} & \gamma_d(x)e^{-sx} + \bar{\psi}(s)e^{-sx}\int_0^x \gamma_\iota(y)\gamma_d(x-y)dy \\
& + \bar{\psi}(s)^2 e^{-sx}\int_0^x \int_0^{x-y_1} \gamma_\iota(y_1)\gamma_\iota(y_2)\gamma_d(x-y_1-y_2)dy_2dy_1 + \cdots
\end{aligned}$$

and (4.21) follows by taking the LT with respect to the energy variable x and summing the resulting infinite geometric series. ∎

Note that the marginal distributions derived previously can be obtained from the above lemma:

$$\bar{g}(s) = \tilde{\phi}(0, s), \quad \bar{h}(\xi) = \tilde{\phi}(\xi, 0)$$

Furthermore, the LT of the pdf of the total time during which the travel process is suspended (due to loss or time-out) is given by:

$$E[e^{-s(T-J)}] = \tilde{\phi}(-s, s) = \frac{\bar{g}_0(\lambda + r)}{1 - \bar{\psi}(s)[1 - \bar{g}_0(\lambda + r)]} = \frac{\Pr[T_0 < \min(\tau_\lambda, \tau_r)]}{1 - \bar{\psi}(s)\Pr[T_0 > \min(\tau_\lambda, \tau_r)]}$$

Lemma 4.4 *Let $h(x, t)$ denote the pdf of the energy consumed by the packet up to time $t \geq 0$, i.e., $h(x, t)dx = \Pr[x \leq J(t) \leq x + dx]$ and*

$$\int_0^t h(x, t)dx = 1$$

Then:

$$\begin{aligned} \tilde{h}(\xi, s) &= \int_0^\infty \int_x^\infty h(x, t)e^{-\xi x - st}dt\ dx \\ &= \bar{\phi}(\xi, s)\left[\frac{1}{s} + \frac{\bar{\gamma}_\iota(s + \xi)}{\bar{\gamma}_d(s + \xi)}\left\{\frac{1}{\lambda + r} + \frac{1 - \bar{\psi}(s)}{s}\right\}\right] \end{aligned} \tag{4.22}$$

Proof At time $t \geq 0$, the packet can be in one of the four states described in Chapter 3, that is $s(t) \in \{\mathbf{P}, \mathbf{S}, \mathbf{L}, \mathbf{W}\}$ which we consider below:

(a) The packet *reached the destination* at some time $\tau \leq t$:

$$h_d(x, t) = \frac{\partial}{\partial x}\Pr[J(t) \leq x,\ s(t) = \mathbf{P}] = \int_x^t \phi(x, \tau)d\tau \tag{4.23}$$

(b) The packet is *searching* for the destination node:

$$\begin{aligned} h_s(x, t) &= \frac{\partial}{\partial x}\Pr[J(t) \leq x,\ s(t) = \mathbf{S}] = e^{-(\lambda+r)t}[1 - G_0(t)]\delta(x - t) \\ &\quad + \int_0^x \gamma_\iota(y)e^{-(\lambda+r)(x-y)}[1 - G_0(x - y)]\psi(t - x)dy + \cdots \end{aligned} \tag{4.24}$$

In the first term, no time-out or loss has occurred up to t and consequently the total energy consumption is equal to t. The i-th term corresponds to the case where at time t the packet is in the i-th attempt to find the destination node (i.e., it has been retransmitted $i - 1$ times); therefore the pdf of the energy utilisation up to t is given by the convolution of the pdf of $i - 1$ interrupted search periods, each followed by an idle period, and a single search period which does not end before the time instant t.

(c) The travel process of the packet is *interrupted* due to a loss or a time-out:

$$\begin{aligned} h_\iota(x,t) &= \frac{\partial}{\partial x}\Pr[J(t) \le x,\ s(t) \in \{\mathbf{L}, \mathbf{W}\}] = \gamma_\iota(x)[1 - \Psi(t-x)] \\ &+ \int_0^x \int_0^{t-x} \gamma_\iota(y_1)\gamma_\iota(x-y_1)\psi(y_2)[1 - \Psi(t-x-y_2)]dy_2 dy_1 + \cdots \end{aligned} \tag{4.25}$$

where $\Psi(t) = \int_0^t \psi(\tau)d\tau$. Here the i-th term denotes the case where the time instant t occurs after the packet's travel process is suspended i times but before retransmission of the $(i+1)$-th packet.

From the law of total probability, the pdf of $J(t)$ can be obtained as:

$$h(x,t) = h_d(x,t) + h_s(x,t) + h_\iota(x,t)$$

If we take the two-dimensional LT for the above equation we obtain:

$$\tilde{h}(\xi,s) = \frac{\tilde{\phi}(\xi,s)}{s} + \frac{\frac{1-\bar{g}_0(s+\xi+\lambda+r)}{s+\xi+\lambda+r} + \bar{\gamma}_\iota(s+\xi)\frac{1-\bar{\psi}(s)}{s}}{1-\bar{\psi}(s)\bar{\gamma}_\iota(s+\xi)}$$

and formula (4.22) follows by substituting the identities:

$$\frac{1-\bar{g}_0(s+\xi+\lambda+r)}{s+\xi+\lambda+r} = \frac{\bar{\gamma}_\iota(s+\xi)}{\lambda+r} \quad \text{and} \quad \frac{1}{1-\bar{\psi}(s)\bar{\gamma}_\iota(s+\xi)} = \frac{\bar{\phi}(s,\xi)}{\bar{\gamma}_d(s+\xi)}$$

from (4.18) and (4.21), respectively. ∎

Remark The LT of $E[J(t)]$ is given by:

$$-\lim_{\xi\to 0}\frac{\partial \tilde{h}(\xi,s)}{\partial \xi} = \frac{(s+\mu)(s+r)[e^{\omega_1 D}-1]}{s(s+\lambda+r)[\mu r + s(s+\mu+r)e^{\omega_1 D}]} = \frac{\bar{g}(s)}{s}\frac{e^{\omega_1 D}-1}{s+\lambda+r} \tag{4.26}$$

Remark In the absence of packet loss and a time-out mechanism we have:

$$\tilde{h}_0(\xi,s) = \frac{\bar{g}_0(s+\xi)}{s} + \frac{1-\bar{g}_0(s+\xi)}{s+\xi}$$

which can be inverted to obtain:

$$h_0(x,t) = \begin{cases} g_0(x) + [1-G_0(t)]\delta(t-x) & ,\ x \le t \\ 0 & ,\ x > t \end{cases} \tag{4.27}$$

and

$$E[J_0(t)] = \int_0^t [1-G_0(\tau)]d\tau = \frac{D}{|b|} - \int_t^\infty [1-G_0(\tau)]d\tau \tag{4.28}$$

4.2. Erasure coding and replication

In this section we compare two known redundancy techniques for improving reliable packet delivery: in the first one the source node sends N duplicate packets along independent paths, while in the second technique "k-out-of-N" coding is used and all packets are sent along independent paths. As in [14] we assume that the N transmitted packets do not interfere with each other so that their travel times $T_1, T_2, \ldots, T_N$ are iid random variables. Furthermore, we assume that the receiver can decode a block as soon as it receives the first arriving k packets. Since replication is a special case of coding with $k = 1$ we will focus our discussion on the latter technique. Our preceding analysis allows us to compare the merits and limitations of these two approaches both in terms of reliable packet delivery times and energy efficiency.

4.2.1. Decoding delay

Let $T_{1,N}, T_{2,N}, \ldots, T_{N,N}$ be the random variables obtained by arranging the packet travel times of the N transmitted packets $T_i,\ i = 1,\ \ldots\ , N$ in ascending order

$$T_{1,N} \leq T_{2,N} \leq \ldots \leq T_{N,N}$$

$T_{k,N}$ is the k-th *order statistic* [112], and it denotes the time required to decode a message of size k when the source sends $N \geq k$ packets. The cdf of the decoding time $G_{k,N}(t) = \Pr[T_{k,N} \leq t]$ is given by

$$G_{k,N}(t) = \sum_{i=k}^{N} \binom{N}{i} G(t)^i [1 - G(t)]^{N-i} \tag{4.29}$$

while the pdf can be written as

$$g_{k,N}(t) = N \binom{N-1}{k-1} G(t)^{k-1} [1 - G(t)]^{N-k} g(t) \tag{4.30}$$

Asymptotic analysis

Let $G^{-1}(p)$ denote the *quantile function* of the distribution of the travel time:

$$G^{-1}(p) = \inf\{t : G(t) \geq p\}, \quad 0 < p < 1 \tag{4.31}$$

It is well known that [14, 113] if the ratio k/N has a limit p as the number of transmitted packets N becomes sufficiently large, that is $k/N \to p$ as $N \to \infty$, then $T_{k,N}$ is the p-th sample quantile and is asymptotically normally distributed

$$T_{k,N} \sim \mathcal{N}\left(G^{-1}(p), \frac{p(1-p)}{N\ [g(G^{-1}(p))]^2}\right) \tag{4.32}$$

It follows that as N tends to infinity, the distribution of the decoding delay converges to a constant which is equal to the p-th quantile of the original distribution. A direct implication of this result is that we can characterise the average decoding delay by simply evaluating the p-th quantile of the travel time of a single packet.

4.2.2. Energy consumption

Now we consider the energy aspects of sending out multiple packets. Since energy utilisation per packet is proportional to the time spent travelling in the network, there is a trade-off between having a small number of packets which travel for a long time in the network and a large number of packets which may spend a shorter time. A sensor network typically have source and intermediate nodes which are very simple and have limited wireless range. Destination nodes which are in charge of collecting the sensory information and forwarding it to some external supervisor may however use a radio channel with enough power to reach all the sensor nodes in one hop. This can then be used to abort any further transmission within the network after k packets have been received [114]. If such a feedback mechanism is not available then each of the remaining $N-k$ packets, which is still being forwarded after decoding is successful, will keep propagating in the network until either lost, destroyed by the time-out mechanism, or received by the destination node, whichever happens first. In this case the following result represents a *lower* bound estimate to the total energy consumption:

Result 4.5 *The pdf of the total energy utilisation of k-out-of-N coding up to the time of receipt of k packets is given by:*

$$\begin{aligned} h_{k,N}(x) \;=\; & N\binom{N-1}{k-1}\int_{t=0}^{\infty}\int_{\sum_{i=1}^{N-1} y_i \le x} \\ & \phi(x-\sum_{i=1}^{N-1} y_i, t)dt \prod_{i=1}^{k-1} h_d(y_i,t)dy_i \prod_{i=k}^{N-1} \big[h_s(y_i,t)+h_\iota(y_i,t)\big]dy_i \end{aligned} \tag{4.33}$$

Proof For the total consumption to be equal to x at some time t, it is necessary that exactly $k-1$, 1 and $N-k$ packets arrive at the destination node in the intervals $[0,t]$, $[t, t+dt]$ and $[t+dt, \infty]$ respectively, and that the energy expended by each individual packet is at most t while their sum is x. The probabilities that a packet arrives in the three respective intervals while consuming w units of energy up to t are $h_d(w,t)dw$, $\phi(w,t)dwdt$

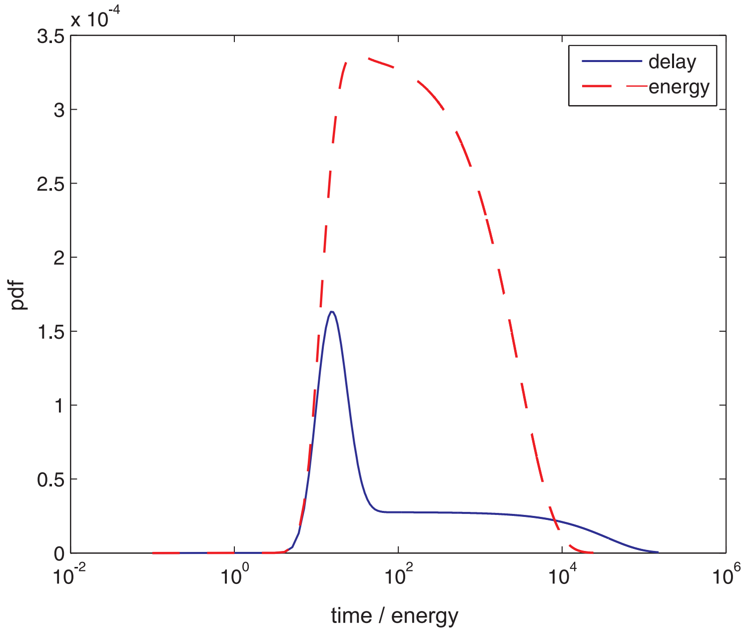


Figure 4.1.: The density functions of the total travel time $g(t)$ and energy consumption $h(t)$ for $b = 0.1$, $c = 1$, $\mu = 0.1$, $r = 0.01$, $\lambda = 0.1$ and $D = 10$.

and $[h_s(w,t) + h_\iota(w,t)]dw$; the result then follows by integrating over all possible values of w and t. ∎

4.3. Numerical results

In this section we illustrate the analytical results with several numerical examples. However, since it may not be possible to obtain the inverse LT of $\bar{g}(s)$ and $\bar{h}(s)$ analytically, we invert them numerically using a MATLAB code [115] based on the algorithm in [116]. The inversion of $\tilde{\phi}(\xi, s)$ and $\tilde{h}(\xi, s)$ is also achieved numerically using a Mathematica program [117] which employs a two-step procedure that concatenates the two one-dimensional inversion algorithms in [118]. In the first step, ξ is regarded as constant and the inversion is performed with respect to s yielding $\bar{\phi}(\xi, t)$ and $\bar{h}(\xi, t)$. In the second step, the transforms are inverted with respect to ξ regarding t as constant in order to finally obtain the desired approximation for $\phi(x,t)$ and $h(x,t)$.

Figure 4.1 compares the density functions for total travel time and energy consumption when both packet losses and uncertainty in routing are high. One can notice that the travel time exhibits a long-tail distribution which is apparent from the logarithmic scale on the x-axis.

Figure 4.2 compares analytical predictions $g(t)$ and $h(t)$ with simulation results for packet traversal through a grid topology. In the simulation, time is assumed to be slotted and at any time slot, a transmission occurs which may take the packet one step towards the destination with probability q or relay it to a node which is further away from the destination with probability $1-q$. Thus the parameter q reflects the quality of the routing tables as it measures the probability of making a correct routing decision at any time slot. Furthermore, in the simulation we assume time-outs are constant since in practice, the source may know an upper bound for the round-trip delay in the network, and as a result it will set a fixed value for the time-to-live (TTL) field in the packet's header. Losses in the simulation are assumed to occur according to a Bernoulli loss model whereby at each hop along the path, the packet is lost with probability $p_\lambda \simeq 1 - e^{-\lambda}$ or it is successfully transmitted to the next hop with probability $1 - p_\lambda$. From the above assumptions, the mean and the variance of the distance travelled per unit time are:

$$b = 1 - 2q, \quad c = 4q(1-q)$$

The results in Figure 4.2 are obtained from 10,000 simulation runs in MATLAB, and the pdf curves are estimated using the Gaussian kernel density estimator method [119]. One can notice that the simulation results are in good agreement with those predicted by the model with some discrepancies which are likely due to the differences in assumptions (regarding the distributions of timers and losses) between the model and the simulator. Note also that the diffusion model is expected to become more accurate as the distance between the source and the destination D becomes large.

4.3.1. The effect of a limited energy budget

Since energy consumption is an important issue in wireless networks, we apply the previous results to estimating the performance of a policy where each packet is imposed a limited energy budget of B units (e.g. micro-watts-secs). By calculating the ratio of the probability that a packet does reach its destination, to the probability that it does not, with an energy budget B, we can determine how the required value of B should be chosen for reliable network operation. In turn this would allow us to evaluate the total energy budget needed for a communication that includes a known number of packets. This is illustrated in Figures 4.3 and 4.4 where we plot the likelihood function

$$\eta(B) = \frac{\Pr[J < B]}{\Pr[J > B]} \tag{4.34}$$

against the budget B for different values of the average travel speed and loss rate. As one would expect, the figures show that as losses decrease or routing accuracy improves, the likelihood of reaching the destination node with a fixed energy budget B increases.

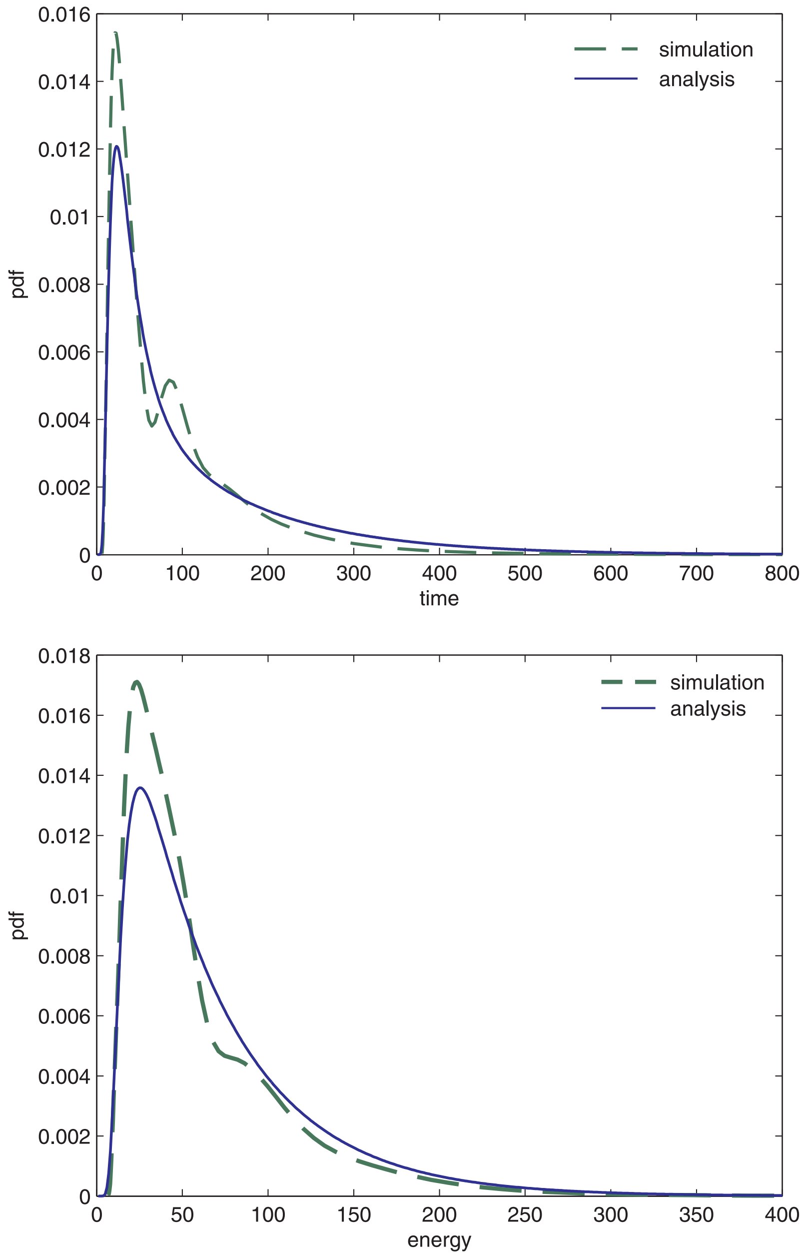


Figure 4.2.: Comparison of analytical predictions and simulations results for a regular topology with $D = 10$. Parameters of the simulation are: constant time-out $1/r = 60$ and retransmission delay $1/\mu = 1$, Bernoulli packet loss probability $p_\lambda = 0.01$ and probability of correct routing decision $q = 0.6$.

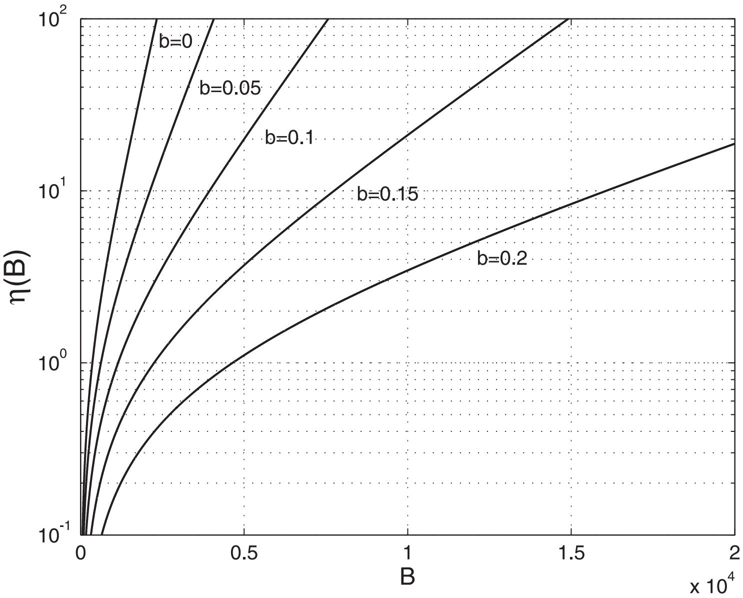


Figure 4.3.: $\eta(B)$ (logarithmic scale) versus the energy budget B for $\lambda = 0.05$, $c = 1$, $\mu = 0.1$, $r = 0.01$, $D = 10$ and different values of b.

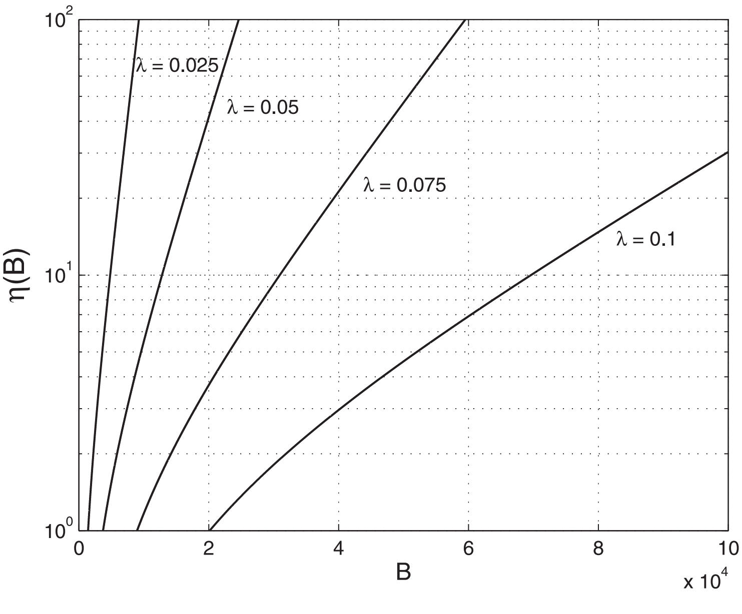


Figure 4.4.: $\eta(B)$ (logarithmic scale) versus the energy budget B for $b = -0.1$, $c = 1$, $\mu = 0.1$, $r = 0.025$, $D = 20$ and different values of λ.

4.3.2. The effect of packet redundancy

In Figure 4.5 we plot the pdf and the cdf of the decoding time for a message of size $k = 8$ and for different values of the number of transmitted packets $N > k$. We can observe that as N increases, the mean and the variance of the decoding time decrease as one would expect.

Figure 4.6 illustrates the asymptotic result of (4.32) showing that convergence is fast and that the error between the exact and asymptotic results is relatively small even for small values of N.

Next in Figures 4.7 and 4.8 we evaluate the delay performance of coding and replication under the same total number of transmitted packets. In particular, we compare the delay performance of sending N packets and waiting for k of them to arrive (i.e., $T_{k,N}$), and sending N/k duplicate packets and waiting for the first to arrive (i.e., $T_{1,N/k}$). This is illustrated in Figure 4.7 where we plot the quantile function that returns the smallest upper bound below which random draws of delay fall in $p \times 100\%$ of observations. More precisely, we plot the function $G_{i,j}^{-1}(p) = \inf\{t : G_{i,j}(t) \geq p\}$ against the probability p, where $i = k,\ j = N$ for coding and $i = 1,\ j = N/k$ for replication. The results indicate that in few cases, packet duplication achieves better performance while coding yields smaller delays most of the time. Figure 4.8 compares the pdf and the cdf for replication and coding with $k = 5$ and a total number of transmitted packets $N = 25$. In this case, the mean and the standard deviation of the packet delay are, respectively, 128.3 and 66.1 with coding and 120 and 132.4 with duplication. Hence, erasure coding is able to eliminate the long-tail observed in Figure 4.1, which makes it more appropriate for applications requiring reduced delay variations. This confirms earlier results obtained in [14] for exponential and Pareto distributions of the delivery delay.

Finally, in Figure 4.9 we examine how N should be selected in order to optimise both delay and energy utilisation. More specifically, we plot the locus of the average decoding time and the total average energy consumption, obtained from (4.33), when the number of transmitted packets N is varied between 9 and 20 for a block of $k = 8$ packets. The results indicate that as N increases the decoding time drops rapidly while the total energy expenditure increases at a much lower rate. For instance, sending double the number of packets (i.e., $N = 16$) increases energy cost by only 12% while it reduces decoding time by 72% as compared to sending only one redundant packet ($N = 9$). Thus sending out multiple packets can reduce delivery time significantly without compromising energy efficiency provided that a feedback mechanism is available at the destination node which can stop further packet transmissions in the network after k packets have been received.

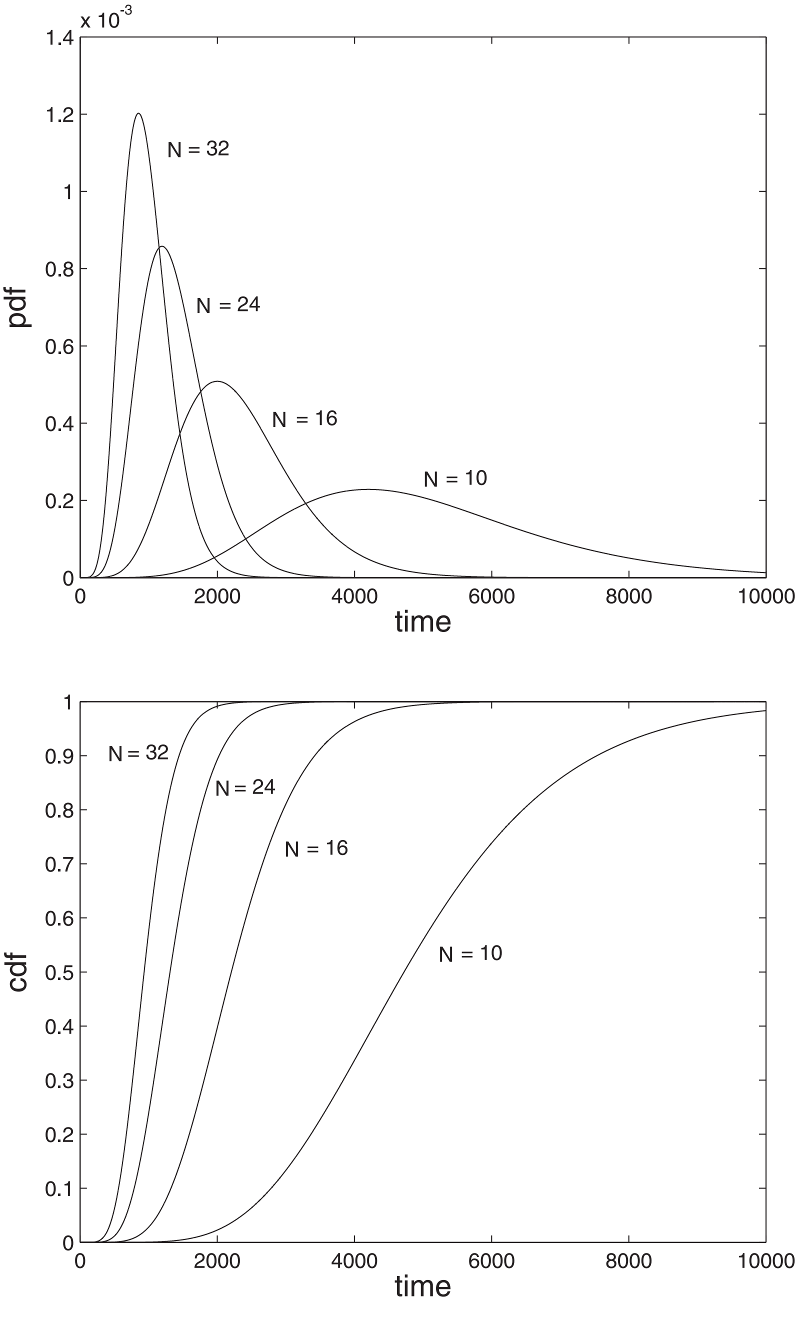


Figure 4.5.: (a) pdf and (b) cdf of the decoding time for a message of size $k = 8$ and for different number of transmitted packets N with $\lambda = 0.05$, $r = 0.04$, $\mu = 0.1$, $b = 0.15$, $c = 1.5$ and $D = 10$.

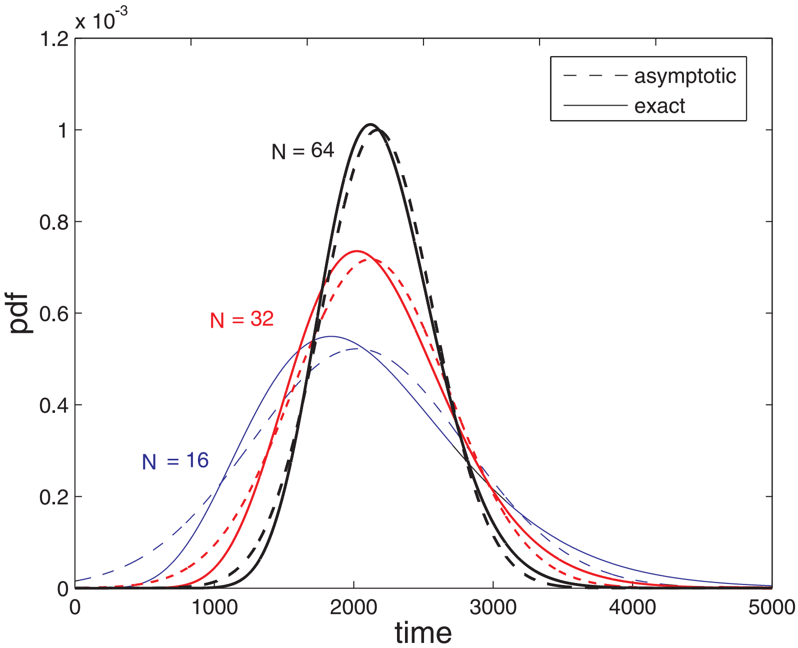


Figure 4.6.: Exact and asymptotic results for the pdf of the decoding delay when the ratio of the message size to the number of sent packets $k/N = 0.5$ with $b = -0.2, c = 0.5, \lambda = 0.1, r = 0.02, \mu = 0.1$ and $D = 10$.

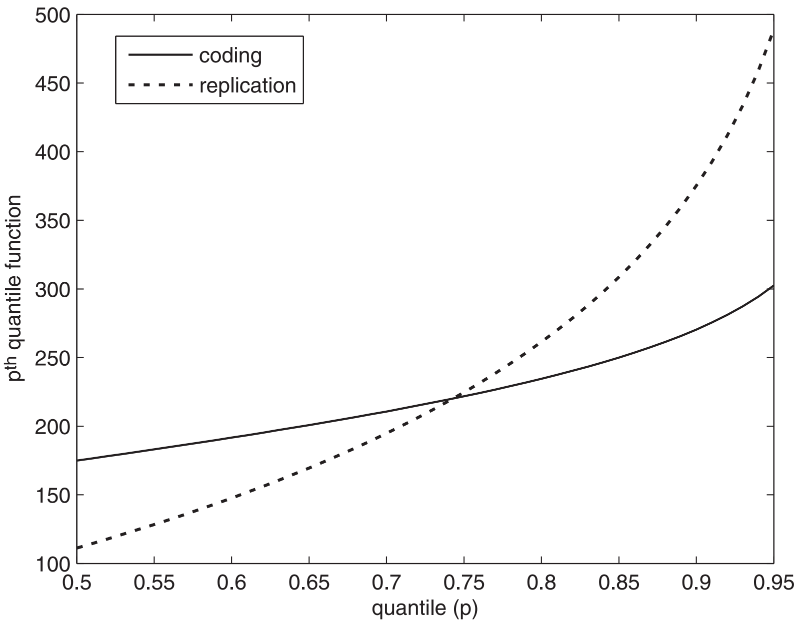


Figure 4.7.: Comparison of the delay performance of sending $N = 32$ packets and waiting for $k = 8$ to arrive (coding) and sending $N/k = 4$ packets and waiting for 1 to arrive (replication) with $b = 0, c = 1, \lambda = 0.01, r = 0.025, \mu = 0.1$ and $D = 10$.

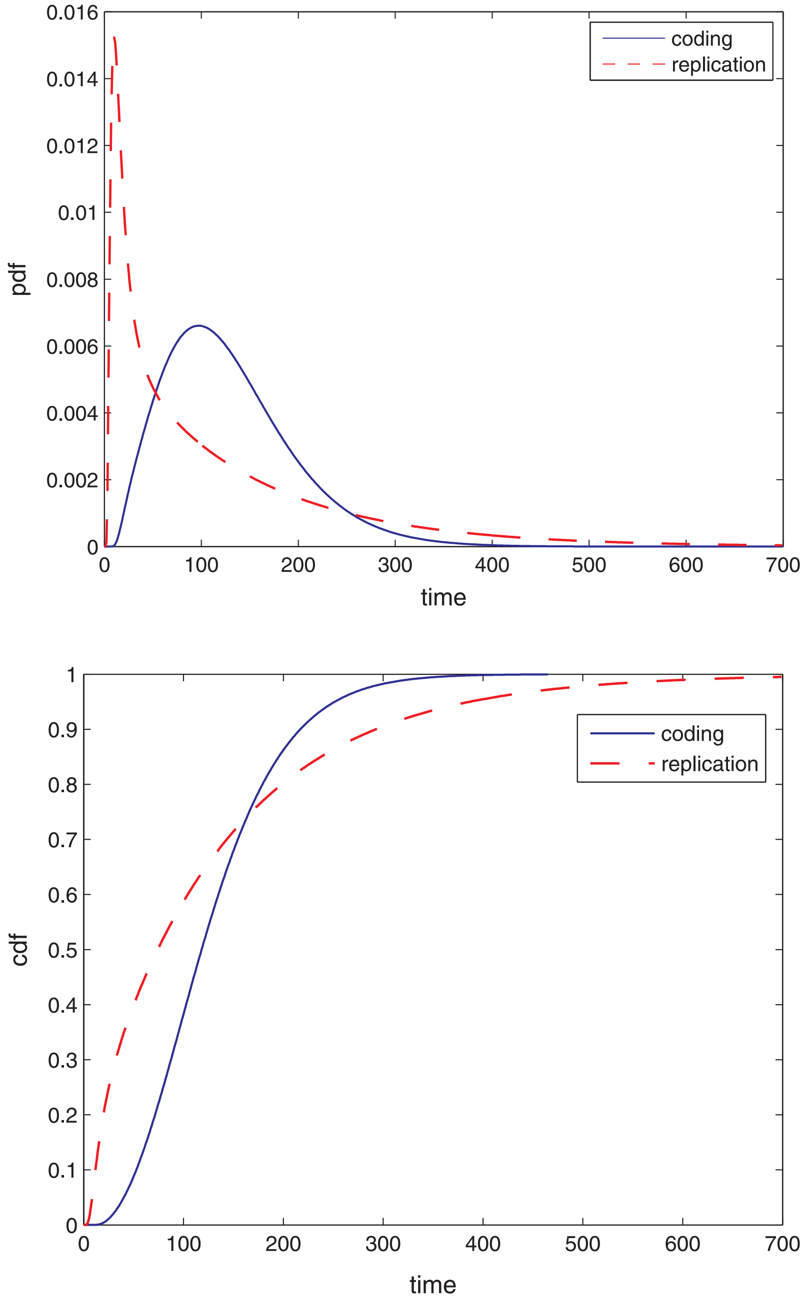


Figure 4.8.: Comparison of the pdf and cdf of the delay for sending $N = 25$ coded packets and waiting for any 5 packets to arrive and sending $N/k = 5$ duplicate packets for any single packet. Parameters of the diffusion model are $b = 0.2, c = 2.5, \lambda = 0.01, r = 0.025, \mu = 0.1$ and $D = 10$.

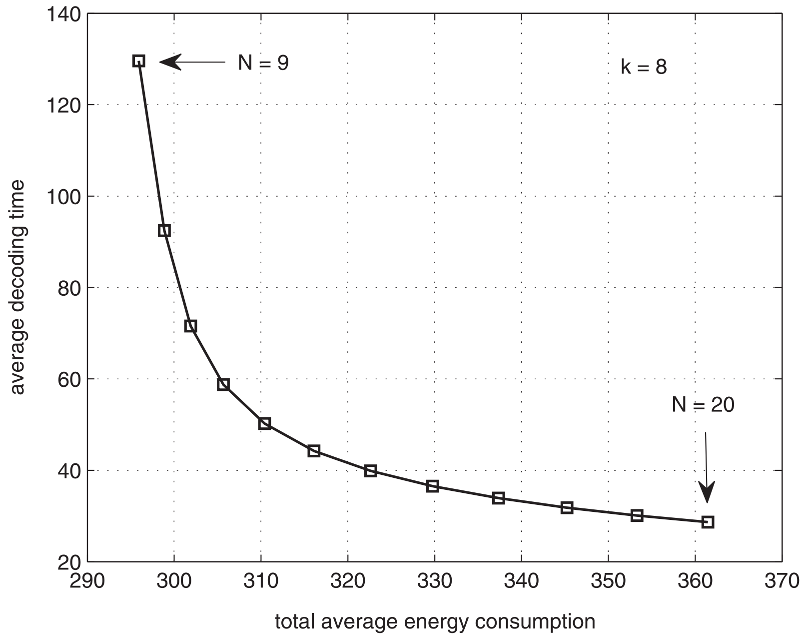


Figure 4.9.: Locus of average decoding time and total average energy consumption as a result of varying N for $k = 8$, $b = -0.35$, $c = 1$, $\mu = 0.1$, $\lambda = 0.01$, $r = 0.015$ and $D = 10$.

4.4. Summary

In this chapter we derived the Laplace transform of the probability density of total packet forwarding delay and energy expenditure in a homogeneous wireless network. The analysis was extended to evaluate the case where the source node sends duplicate or coded packets in order to mitigate the effects of packet loss and uncertainty in routing information. Numerical examples suggest that while erasure coding may result in higher overall packet travel delay and energy consumption on average, it reduces delay variations significantly and thus reduces the uncertainty in packet delivery times. The trade-off in erasure coding between energy and delay as a result of varying the number of transmitted packets has been examined, and we have observed desirable operating areas where delivery time is reduced significantly at the cost of a small increase in energy utilisation.

5. Queueing models for network coding

The average time and energy required for a packet to travel from a source node to a destination node in a large multi-hop non-homogeneous network with imprecise routing information and packet losses have been derived in Chapter 3. In Chapter 4 we obtained the distribution of these performance measures in a homogeneous medium, and examined the merits and limitations of using k-out-of-N coding techniques at the source nodes, in terms of reliable packet delivery times and energy efficiency. So far we have assumed that intermediate nodes do not manipulate the contents of packets and simply relay them, which is not efficient for utilising network resources [4]. Thus the present chapter proposes models that can assist in the analysis and optimisation of coding at the interior nodes in the network.

In particular, we consider a store and forward packet network in which NC is being used to co-encode packets from distinct flows. In such a system both the encoding and decoding process may introduce additional delays, and we develop analytical models to evaluate the resulting performance. The approach we adopt is based on the analysis of queueing systems with specific service processes that capture the effect of NC, and we analyse single and multi stage queueing models for a router that carries out NC, in addition to its standard packet routing function. The approach is extended to the study of multiple hops, which leads to an interesting constrained optimisation problem that characterises the optimal time that packets should be held back in a router so that the total packet end-to-end delay, including encoding, queueing and decoding at the output is minimised. We assume certain basic network structures, where inter-flow coding can be useful, and which nevertheless can represent small fragments of larger arbitrary topologies such as the ones studied in previous chapters. Trade-offs between delay and bandwidth or energy are also investigated, and the results indicate that NC can offer significant gains in all respects provided that decisions to code packets from distinct flows is made a function of network traffic conditions. The accuracy of the analytical solutions is verified through comparison with discrete event simulations with ns-2 [15].

The remainder of the chapter is organised as follows. Section 5.1 presents a single server queueing model for a cross-layer design of NC which coordinates packet coding and transmission at the intermediate node. The benefit of the abstract model is that, even under non-Markovian assumptions and for any number of flows, it can be quite accurately

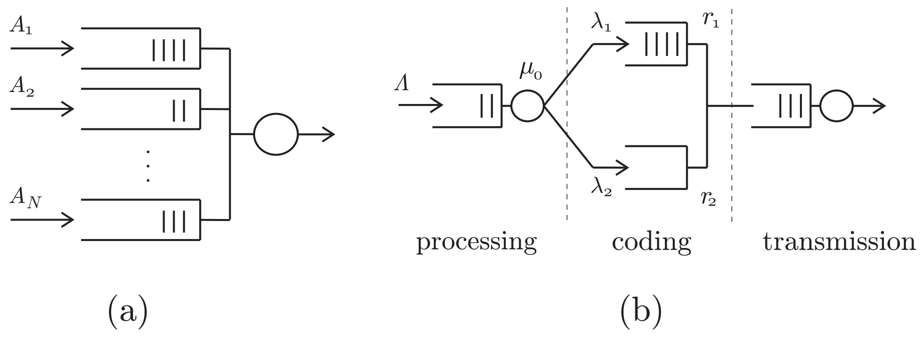


Figure 5.1.: (a) Single and (b) multi stage queueing models for an encoding node.

analysed using a simple decoupling approximation. In Section 5.2 we propose a multistage queueing model which introduces processing and coding queues prior to the transmission queue in order to analyse the performance of NC when these distinct functionalities are decoupled; the model also incorporates a time-out mechanism to modify coding opportunities. A queueing network model is then presented along with a heuristic for optimising the end-to-end delay. Finally, Section 5.3 presents a summary of the results.

5.1. Single stage queueing model

In this section we analyse the performance of an intermediate node that receives and encodes an arbitrary number of independent sessions with general arrival and packet size distributions. The encoding node is represented by a single-server queue and the effect of NC on the service process is captured through the potential increase in transmission times resulting from combining packets of random lengths. This model corresponds to a cross-layer NC implementation which coordinates packet coding and transmission; it does not represent the case where coding is performed independently of the hardware that sends packets out over links, which is addressed in the subsequent section. Note that this abstract queueing representation follows the classical approach [16, 17] of modelling data networks as networks of transmission queues.

Consider a network node which receives N distinct independent flows of packets with general inter-arrival time distribution $A_i(x) = \Pr[A_i \leq x]$ for the i-th flow, which queue up in distinct buffers of unlimited capacity as depicted in Figure 5.1(a). Denote by λ_i the average arrival rate of the i-th flow $\lambda_i = E[A_i]^{-1}$. Assume that packet lengths in each stream are independent random variables and that they are mutually independent between flows, with general distribution $L(x) = \Pr[L \leq x]$ where L is the random variable representing packet length. We assume that the transmission time S is directly proportional to packet length, that is $S = L/c$ where c is the capacity of the transmission link measured in bit/s, and define $S(x) = \Pr[S \leq x]$.

As discussed in Section 2.2, synchronous NC (SNC) incurs very large delay and loss penalties particularly when the network is lightly loaded; therefore we will consider an ONC scheme which encodes packets from *distinct* flows with any packets present, except for packets belonging to the same flow. Thus after the encoding node forwards a packet, the next packet forwarded will simply be the encoded version of the packets from distinct flows that are present in the queues. If only one of the flows has a packet present, then just that one un-encoded packet will be forwarded, while if more than one flow has at least one packet present in their queues, then the encoded packet will include the head-of-line (HOL) packet from each of those flows.

When a coding operation that involves $1 < n \leq N$ packets starts, the server will pack the shorter packets with *zero*-bits to reach the length of the longest packet, and encode the resulting packet bit by bit, so that its length will be equal to the largest of the n packet lengths. The transmission time of a packet is then proportional to the length of the largest co-encoded packet and its distribution is given by $S(x)^n$.

5.1.1. The queueing behaviour of opportunistic coding

In order to have more understanding of the queueing process in ONC, consider a node with $N = 2$ flows, and define the following $\Re^+$ valued variables for the i-th flow:

$a_{i,n}$ Arrival instant of the n-th packet to the encoding node.

$S_{i,n}$ Service time of the n-th packet.

We can also define the following quantities which depend on the above variables:

$A_{i,n}$ Inter-arrival time between the n-th and the $(n-1)$-th packets, i.e., $A_{i,n} = a_{i,n} - a_{i,n-1}$.

$d_{i,n}$ Departure instant of the n-th packet from the encoding node.

We assume that as $n \to \infty$ the sequences $\{A_{i,n}\}$ and $\{S_{i,n}\}$ converge in law to the random variables A_i and S, respectively.

If we assume the system to be initially empty and adopt the convention that $a_{i,0} = a_{j,0} = 0$ where $j = 3 - i$, then we can write Lindley-type [120] recursive equations for the sequence $\{d_{i,n}\}_{n \geq 1}$ as follows:

1) $d_{i,n} = d_{i,n-1} + S_{i,n}$ if $a_{i,n} \leq d_{i,n-1}$ and (a) there exists m such that $d_{i,n-1} \geq d_{j,m-1}$, (b) $a_{j,m} > d_{i,n-1}$.

2) $d_{i,n} = d_{i,n-1} + \max[S_{i,n}, S_{j,m}]$ if $a_{i,n} \leq d_{i,n-1}$ and (a) there exists m such that $d_{i,n-1} = d_{j,m-1}$, (b) $a_{j,m} \leq d_{j,m-1}$.

3) $d_{i,n} = d_{i,n-1} + \max[S_{i,n}, S_{j,m}]$ if $a_{i,n} \leq d_{i,n-1}$ and (a) there exists m such that $d_{i,n-1} > d_{j,m-1}$, (b) $\max[a_{i,n-1}, d_{i,n-2}, d_{j,m-1}] < a_{j,m} \leq d_{i,n-1}$.

4) $d_{i,n} = a_{i,n} + S_{i,n}$ if $a_{i,n} > d_{i,n-1}$ and (a) there exists m such that $d_{j,m-1} \leq a_{i,n}$, (b) $a_{j,m} > a_{i,n}$.

5) $d_{i,n} = a_{i,n} + V_{i,n} + S_{i,n}$ if $a_{i,n} > d_{i,n-1}$ and (a) there exists m such that $\max[a_{j,m-1}, d_{j,m-2}] < a_{i,n} < d_{j,m-1}$, (b) $V_{i,n} = d_{j,m-1} - a_{i,n}$, (c) $a_{j,m} > d_{j,m-1}$.

6) $d_{i,n} = a_{i,n} + V_{i,n} + \max[S_{i,n}, S_{j,m}]$ if $a_{i,n} > d_{i,n-1}$ and (a) there exists m such that $\max[a_{j,m-1}, d_{j,m-2}] < a_{i,n} < d_{j,m-1}$, (b) $V_{i,n} = d_{j,m-1} - a_{i,n}$, (c) $a_{j,m} \leq d_{j,m-1}$.

where the variable $V_{i,n}$, appearing in the last two cases, represents the additional delay a packet experiences when it arrives at an empty queue and finds the server busy transmitting a packet from the other queue. In such instances, the packet must wait until the current service has finished, after which it will be either transmitted without coding if the other buffer becomes empty (case 5) or it will be combined with the HOL packet from the other flow and transmitted (case 6).

5.1.2. A decoupling approximation

The recursive equations above indicate that, even for the simplest case of 2 flows, exact analysis of ONC is extremely difficult due to the facts that (a) transmission times depend on the number of combined packets, and (b) it is difficult to characterise exactly the distribution of the random variable $V_i = \lim_{n\to\infty} V_{i,n}$. Thus we propose a solution based on the "decoupling approximation" along the lines of [121]. In particular, we study a queue, say the i-th, in isolation from the others and consider that the queues interact with each other via the steady-state probabilities, which we call q_i for the i-th queue, that the i-th flow is not involved in a coding operation. In this case this may occur either because that queue is idle (no customers), or because the queue is busy but there is another currently ongoing encoding and transmission which started while the i-th queue was empty.

The approximation we propose is based on constructing an "equivalent" $G/G/1$ model for any of the N individual queues, having the "server with vacations" property [122,123]. Note that in this case the assumption of finite buffers is not needed because (contrary to the SNC case) the system with unlimited buffer size is not always unstable.

The server with vacation is a queueing system in which, after each service ends, if the queue is empty then the server will "go off" for a vacation time V, and the process repeats itself if the server finds the queue empty at the end of the vacation time. Service starts again when, at the end of a vacation time, there is at least one customer in queue. Such models usually assume that each successive vacation time is an iid random variable. The key result about a queue with vacations can be summarised as follows.

Result 5.1 (Gelenbe, Iasnogorodski [123]) *Let W be the random variable representing the steady-state distribution of the customer waiting time for a queue (with unlimited buffer size) and a server with vacations, for a single server queue with service time S and vacation time V, which are assumed to be mutually independent random variables, while both the successive service times and the successive vacation times are sequences of iid random variables. We assume that the inter-arrival times of customers to the queue are also iid (but the arrival process need not be Poisson). Let W_0 be the waiting time for exactly the same queue, but* without *vacations (i.e. when $V = 0$). Then the following equality holds in distribution:*

$$W = W_0 + \hat{V} \tag{5.1}$$

where $\hat{V}$ is the forward recurrence (residual) vacation time whose distribution is given by [122]:

$$\hat{V}(x) = \frac{1}{E[V]} \int_0^x [1 - V(y)]\ dy \tag{5.2}$$

Thus the result summarised in (5.1) allows us to map all properties of interest of a queue with vacations in steady state to those of a system without vacations using the probability distribution of the vacation time V. In particular we can see that only W_0 depends on the arrival process, and therefore the stability condition for the queue with vacations is *not* affected by the vacation time distribution, and is identical to the stability conditions for the corresponding *ordinary* queue.

Now turning back to the model for ONC, note that if at the end of a service time the i-th queue is *not empty*, then the subsequent service time distribution $S_i(x)$ will be the maximum of the service times for the set of non-empty queues including the i-th queue. Thus, for $Z(i) = \{1,\ ...\ i-1, i+1,\ ...\ N\}$, the resulting approximation for the service time distribution after a departure that *does not* leave an empty queue at i will be:

$$S_i(x) = S(x) \sum_{Z \subseteq Z(i)} S(x)^{|Z|} \prod_{j \in Z} [1 - q_j] \prod_{j \notin Z} q_j \tag{5.3}$$

On the other hand if the i-th queue is empty after a service time ends, then when the next arrival to that queue occurs, the arriving packet will have to wait for its service until after any currently ongoing service involving the *other* queues ends.

Thus after the i-th queue becomes empty at the end of a service time, a sequence of vacation times involving the services at *other* queues will take place, and some of these other services will possibly be of zero duration if the queues are empty. These vacation times have a probability distribution similar to $S_i(x)$, *except* that the i-th queue is not involved. We will denote the vacation time distribution for the i-th queue by $V_i(x)$, where:

$$V_i(x) = \sum_{Z \subseteq Z(i)} S(x)^{|Z|} \prod_{j \in z} [1 - q_j] \prod_{j \notin Z} q_j \tag{5.4}$$

We can now use the formula for the probability q_i that the i-th queue is empty or that it is busy but that its equivalent server is idle due to a vacation time, from the corresponding result for the model with vacations, where $S_i(x)$ and $V_i(x)$ are, respectively, the service time and vacation time distributions:

$$q_i = 1 - \lambda_i E[S_i] \tag{5.5}$$

For Poisson arrivals, the probability of the i-th queue being empty at the end of a service or a vacation time has been obtained [122] and is given by:

$$q_i = \frac{1 - \lambda_i E[S_i]}{1 + \lambda_i (E[V_i] - E[S_i])} \tag{5.6}$$

However, in the numerical results we have found that the first expression for q_i gives more accurate results even though the latter captures the dynamics of the system more accurately.

In [124], we consider the use of ONC in scenarios where combining packets from all flows may render some of the packets undecodable at the output. We propose a combined scheduling-coding scheme which restricts NC within disjoint subsets of the traffic streams, and we extend the decoupling approximation to the case where coding subsets are served in a round-robin (RR) manner.

5.1.3. Performance evaluation

The performance measures of interest for ONC include the average response time, throughput, bandwidth utilisation and stability region which are considered below.

Response time

When the arrival processes are Poisson, we apply the well known *Pollaczek-Khinchin* formula [27] separately to each queue in order to obtain the mean waiting times. Then, utilising the decomposition property of vacation queues (5.1), the mean response time in queue i becomes:

$$E[R_i] = \frac{\lambda_i E[S_i^2]}{2(1 - \lambda_i E[S_i])} + E[S_i] + E[\hat{V}_i] \tag{5.7}$$

For general arrival processes, we use a diffusion approximation [8, 9, 125] to solve the equivalent $G/G/1$ systems. Figure 5.2 presents numerical results for the mean response

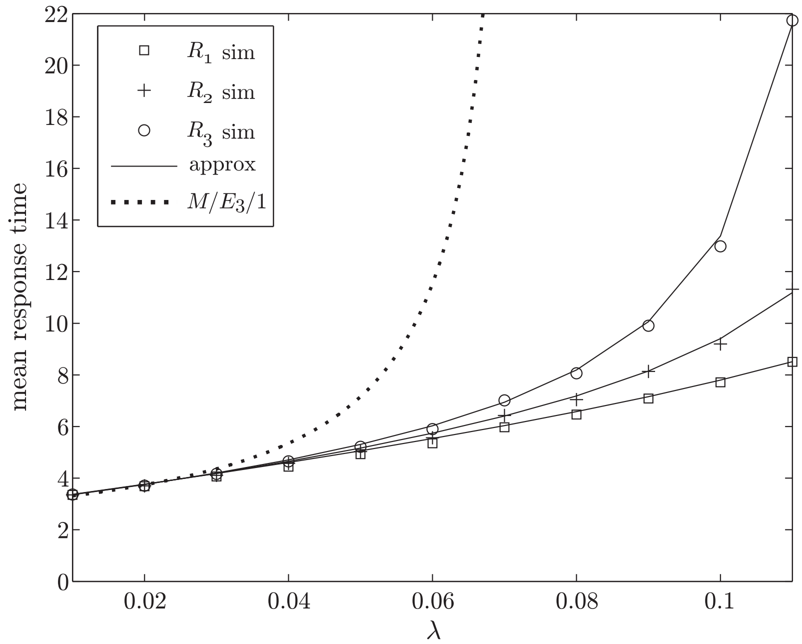


Figure 5.2.: Mean response times for ONC and plain forwarding of $N = 3$ Poisson traffic streams of rates $\underline{\lambda} = (1 \quad 1.5 \quad 2)\ \lambda$ and Erlang-3 packet size distribution with $E[S] = 3$.

times for three asymmetric Poisson flows with Erlang-4 packet size distribution. We find that the approximation yields very accurate results, which in most instances remain within 2% of the simulation results. Figure 5.3 presents results for four symmetric flows with Erlang-2 inter-arrival and service times distributions. In this case, the diffusion approximation introduces additional errors in computations as can be observed in the figure. However, the error between the analytical and simulation results remains within 10%. Overall the quality of the approximation appears to be relatively insensitive to the number of encoded flows and the traffic load. The figures also compare the average packet delay of ONC with a peer non-coding scheme in which packets from all incoming flows are stored into a single buffer and transmitted in a first-come-first-served (FCFS) order; as one would expect, ONC offers significant delay improvement.

Throughput

In order to estimate the average output *packet rate*, we first note that the notion of vacations exists only when we consider a single buffer in isolation but if we look at the system as a whole, the server will never be idle while any of the buffers is non-empty. Thus computing the packet throughput consists in summing the steady state probabilities over all possible subsets of non-empty queues $Z \subseteq \{1, ..., N\} : |Z| \geq 1$ each weighted by

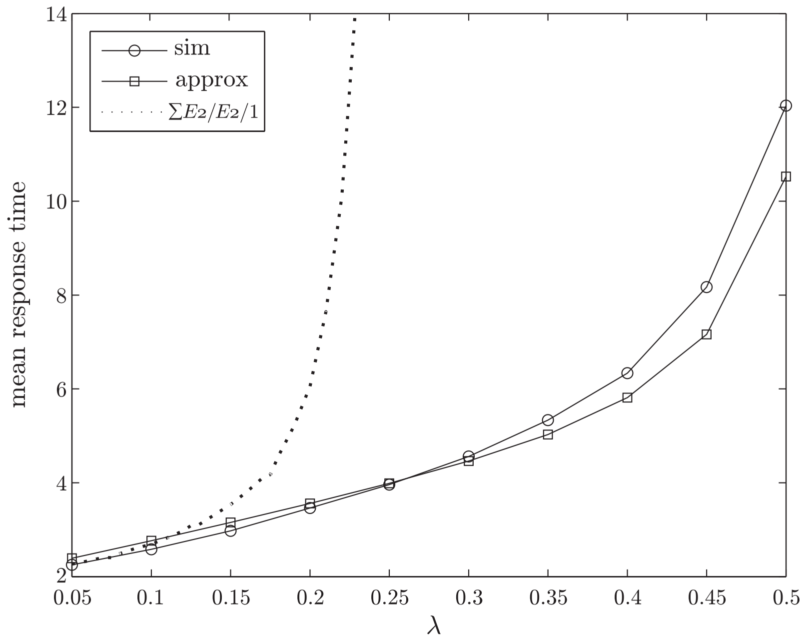


Figure 5.3.: Mean response times for ONC and plain forwarding of $N = 4$ balanced traffic streams with Erlang-2 inter-arrival times each with rate λ and Erlang-2 packet size distribution with $E[S] = 2$.

the corresponding average transmission rate. Using the decoupling approach, this can be approximated by:

$$\varphi = \sum_{Z \subseteq \{1,\ldots,N\}:|Z|\geq 1} \prod_{j \in Z} [1 - q_j] \prod_{j \notin Z} q_j \Big(\int_0^\infty x \; dS(x)^{|Z|} \Big)^{-1} \tag{5.8}$$

Alternatively the output rate can be obtained from the inverse of the time interval between two successive packet departures which can be approximated by:

$$\varphi^{-1} = \prod_{j=1}^{N} q_j \; E[\hat{A} + S] \; + \sum_{Z \subseteq \{1,\ldots,N\}:|Z|\geq 1} \prod_{j \in Z} [1 - q_j] \prod_{j \notin Z} q_j \int_0^\infty x \; dS(x)^{|Z|} \tag{5.9}$$

The first term in (5.9) represents the case where all queues are empty at the end of a packet transmission so that the time until the next departure will include the residual inter-arrival time until the first packet arrives $\hat{A}$, followed by its transmission time S. Let $\hat{A}_i$ denote the forward recurrence time of the inter-arrival time to the i-th queue; we approximate the distribution of $\hat{A}$ by:

$$\hat{A}(x) = \Pr[\min_{1\leq i\leq N} \hat{A}_i \leq x] = 1 - \prod_{j=1}^{N}\left(1 - \lambda_j \int_0^x [1 - A_j(y)]\ dy\right)$$

The second term in (5.9) denotes the case where there are $|Z| \geq 1$ non-empty queues at the end of a transmission epoch, so that the time until next packet departure will be distributed according to $S(x)^{|Z|}$.

Coding gain and opportunities

We define *coding gain* for stable traffic load (i.e., $q_i > 0,\ \forall i$) as the ratio of the input bit rate to the output bit rate:

$$\eta = \frac{\sum_{i=1}^{N} \lambda_i\ E[L]}{\left(1 - \prod_{i=1}^{N} q_i\right) c} = \frac{\sum_{i=1}^{N} \lambda_i\ E[S]}{1 - \prod_{i=1}^{N} q_i} \in [1, N] \tag{5.10}$$

which is a measure of transmission costs along the output link. One may also be interested in estimating *coding opportunities* or the average number of packets that are combined in each transmission:

$$\zeta = \frac{\sum_{i=1}^{N} \lambda_i}{\varphi} \tag{5.11}$$

Note that $\eta \leq \zeta$ with the equality holds if and only if the packet size is constant; this follows from (5.8) by writing

$$\begin{aligned}\varphi \ &\leq \ \sum_{\substack{Z\subseteq\{1,\ldots,N\}\\ |Z|\geq 1}} \prod_{j\in Z}[1 - q_j] \prod_{j\notin Z} q_j E[S]^{-1} \\ &= \ (1 - \prod_{i=1}^{N} q_i)E[S]^{-1} = \frac{\sum_{i=1}^{N} \lambda_i}{\eta}\end{aligned}$$

Hence the fact that the encoded packet size is dominated by the longest co-encoded packet reduces the gain from NC.

Figure 5.4(a) compares analytical and simulation results for the packet throughput of ONC of three symmetric Poisson flows using different packet size distributions with the same mean. The analytical results are obtained using (5.9) for exponential and Erlang-3 distributions; (5.9) for constant length; and the average of the two approximations for Erlang-5 distribution. In general, we have observed that (5.8) tends to underestimate the output rate whereas (5.9) overestimates it. The error between the analytical and simulation results, however, is less than 5% in most of the cases tested. Interestingly, the figure shows, for non-deterministic packet size, a drop in the output packet rate as the input rates approach saturation. This suggests that when the node is congested, any slight increase in the input rate will significantly increase coding opportunities and as a result fewer but longer packets will be transmitted. Figure 5.4(b) depicts the coding gain η showing that the lower the variance of the packet size distribution, the higher the coding gain and the maximum stable traffic load. Indeed, although constant packet length yields higher output packet rate and thus fewer coding opportunities ζ, it still provides the best utilisation of the output link.

Stability region

In order to illustrate how the decoupling approximation can be used to characterise the stability conditions for ONC, we consider a node with $N = 2$ flows with arrival rate λ_i and service rate μ_i for the i-th flow. If we call $\nu = E[\max(S_1, S_2)]^{-1}$ then from (5.5) the probability q_i is coupled with q_j, where $j = 3 - i$, by the relation:

$$1 - q_i = [1 - q_j]\frac{\lambda_i}{\nu} + q_j\frac{\lambda_i}{\mu_i}$$

Solving the two equations for $i = 1, 2$ simultaneously yields:

$$q_i = 1 - \lambda_i\nu \; \frac{\nu\mu_j + \lambda_j(\mu_i - \nu)}{\nu^2\mu_i\mu_j - \lambda_i\lambda_j(\mu_i - \nu)(\mu_j - \nu)}$$

and the stability conditions become:

$$\begin{aligned} \lambda_i &< \frac{\mu_i\nu^2}{\nu^2 + \lambda_j(\mu_i - \nu)}, && 0 \leq \lambda_j \leq \nu \\ \lambda_i &< \frac{\nu^2}{\lambda_j}\,\frac{\mu_j - \lambda_j}{\mu_j - \nu}, && \nu \leq \lambda_j < \mu_j \end{aligned} \tag{5.12}$$

On the other hand, under the store and forward paradigm the node acts as a FCFS queue with two classes of packets; thus its stability region is given by:

$$\frac{\lambda_1}{\mu_1} + \frac{\lambda_2}{\mu_2} < 1 \tag{5.13}$$

Figure 5.5 compares the stability regions (5.12) and (5.13) using different packet size distributions with average rates $\mu_1 = 0.8$ and $\mu_2 = 0.7$.

5.2. Multistage queueing model

In this section we propose a queueing model for a NC implementation that preserves the decoupling between packet processing and transmission which exists in current Internet

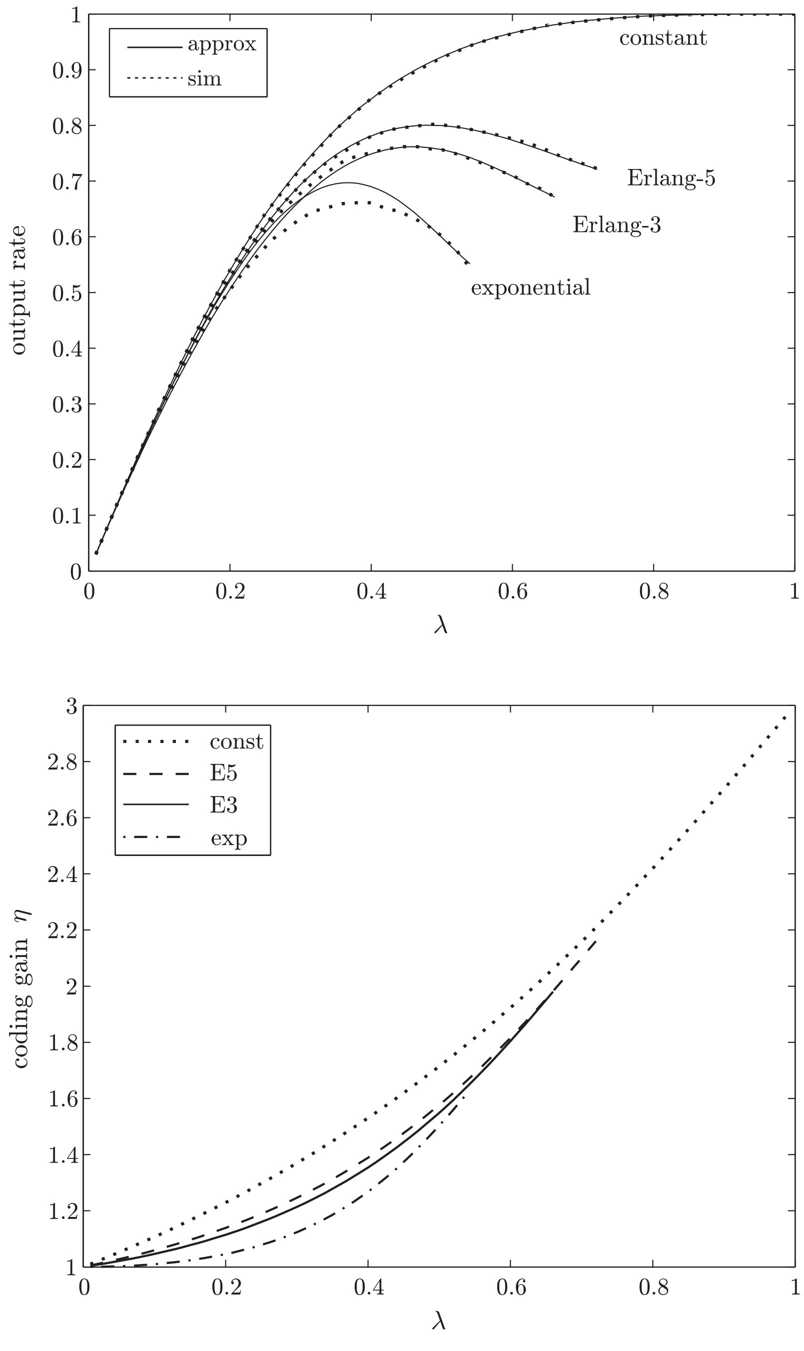


Figure 5.4.: (a) Output packet rate and (b) coding gain η for ONC of $N = 3$ balanced Poisson flows each of average rate $\lambda < \left(\int_0^\infty x dS(x)^N\right)^{-1}$, for different packet size distributions with $E[S] = 1$.

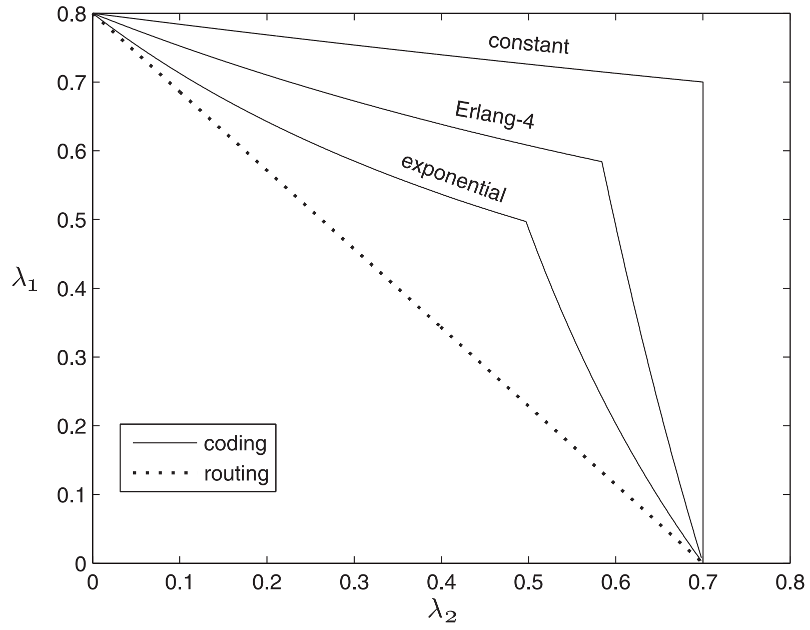


Figure 5.5.: Stability regions for routing and ONC with two flows with average service rates $\mu_1 = 0.8$ and $\mu_2 = 0.7$.

routers. These routers consist of four main components: a processing module, memory, an internal interconnection unit and several network interfaces to the attached network [126]. Typically, packets are received at an inbound network interface, processed by the processing component and finally forwarded through the internal bus to the outbound interface that transmits them on the next hop towards their destination. NC would add an additional coding module that sits between the processing and transmission components. This module, which can be implemented in either software or hardware, should be seamless to flows that are not participating in NC so that their packets are forwarded directly from the processing module to the network interfaces without further delay.

To represent such a NC router, we propose a multistage queueing model which introduces processing and coding queues prior to the transmission queue. Since packets will arrive individually at the coding queue, the router can use timers in order to modify coding opportunities. We derive analytical expressions for the throughput, coding gain, energy consumption and response time in terms of the traffic rates, packet size distribution and timers durations for the case of two Poisson traffic streams. We then incorporate the model into a networking setting and present a simple heuristic for optimising the end-to-end delay performance, including packet assembly and decoding at the output.

5.2.1. Queueing analysis

In order to evaluate the performance of the above NC router architecture, we consider a network node that receives and encodes two distinct independent Poisson flows of packets A_i with rate λ_i for the i-th flow. Denote by A the total arrival process to the node which is also a Poisson process with rate $\Lambda = \lambda_1 + \lambda_2$, and let $\rho_i = \lambda_i/\Lambda$. The node's buffers are assumed to be of unlimited capacity so that loss of packets due to buffer overflow cannot occur.

We assume that packets arriving at the node are stored into an input queue where their headers are processed according to an exponential service time, with parameter $\mu_0 >> \Lambda$, which is independent of the packet size L. Thus the processing module is modelled as an $M/M/1$ system with negligible service time. The received packets are then forwarded to the coding module which consists of two buffers, one for each packet class, of unlimited capacity. The policy employed by the coding component to process the packets from the two buffers is as follows:

- When a packet reaches the head of buffer i, if the other buffer $j = 3 - i$ is not empty, then this packet is immediately XOR-ed with the head of the line packet from buffer j and the coded version is placed in the transmission queue. We assume that the XOR operation is instantaneous so that no further delay is incurred after the two packets are matched. This is a fairly reasonable assumption given the current processing power.

- When a packet arrives at the head of buffer i and the buffer holding the traffic from class j is empty, then the packet waits for an exponential time-out period with parameter r_i. If a packet arrives at queue j before the timer elapses, then the two packets will be coded together and the coded version will be placed in the transmission queue. If, however, the timer expires before a packet arrives at buffer j, then the head of the line packet from buffer i will be forwarded to the transmission queue without coding.

The queueing network model for the encoding node is depicted in Figure 5.1(b). The question that arises from the above coding policy is: how long should the encoding node wait before going ahead and sending the information that it has uncoded? Short waiting times imply few coding opportunities and therefore inefficient network operation while long waiting times imply added delay which eventually overtakes the original benefit of coding.

Coding queues

Denote by $\mathcal{Q}_i(t)$ the number of packets waiting in buffer $i \in \{1,2\}$ at time instant t. Note that due to the coding policy described previously, at most one of the two coding queues can be non-empty. Thus, we can represent the state of the two queues at time instant t by a single number $\mathcal{Q}(t) = \mathcal{Q}_1(t) - \mathcal{Q}_2(t) \in [-\infty, \infty]$. If the i-th coding queue is non-empty, then the time until the next packet departure is an exponential random variable U_i with parameter $\mu_i = r_i + \lambda_j$. The stationary probability $p_n = \lim_{t\to\infty} \Pr[\mathcal{Q}(t) = n]$ can be shown to have the following form:

$$p_n = \begin{cases} p_0 \left(\frac{\lambda_1}{\mu_1}\right)^n & , \; n > 0 \\ p_0 \left(\frac{\lambda_2}{\mu_2}\right)^{-n} & , \; n < 0 \end{cases} \tag{5.14}$$

where p_0 is obtained from the normalisation condition $\sum_{n=-\infty}^{\infty} p_n = 1$, yielding:

$$p_0 = \left[1 + \frac{\lambda_1}{\mu_1 - \lambda_1} + \frac{\lambda_2}{\mu_2 - \lambda_2}\right]^{-1} \tag{5.15}$$

Consider the queue length process at time instants when packets depart from the coding stage, and denote by π_n the stationary probability associated with the queue length at those instants which, if exist, must satisfy:

$$\begin{aligned} \pi_0 &= [\pi_1 + \pi_0\rho_1]\,\alpha_{1,0} + [\pi_{-1} + \pi_0\rho_2]\,\alpha_{2,0} \\ \pi_n &= \sum_{k=0}^{n} \pi_{n-k+1}\alpha_{1,k} + \pi_0\rho_1\alpha_{1,n}, \quad n > 0 \\ \pi_n &= \sum_{k=0}^{-n} \pi_{n+k-1}\alpha_{2,k} + \pi_0\rho_2\alpha_{2,-n}, \; n < 0 \end{aligned} \tag{5.16}$$

where $\alpha_{i,k}$ denotes the probability of k arrivals to queue i during a service time at the queue:

$$\alpha_{i,k} = \int_0^\infty \frac{(\lambda_i t)^k e^{-\lambda_i t}}{k!}\, \mu_i e^{-\mu_i t} dt = \frac{\mu_i}{\lambda_i + \mu_i}\left(\frac{\lambda_i}{\lambda_i + \mu_i}\right)^k \tag{5.17}$$

yielding:

$$\pi_n = \begin{cases} \pi_0\rho_1 \left(\frac{\lambda_1}{\mu_1}\right)^n & , n > 0 \\ \pi_0\rho_2 \left(\frac{\lambda_2}{\mu_2}\right)^{-n} & , n < 0 \end{cases} \tag{5.18}$$

and

$$\pi_0 = \left[1 + \rho_1\frac{\lambda_1}{\mu_1 - \lambda_1} + \rho_2\frac{\lambda_2}{\mu_2 - \lambda_2}\right]^{-1} \tag{5.19}$$

The distribution of the response time for coding queue $i \in \{1, 2\}$ can be obtained as follows:

$$T_i(x) = \sum_{n=\mp 1}^{\mp\infty} p_n + \int_0^x \sum_{n=0}^{\pm\infty} p_n \frac{\mu_i^{n+1} y^n e^{-\mu_i y}}{n!}\, dy = 1 - p_0 \frac{\mu_i}{\mu_i - \lambda_i}\, e^{-(\mu_i - \lambda_i)x} \tag{5.20}$$

The first term represents the case where a packet arriving at the i-th queue finds $n \geq 1$ packets waiting in the other buffer so that it does not experience any delay and is immediately encoded and forwarded to the transmission queue. In the second term, the arriving packet finds the other buffer empty and $n \geq 0$ packets waiting in its buffer; as a result its response time is the sum of $n + 1$ successive service times each of duration U_i. Note that the distribution $T_i(x)$ is exponential with a jump of size $1 - p_0 \frac{\mu_i}{\mu_i - \lambda_i}$ at the origin. By straight forward calculations, the mean response time of coding queue i is:

$$E[T_i] = p_0 \frac{\mu_i}{(\mu_i - \lambda_i)^2} \tag{5.21}$$

The average output packet rate from the coding stage can be obtained as follows:

$$\varphi = \sum_{n=1}^{\infty} p_n \mu_1 + \sum_{n=-1}^{-\infty} p_n \mu_2 = \Lambda - \frac{\lambda_1 \lambda_2 (r_1 + r_2)}{\lambda_1 r_1 + \lambda_2 r_2 + r_1 r_2} \tag{5.22}$$

which can be written as $\varphi = \Lambda - \varphi_c$, where φ_c denotes the output rate of coded packets. We can see that $\varphi_c > 0$ for all values of $r_i < \infty$; thus an opportunity for NC may arise even when the timers are very small.

Transmission queue

Exact analysis of the transmission queue under the present assumptions is very difficult; indeed even for the simplest case of exponential transmission times, the generating function for the joint coding and transmission queue lengths can only be solved using boundary value problem techniques [127–130] which may not be practical for optimising performance in real time. Thus we analyse the response time in the transmission queue approximately using the decomposition approach in [27] which will be based on the unjustified assumption that the departure process from the coding stage is a renewal process.

In particular, we approximate the distribution of the arrival process to the transmission queue A_3 or equivalently the departure process from the coding stage using the stationary interval method that ignores the correlation between successive departures:

$$A_3(x) = \sum_{n=1}^{\infty} \pi_n U_1(x) + \sum_{n=-1}^{-\infty} \pi_n U_2(x) + \pi_0 \int_0^x A(x-y)\big[\rho_1\, dU_1(y) + \rho_2\, dU_2(y)\big] \tag{5.23}$$

Note that $\varphi = E[A_3]^{-1}$. The transmission queue contains both coded and non-coded packets which have different distribution when packet size is not constant. In order to circumvent this difficulty, we apply an approximation similar to the one used for ONC so as to obtain an equivalent service time distribution:

$$S_3(x) = \frac{\varphi_c}{\varphi}\ S(x)^2 + \left[1 - \frac{\varphi_c}{\varphi}\right]\ S(x) \tag{5.24}$$

The average packet delay in the transmission queue $E[R_3]$ is then estimated using the diffusion approximation for a $G/G/1$ system [125] with arrival and service time distributions $A_3(x)$ and $S_3(x)$ respectively.

5.2.2. Optimising the time-outs over a single-hop

Now we use the analytical results derived previously in order to investigate the trade-offs associated with the choice of the time-outs when decoding is immediate at the next hops. In particular, we consider the two-way relay network in Figure 2.2 assuming that packets are received successfully by the relay node even if more than one packet arrives at the same time [84]. The performance measures of interest are delay, bandwidth utilisation and energy consumption.

Response time

Given a set of input rates λ_i, a packet size distribution L and a capacity of the output link c, we seek to determine the optimal time-out parameters r_i that minimise an aggregate objective function of the response time of class 1 and 2 packets in the system:

$$\begin{aligned} \underset{r_1,r_2}{\text{Minimise}} \quad & \mathcal{D} = E\big[wT_1 + (1-w)T_2 + R_3\big] \\ \text{subject to} \quad & r_i > \lambda_i - \lambda_{3-i}, \quad i \in \{1,2\} \\ & \varphi_c\ E[\mathcal{L}] + \big(\Lambda - 2\varphi_c\big)\ E[L] < c \end{aligned} \tag{5.25}$$

where $\mathcal{L}$ is the random variable representing the largest of the two packet lengths, and $w \in [0,1]$. We choose $w = 0.5$ in order to provide equal priority to both packet classes.

For symmetric traffic load $\lambda_i = \lambda$ and constant packet size with $\mu = E[S]^{-1}$, the stability conditions for the coding and transmission queues (the constraints in (5.25)) reduce to:

$$\begin{aligned} & 0 < r, && \text{if} \quad \Lambda < \mu \\ & 0 < r < \frac{\Lambda(\mu-\lambda)}{\Lambda-\mu}, && \text{if} \quad \mu \leq \Lambda < 2\mu \end{aligned} \tag{5.26}$$

Note that when the output link is lightly loaded $\Lambda << \mu$, NC cannot improve delay performance and therefore plain routing is optimal. On the other hand, under moderate traffic conditions where $\Lambda < \mu$, the node is stable with routing but using NC can reduce both delay and transmission costs. Finally, when the output link is saturated ($\mu \leq \Lambda < 2\mu$), NC must be performed in order to stabilise the transmission queue.

Bandwidth utilisation

We use coding gain η as a measure of bandwidth efficiency of NC in comparison to plain forwarding. We have

$$\eta = \frac{\Lambda E[L]}{\varphi_c E[\mathcal{L}] + (\Lambda - 2\varphi_c)E[L]} \in [1, 2] \tag{5.27}$$

The problem of maximising the coding gain, however, has a trivial solution $r_i^* = 0$, which renders the coding stage unstable. Hence, there is a compromise to be achieved between delay and coding gain which will be illustrated in the numerical results.

Energy consumption

Reducing energy consumption is becoming a major design issue in networking especially in sensor networks where nodes are typically resource-constrained and rely on batteries or energy harvesting. Thus we consider the energy efficiency of NC against routing in the two-way relay network. We follow an approach similar to the one presented in [103] but unlike their work, we allow coding and energy costs to depend on the packet size. Moreover, we include an energy cost $\mathcal{E}_p$ for receiving a packet by a node even if the node is not the intended receiver (due to the broadcast nature of the wireless medium). Let ε_c and ε_{tr} be the energy required to XOR two bits and to transmit a single bit, respectively. The average energy consumption per unit time (power) under plain forwarding is then given by:

$$\mathcal{P}_f = 3\Lambda\mathcal{E}_p + \Lambda E[L]\varepsilon_{tr} \tag{5.28}$$

Here the first term accounts for the fact that each packet will be processed by three nodes: the relay node and the two neighbouring nodes. The energy cost of NC involves a trade-off: on the one hand, it reduces the number of transmissions by the relay node and, consequently, the number of packets processed at the two receivers; on the other hand, it adds a coding cost at the relay node and a decoding cost at each of the neighbouring nodes. Hence, the average energy consumption per unit time under NC is given by:

$$\mathcal{P}_c = \big(\Lambda + 2\varphi\big)\mathcal{E}_p + \Big(\varphi_c E[\mathcal{L}] + \big(\Lambda - 2\varphi_c\big)E[L]\Big)\varepsilon_{tr} + 3\varphi_c E[\mathcal{L}]\varepsilon_c \tag{5.29}$$

NC therefore reduces the power consumption if and only if the following condition is satisfied:

$$3E[\mathcal{L}]\varepsilon_c < \big(2E[L] - E[\mathcal{L}]\big)\varepsilon_{tr} + 2\mathcal{E}_p \tag{5.30}$$

Next we present some numerical examples. Figure 5.6(a) depicts the mean coding, transmission and total delays in the system vs the mean time-out period for a balanced system under moderate traffic condition, $\Lambda = 2$ and $\mu = 2.5$. In this case, the range of time-out parameters which stabilises the system is $[0, \infty)$ where the lower bound corresponds to the non-coding case. The figure indicates that the approximation yields very accurate results as compared to simulations. We can also observe that the coding delay is monotonically increasing while the transmission delay is exponentially decaying with the time-out period. Hence, the total delay in the system decreases up to a point (the optimal) after which it increases continuously. More specifically, network coding reduces the average response time by up to 24% as compared to plain forwarding while at the same time offering a coding gain of 1.23. Furthermore, a coding gain of 1.5 can be achieved while maintaining a similar response time to the non-coding system.

Figure 5.6(b) depicts similar numerical results under heavy traffic conditions where forwarding packets without coding renders the transmission queue unstable. The parameters used in the figure are $\Lambda = 2$ and $\mu = 1.4$, which imply that the average time-out periods can vary in the range $(0.75, \infty)$ in order to stabilise the system. In this case, network coding yields a minimum response time of about 3.7 *sec* while offering a coding gain of 1.7. Higher coding gains can also be achieved at the expense of higher delays.

Figures 5.7 and 5.8 depict the trade-off curve between delay and coding gain as a result of varying the time-out for the set of parameters of Figure 5.6 and for a balanced system under different traffic conditions, respectively. The results suggest that the time-out should not be smaller than the optimal value since the curve is almost symmetric in the vicinity of this point. In other words, if δ^- and δ^+ are positive numbers then, for any average time-out $1/r^* - \delta^-$ within the stability region there is another time-out value $1/r^* + \delta^+$ which maintains the same delay performance while achieving higher coding gain. Figure 5.8 also shows that the trade-off curve becomes narrower as the traffic load increases.

In Figure 5.9, we plot the inverse of the aggregate delay function when the arrival rates of the two flows are moderately different. The figure shows that there is an optimal set of time-out parameters which maximises the inverse of the delay cost function, yielding a delay improvement of about 54% and a coding gain of 1.26 in comparison to routing. When the arrival rates of the two flows are highly unbalanced, we have observed that the optimisation problem depends mainly on the time-out parameter for the faster flow while the optimal time-out parameter for the slower flow is simply zero.

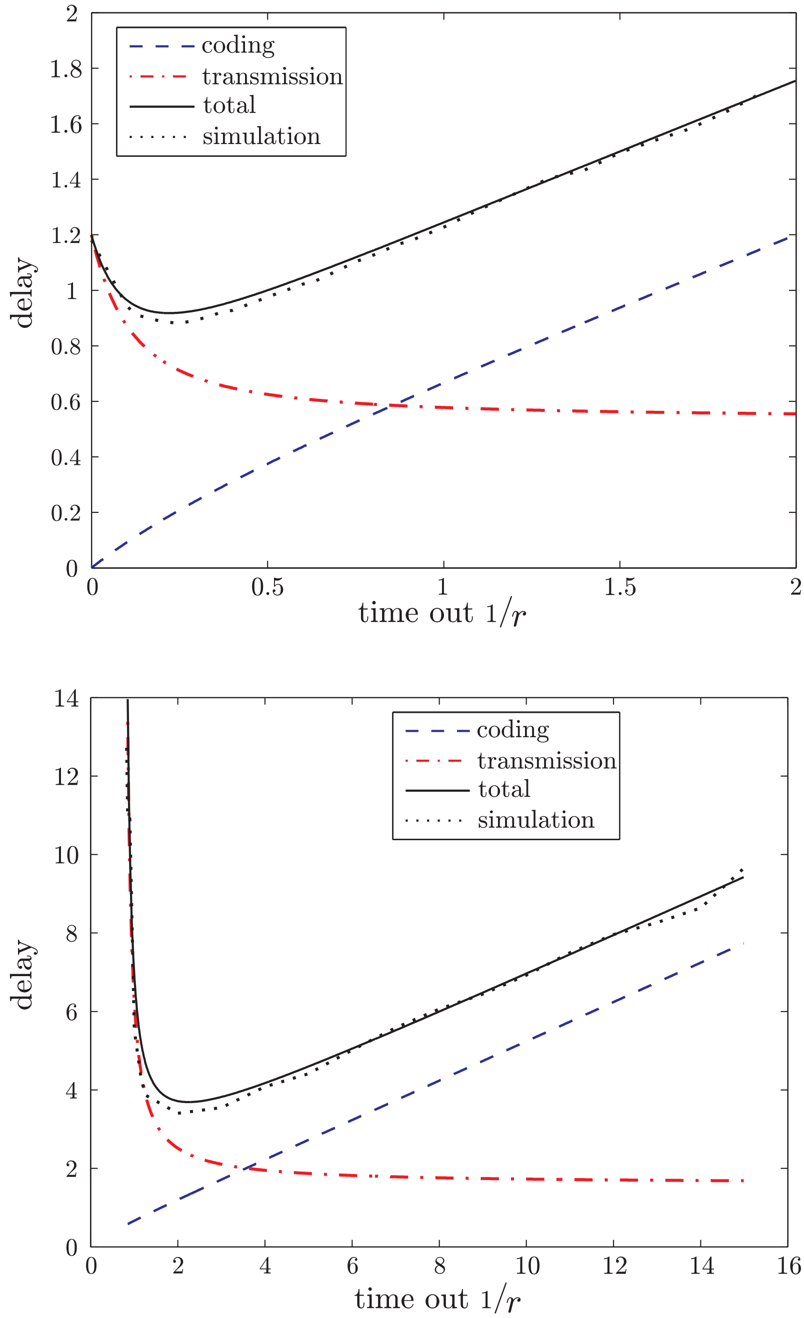


Figure 5.6.: Average coding, transmission and total delays vs average time-out $1/r$ for balanced load $\lambda = 1$ and constant packet length with (a) $\mu = 2.5$ and (b) $\mu = 1.4$.

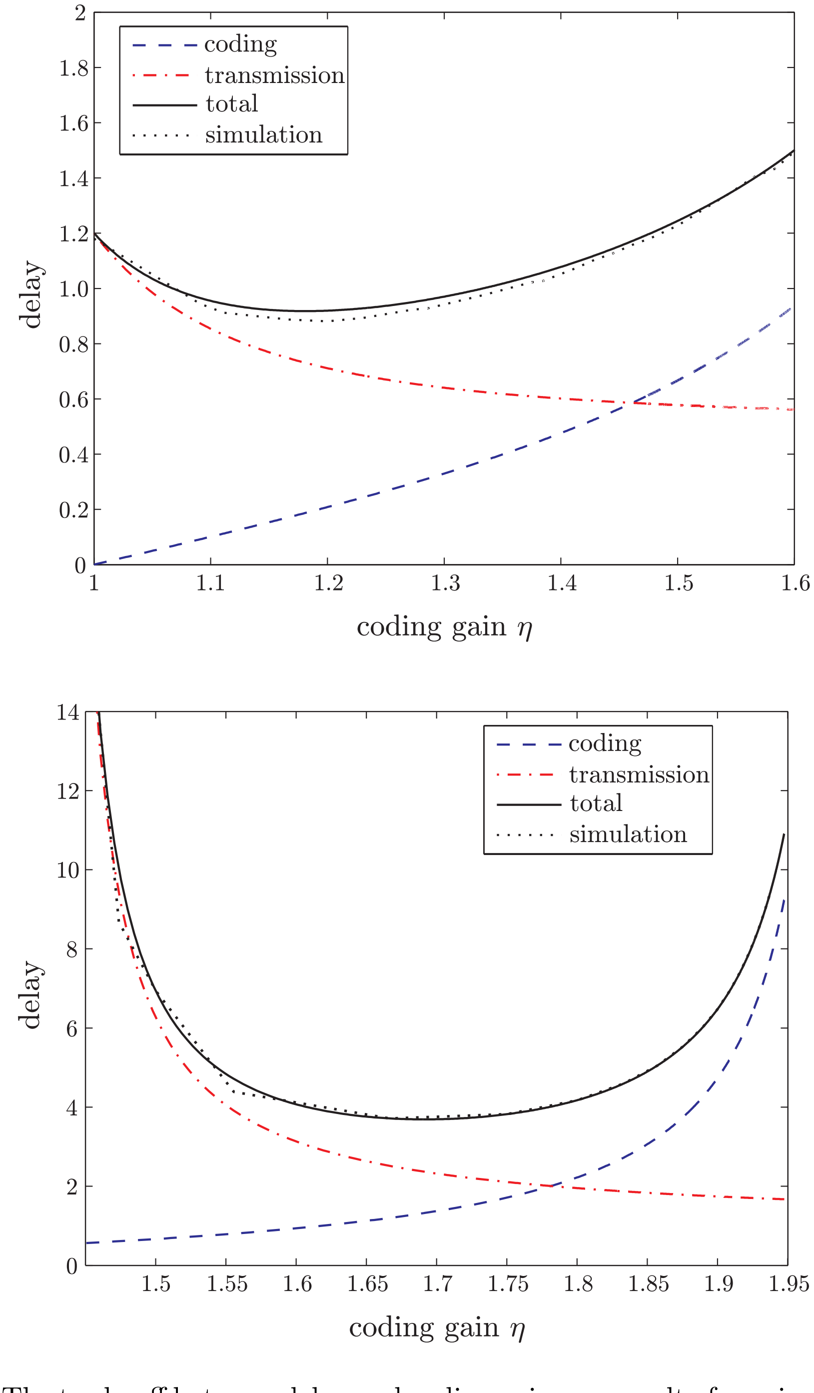


Figure 5.7.: The trade-off between delay and coding gain as a result of varying the time-out using parameters of Figure 5.6.

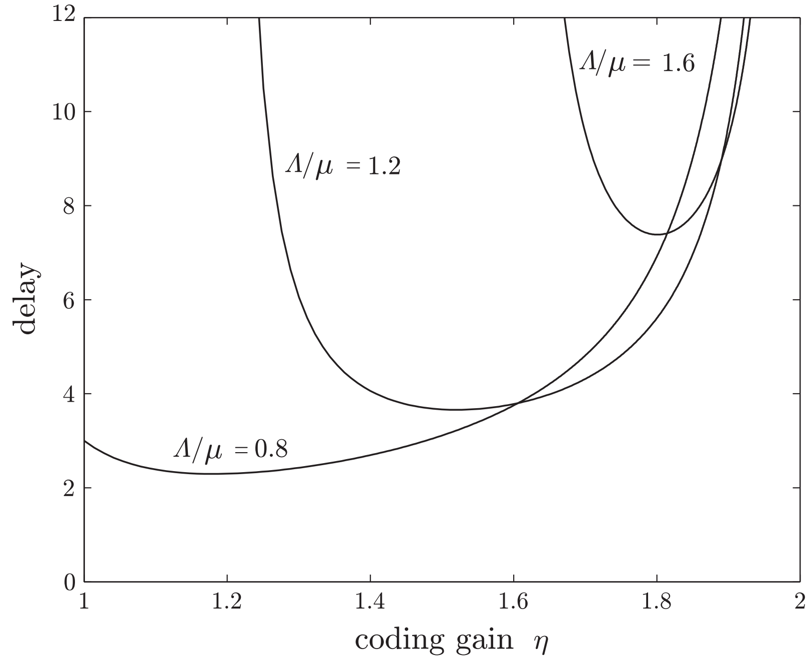


Figure 5.8.: The trade-off between delay and coding gain as a result of varying the time-out period for a balanced traffic under different traffic conditions with constant packet length $\mu = 1$.

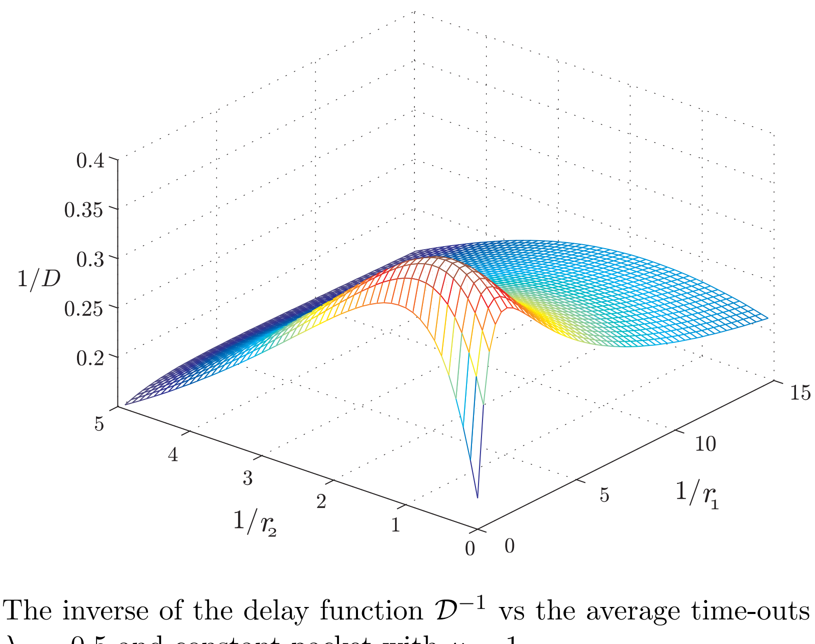


Figure 5.9.: The inverse of the delay function $\mathcal{D}^{-1}$ vs the average time-outs for $\lambda_1 = 0.4$, $\lambda_2 = 0.5$ and constant packet with $\mu = 1$.

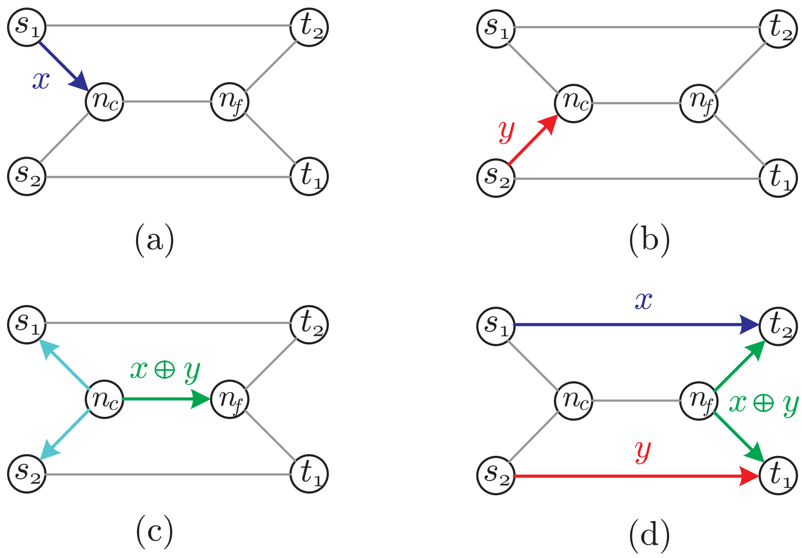


Figure 5.10.: The feedback mechanism in a butterfly network with two independent unicast sessions between the source-destination pairs (s_i, t_i), $i \in \{1, 2\}$. When two packets are combined, node n_c sends a feedback message to s_i requesting retransmission of a packet to t_{3-i}.

5.2.3. Extension to multi-hop networks

In multi-hop networks, where decoding is not possible at the next hop of the encoding node, each source needs to send *remedy* packets to the destination nodes of those packets with which its own are combined. A direct consequence of performing NC asynchronously, however, is that senders will not know in advance which of their packets will be combined at intermediate nodes. Yet, sending a remedy for *every* packet in a flow will lead to poor utilisation of network resources as only a small proportion of those packets might be used for decoding. A potential solution to this problem is to implement a feedback mechanism at the encoding node [81] which requests retransmission of additional packets if NC is performed. Note that in a directed graph, edges in the reverse direction can be considered as low bandwidth links that can be used to relay feedback and protocol information, but they may not be used by the underlying routing protocol to carry traffic.

As an example, consider the butterfly network of Figure 5.10; if two packets are combined at node n_c then the node can send feedback packets to source nodes s_1 and s_2 requesting retransmission of remedy packets to receivers t_2 and t_1, respectively. In turn, node n_f will multicast received coded packets to both destination nodes and unicast un-coded packets to their intended recipient. This scheme thus requires source nodes to retain copies of their transmitted packets until a remedy transmission request is received, after which the requested packet in addition to all packets with smaller sequence numbers can be discarded

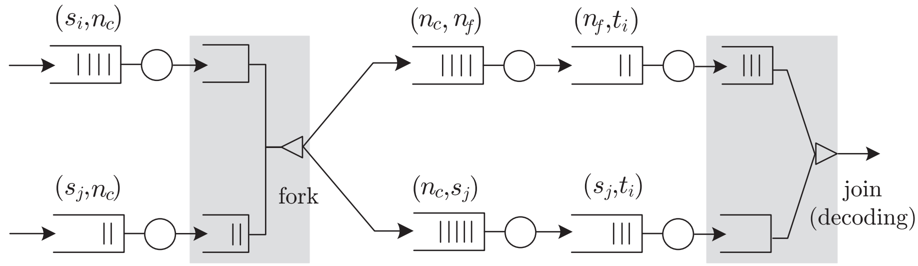


Figure 5.11.: Queueing network model for the source destination pair (s_i, t_i) of the butterfly network when coding takes place at n_c.

(assuming single path routing so that packets arrive at the encoding node in sequence). Figure 5.11 depicts the queueing network model for the source-destination pair (s_i, t_i) when coding is performed at the intermediate node. If, on the other hand, a packet passes through node n_c without being encoded, then the equivalent model is a tandem queueing network.

Note that the queueing behaviour of coded packets resembles that of fork-join queueing networks (FJQNs), where a fork occurs at the encoding node to represent the simultaneous transmission of coded and feedback packets, while join primitives occur at the destination nodes to represent decoding operations. Such queueing networks are analytically intractable and only bounds are available for their performance [106].

We will assume that packets must be received at the destination node in sequence so that un-coded packets may experience a form of *resequencing* delay upon arrival at the destination node if a previously received coded packet has not been decoded yet. Thus the arrival of a remedy packet at the output node may release a batch of packets from the decoding buffer; this highly correlated batch departure process also complicates the analysis of the queueing network significantly. In the sequel we will obtain an approximate upper bound for the average end-to-end delay for both coded and un-coded packets, which will be achieved by focusing on the former since they experience, on average, higher transmission delays in links and synchronisation delays at the output.

Heuristic for optimising the average end-to-end delay

A well known upper bound for the total delay of parallel FJQNs is based on the properties of associated random variables [106] which we summarise as follows. Denote by D_i^p and D_i^r, respectively, the total delays over the primary routing path $n_c \to n_f \to t_i$ and the remedy path $n_c \to s_j \to t_i$; these two random variables are associated because of the fork synchronisation primitive. If $\mathcal{D}_i^p$ and $\mathcal{D}_i^r$ are the independent versions of the associated variables such that:

(a) D_i^p and $\mathcal{D}_i^p$ (D_i^r and $\mathcal{D}_i^r$) have the same distribution;

(b) $\mathcal{D}_i^p$ and $\mathcal{D}_i^r$ are mutually independent

then the following inequality holds:

$$\Pr[\max(D_i^p, D_i^r) \leq x] \geq \Pr[\max(\mathcal{D}_i^p, \mathcal{D}_i^r) \leq x]$$

and as a consequence:

$$E[\max(D_i^p, D_i^r)] \leq E[\max(\mathcal{D}_i^p, \mathcal{D}_i^r)] \tag{5.31}$$

The above bound indicates that we can obtain an upper bound for the expected fork-join delay of coded packets by assuming the two path delays independent when in fact they are associated but not independent. However, obtaining the distribution of the independent versions of the path delays is yet extremely difficult. Thus we propose a simple heuristic for selecting the time-out parameters which requires knowledge of the average traffic flow characteristics over each link in the network. More precisely, we assume that link $u \to v$ behaves as an $M/M/1$ queueing system with arrival rate $\lambda(u, v)$ which is equal to the average packet rate on the link, and service rate $\mu(u, v)$ given by the ratio of the capacity of the link $c(u, v)$ to the average packet size in the flows traversing the link. Consequently, the response time of link $u \to v$ follows the exponential distribution:

$$R(u, v) \sim \text{Exp}\big(\mu(u, v) - \lambda(u, v)\big) \tag{5.32}$$

Applying this approximation to the butterfly network yields:

$$\begin{aligned}
\lambda(n_c, n_f) &= \varphi, \quad \mu(n_c, n_f) &= \frac{c(n_c, n_f)\ \varphi}{\varphi_c E[\mathcal{L}] + \big(\varphi - \varphi_c\big) E[L]} \\
\lambda(n_f, t_i) &= \lambda_i, \quad \mu(n_f, t_i) &= \frac{c(n_f, t_i)\ \lambda_i}{\varphi_c E[\mathcal{L}] + \big(\lambda_i - \varphi_c\big) E[L]} \\
\lambda(n_c, s_j) &= \varphi_c, \quad \mu(n_c, s_j) &= \frac{c(n_c, s_j)}{FB} \\
\lambda(s_j, t_i) &= \varphi_c, \quad \mu(s_j, t_i) &= \frac{c(s_j, t_i)}{E[L]}
\end{aligned}$$

where FB denotes the size of feedback packets.

Now we select the time-out parameters to minimise a cost function $\mathcal{D}$ which combines the coding delay at the encoding node (5.20) and an approximate upper bound for the

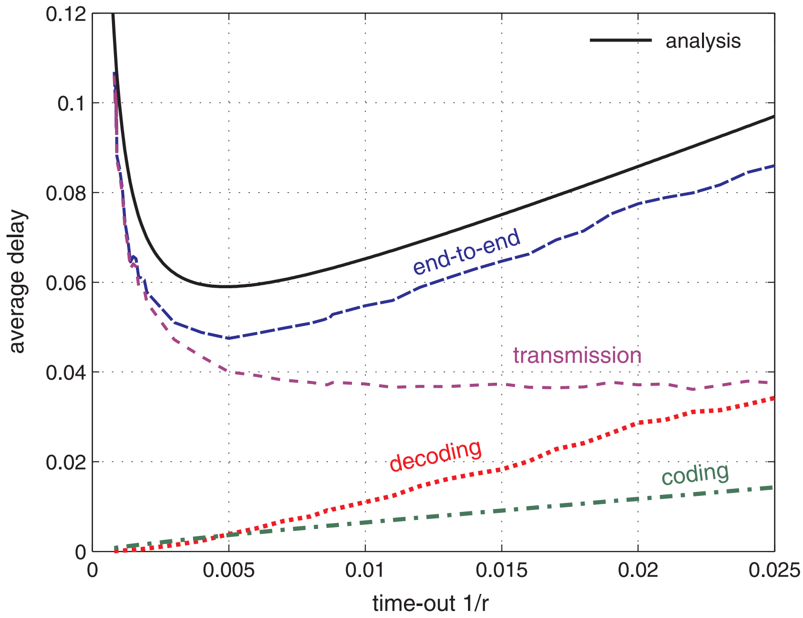


Figure 5.12.: Simulation results and the analytical delay bound of (5.33) for the butterfly network with the following parameters: capacity of the remedy links $0.5Mb/s$, capacity of feedback links $128kb/s$, capacity of other links in the network $1Mb/s$, symmetric Poisson traffic of rate $0.5Mb/s$ with exponential packet length $E[L] = 512B$, and $FB = 80B$.

delay of coded packets obtained using (5.31) and (5.32):

$$\begin{aligned} \underset{r_1,r_2}{\text{Minimise}} \quad & \mathcal{D} = \frac{1}{2}\sum_{i=1}^{2} E\Big[T_i + \max\Big(R(n_c,n_f) + R(n_f,t_i), R(n_c,s_j) + R(s_j,t_i)\Big)\Big] \\ \text{subject to} \quad & r_i > \lambda_i - \lambda_j, \qquad i \in \{1,2\}, \quad j = 3-i \\ & \lambda(u,v) < \mu(u,v), \quad u,v \in \{s_i, n_c, n_f, t_i\} \end{aligned} \tag{5.33}$$

Figure 5.12 presents simulation results for the expected value of the different delay components in the butterfly network: coding delay at the encoding node; sum of link delays on the main routing path; synchronisation delay at the output including decoding of coded packets and resequencing of un-coded packets; and the total end-to-end delay. In the figure, decoding delay increases steadily with the time-out due to congestion in the feedback and remedy links. Figure 5.12 also shows the delay cost function $\mathcal{D}$, from which we can observe that both simulation and analytical predictions have a minimum around a time-out value of $0.05ms$. This indicates that the proposed heuristic provides a simple technique for optimising the end-to-end delay performance of NC.

5.3. Summary

In this chapter queueing models for two possible implementations of inter-session NC have been proposed and analysed. We first presented a single stage model for joint opportunistic coding and transmission, and analysed its performance by a decoupling approximation. We then presented a multistage queueing model for a NC implementation in which packet coding and transmission are performed independently, and nodes use timers to modify coding opportunities. We have incorporated this model into a network setting and proposed a simple heuristic, based on fork-join synchronisation primitives, to optimise the end-to-end delay performance for butterfly-like network structures. In both cases, the accuracy of the proposed approximations was verified through many comparisons with simulations, and the approximations were found to work out well. We have used the analytical models to evaluate the performance of the proposed coding schemes in comparison to a conventional *store and forward* network, and we have shown that NC can improve performance significantly in a moderate to heavily loaded system.

6. Conclusions and future work

The aim of this thesis was to develop mathematical models that would assist in optimising the performance of multi-hop networks and improve our understanding of the trade-offs that govern their behaviour. Specifically, the thesis addressed the following issues:

- Spatial non-homogeneity of large networks which may arise from their structure, transmission medium, routing policy and security mechanism used by relay nodes. Optimisation problems of interest include: optimising packet delivery performance through a judicious choice of the time-out, and allocating resources to protect a node against malicious packets.
- The delay-energy trade-offs associated with sending redundant packets into large-scale networks to mitigate the effects of packet losses and routing inaccuracies.
- The queueing behaviour of NC of multiple stochastic flows, including packet synchronisation at encoding and decoding nodes. The figures of merit being delay, throughput, energy consumption and bandwidth utilisation.

The following section summarises the contributions of the thesis towards solving the above problems and outlines the main results of the research.

6.1. Conclusions

In this thesis we used Brownian motion as a model of packet traversal of a large wireless or wired network. The model accounts for network non-homogeneity regarding routing and the loss rate that the packet experiences as it passes successive segments of a source to destination route. A mixed analytical-numerical solution technique has been developed to compute the average time to reach the destination, including recurrent retransmission by the source after time-outs, and the average energy expended by the packet while moving through the network. The approach is based on using a finite but unbounded number of internally homogeneous segments. The model is able to capture the effects of increased loss rate in areas remote from the source and destination, variable rate of advancement towards destination over the route, as well as of defending against malicious packets within a certain distance from the destination. The role of time-outs to optimise the performance of wireless networks has been examined; by varying the value of the time-out and studying the locus of the energy expended versus the time taken by the search, we can notice desirable operating areas where both of these parameters of interest are minimised. In the context of network security, we have applied the model to investigate the effect of using intermediate nodes that perform DPI and packet drops in protecting a destination node from malicious packets. We have shown that a scaling of the drop rate can be found such that the destination can be indefinitely protected from dangerous traffic in spite of infinite resending. A form of phase transition has also been observed concerning the eventual success of the attack depending on the relative speed of approach of the dangerous traffic and the intensity of protections which block the attacker's progress. Another interesting result is that, with fixed protection resources, there is an optimum size of protection space around the destination which maximises the delay before the attacker reaches the destination.

We then applied the Brownian motion model to analyse the time-dependent properties of the travel processes of a single, duplicate and coded packets over homogeneous networks. In particular, we derived the distribution of the total forwarding delay, total energy consumption and energy expended by a packet at any time instant from the start of the search. The results were obtained in the form of LTs which were inverted numerically and applied to some useful questions. We first applied the results to estimating the total energy budget needed for a communication that includes a known number of packets. Then we compared two known techniques for improving reliable packet delivery by sending N redundant packets along independent paths. In the first one, the source node sends duplicate packets so that transmission is considered to be successful when at least one of the copies reaches the destination node. In the second, erasure coding is used such that any k-out-of-N packets are enough to decode a block of k packets. Numerical examples suggest that while coding may result in higher overall packet delay and energy consumption on average, it reduces variations significantly and thus reduces the uncertainty in packet delivery performance. This indicates that erasure coding could be more appropriate for real-time applications which have stringent delay requirements. We also investigated the performance trade-offs in erasure coding resulting from varying the number of transmitted packets N. We have shown that a small excess in N above k can reduce decoding time significantly at the cost of a small increase in energy utilisation; this can be achieved provided that the destination node is equipped with a feedback mechanism to abort further transmissions in the network after decoding is successful.

Finally, we considered a store and forward packet network in which NC is being used to co-encode packets from distinct flows. In such a system both the encoding and decoding process introduce additional delays, and we developed analytical models to evaluate the resulting performance. In particular, we proposed and analysed queueing models for two possible implementations of inter-flow NC at a network router. We first presented a single server queueing model for a cross-layer NC design which coordinates packet coding and transmission. We proposed a decoupling approximation which reduces the solution method to an iterative non-linear numerical solution. The approach is based on constructing, for each flow, an equivalent queue with vacations that takes into account the additional waiting time for service which a flow encounters while other flows are being encoded. We derived results for the average response time, throughput and coding gain for arbitrary number of flows that have non-Markovian arrival and service processes. We then presented a multistage queueing model which introduces processing and coding queues prior to the transmission queue in order to analyse the performance of NC when these distinct functionalities are decoupled. We assumed that the router employs a time-out mechanism in order to accumulate packets for coding, and we evaluated the trade-offs associated with varying the length of the waiting time. We obtained analytical expressions for the throughput, coding gain, energy consumption and response time in terms of traffic rates, packet size distribution and time-outs for the case of two Poisson traffic streams. We have incorporated the model into a network setting and proposed a simple heuristic, based on fork-join synchronisation primitives, to optimise the average end-to-end delay in butterfly-like network topologies. The proposed approximations were validated through many comparisons with discrete event simulations and were found to work out very well. We have also used the analytical models to evaluate the performance of the proposed coding schemes in comparison with a conventional store and forward network, and we have shown that NC can offer significant gains provided that decisions to code packets from distinct flows is made a function of network traffic conditions.

6.2. Future work

There are several areas where the research presented in this thesis can be extended. In this section we discuss open issues and provide possible directions for future work.

In Chapter 3 we focused on developing a generic analytical framework to represent a packet's travel in large networks, but we have not addressed the problem of how the parameters of the model can be obtained when a specific routing protocol is deployed over a complex topology. To go beyond a purely theoretical impact, realistic examples of translating the assumed loss and advancement rates into existing environments and protocols should be considered. Examples of application areas include: packet traversal through an opportunistic network [14, 131, 132], data query [47] and gradient routing [133, 134] in a wireless sensor network, and search for a file in an unstructured peer-to-peer network [135]. In such networks, the location of the destination may not be known when a packet starts its travel. Furthermore, it may be difficult to construct accurate routing tables in highly dynamic topologies, and intermediate nodes may relay the packet away from its destination. As the packet approaches the destination, routing may improve as more accurate or recent information becomes available regarding its whereabouts. Similarly, we presented a theoretical analysis of how network attacks can be prevented through the use of DPI and packet drops. We addressed the question of whether such a mechanism can effectively stop an attack, but we did not investigate the relation between DPI parameters and the loss rates in the model. Parameters such as the number of nodes performing DPI, the percentage of packets that undergo DPI and the mechanism employed to detect a malicious packet would all affect average detection rates. Thus mapping techniques that can accurately capture the network parameters of interest are required.

The diffusion model presented in this thesis deals with a particular problem, namely the search by a packet for a destination node in a multi-hop network. However, it addresses a large class of other problems concerning search for recognisable objects in large random or imprecisely known environments. Such problems arise in robotic search for mines in a minefield [136]; search for information in the web or in large databases with uncertain or approximately represented data such as the content of images [137]; and automatic theorem proving where portions of proofs have to be sought by sifting through a large database. In many such contexts the search space is not homogeneous, and it may be possible to move or search more rapidly in some parts of the search than in others; search may also be easier in certain parts of the search space due to hints or other useful directional information that may be available. Furthermore, in many such contexts, the searcher's progress may be impeded or blocked, or even destroyed for instance for lack of fuel during a robotic search or due to unreliability in the computational mechanisms when we are dealing with a software search for data. The diffusion model also provides a framework for the study of intermittent search strategies [55–57, 60] which mimic animals' foraging patterns [58, 59]. Specifically, the combination of local diffusive and ballistic motions characterising such search mechanisms can be represented by timer that triggers the searcher to jump to a random location instead of restarting from the initial position, as in our approach. Thus an interesting research direction is to apply the diffusion model to the aforementioned research areas and perhaps relate its findings to real world experiments.

The time-dependent analysis presented in Chapter 4 can be extended in several ways. Firstly, the distributions of the travel time and energy consumption of a single packet were obtained in the form of one-dimensional LTs that were inverted numerically. Since it is difficult to invert the LTs analytically, one can attempt to deduce the asymptotic behaviour which could be determined from singularities in the complex plane. Furthermore, the energy consumption results for coded transmissions require numerical inversions of two-dimensional LTs which can be time-consuming. Thus it would be useful to extend the steady-state approach in [9] to the case of erasure coding, in order to derive analytical expressions for the average performance. Secondly, we have assumed that the N searchers travel independently of each other, which may not represent some cases of interest that involve collaborative behaviour or use of memory; for instance, a searcher which has exhaustively searched a particular area may leave “negative hints” that would encourage others to relaunch their search away from where they are currently. Lastly, one can generalise the time-dependent results to non-homogeneous environments which would require more mathematically sophisticated techniques.

For the problem of optimising the performance of inter-flow NC using time-outs, we have only considered in Chapter 5 the case of two Poisson flows in butterfly-like network structures. It would be useful to extend the results to arbitrary number of flows, which is likely to require further approximations as the state space of the model becomes very large. Bounds and approximations are also required in order to optimise performance in more complex network topologies, for instance networks with multiple encoding and decoding points. One can also apply similar techniques to optimise the multicast performance of intra-flow NC. Indeed, the subgraph representing the flow of packets from a source to a single destination of a multicast session can be transformed into a FJQN. In this network, fork primitives occur at points where multicast operations are performed, while join primitives represent both encoding at intermediate nodes and decoding at the output of the network. Finally, the queueing models presented in this thesis, which seem to provide adequate delay approximations, can also be used to design efficient algorithms for routing and flow control under NC.

Appendices

A. The diffusion equation

In this appendix we outline the derivation of the diffusion equation which is a second-order partial differential equation describing *approximately* the time evolution of the pdf of the position of a particle undergoing Brownian motion [6]. Our treatment follows that of [108, 138–140].

Let $\{X(t) : t\}$ be a one-dimensional stochastic process denoting the position of a Brownian particle at time $t \geq 0$ and let $f(x,t)$ indicate the pdf of $X(t)$, i.e. $f(x,t)dx = \Pr[x \leq X(t) \leq x + dx]$. The process $X(t)$ is assumed to be a Markov process which is continuous in both time and space, thus for any arbitrary times $t_1 < t_2 < \cdots < t_n$ we have

$$\begin{aligned}\Pr[X(t_n) = x_n | X(t_{n-1}) = x_{n-1}, \cdots, X(t_1) = x_1]& \\ = \Pr[X(t_n) = x_n | X(t_{n-1}) = x_{n-1}]&\end{aligned}$$

Furthermore, for a small time interval Δt, the pdf of $X(t)$ satisfies the following equation:

$$f(x, t + \Delta t) = \int_0^\infty f(x - \Delta x, t) q(x - \Delta x, t; \Delta x, \Delta t) d\Delta x \tag{A.1}$$

where the transition probability q is defined as

$$q(x, t; \Delta x, \Delta t) = \Pr[X(t + \Delta t) = x + \Delta x | X(t) = x]$$

From the Taylor expansion of the right hand side of (A.1) about the point x we obtain:

$$f(x, t + \Delta t) = \int_0^\infty \sum_{n=0}^\infty \frac{(-\Delta x)^n}{n!} \frac{\partial^n [f(x,t) q(x, t; \Delta x, \Delta t)]}{\partial x^n} \; d\Delta x$$

If we assume that $X(t)$ evolves through the effect of many small displacements, then high order moments of the jumps with $n > 2$ become negligible as $\Delta t \to 0$:

$$\begin{aligned} f(x, t + \Delta t) \approx \; & f(x,t) - \frac{\partial}{\partial x} \big[f(x,t) \int_0^\infty \Delta x \; p(x, t; \Delta x, \Delta t) d\Delta x \big] \\ & + \frac{1}{2} \frac{\partial^2}{\partial x^2} \big[f(x,t) \int_0^\infty (\Delta x)^2 p(x, t; \Delta x, \Delta t) d\Delta x \big] \end{aligned} \tag{A.2}$$

Let $b(x,t)\Delta t$ be the mean change over time $[t, t + \Delta t[$ of the position of the particle given that $X(t) = x$, and let the variance of the distance travelled over the same time interval be $c(x,t)\Delta t$:

$$b(x,t)\Delta t = \int_0^\infty \Delta x \; q(x,t;\Delta x,\Delta t) \; d\Delta x$$
$$c(x,t)\Delta t = \int_0^\infty (\Delta x)^2 q(x,t;\Delta x,\Delta t) \; d\Delta x - o(\Delta t) \tag{A.3}$$

Now we can substitute (A.3) into (A.2), divide both sides by Δt and take the limit as $\Delta t \to 0$ to get:

$$\frac{\partial f(x,t)}{\partial t} = -\frac{\partial [b(x,t)f(x,t)]}{\partial x} + \frac{1}{2}\frac{\partial^2 [c(x,t)f(x,t)]}{\partial x^2} \tag{A.4}$$

which is the well-known diffusion equation.

The probability current $I(x,t)$ or the rate of flow of probability, in the positive direction, across a point x at time t is given by:

$$I(x,t) = -\left[-b(x,t)f(x,t) + \frac{1}{2}\frac{\partial [c(x,t)f(x,t)]}{\partial x}\right] \tag{A.5}$$

This follows from the fact that the probability of the diffusion process lying within an interval $[x, x+\Delta x]$ is $f(x,t)\Delta x$ and so the rate of change of the probability current across this narrow segment is [140]:

$$\frac{\partial f(x,t)\Delta x}{\partial t} = I(x,t) - I(x+\Delta x,t) \tag{A.6}$$

Dividing both sides by Δx and taking the limit as $\Delta x \to 0$ yield:

$$\frac{\partial f(x,t)}{\partial t} = -\frac{\partial I(x,t)}{\partial x} \tag{A.7}$$

hence (A.5) is obtained from (A.4) and (A.7).

The diffusion process $X(t)$ can be expressed by the following stochastic differential equation [27]:

$$dX(t) = b(X(t),t)dt + \sqrt{c(X(t),t)}dX_0(t) \tag{A.8}$$

where $X_0(t)$ denotes the Wiener process (Standard Brownian motion) with $b(z,t) = 0$ and $c(z,t) = 1$. More specifically, $X_0(t)$ is characterised by: (a) It is continuous, (b) $X_0(0) = 0$ and (c) $X_0(t) - X_0(\tau) \sim \mathcal{N}(0, t-\tau)$, i.e., independent and normally distributed increments with mean 0 and variance $(t-\tau)$. Thus if $X(t) = x$ then in a small time interval Δt, the process changes its position by an amount which is *approximately* normally distributed with mean $b(x,t)\Delta t$ and variance $c(x,t)\Delta t$ and is independent of the past behaviour of the process.

Bibliography

[1] E. Gelenbe and G. Loukas, "A self-aware approach to denial of service defence," *Computer Networks*, vol. 51, no. 5, pp. 1299 – 1314, 2007.

[2] I. Akyildiz, W. Su, Y. Sankarasubramanian, and E. Çayirci, "A survey on sensor networks," *IEEE Communications Magazine*, vol. 40, no. 8, pp. 102–114, 2002.

[3] C. E. Perkins, *Ad Hoc Networking.* New York: Addison Wesley, 2000.

[4] R. Ahlswede, N. Cai, S.-Y. Li, and R. Yeung, "Network information flow," *IEEE Transactions on Information Theory*, vol. 46, no. 4, pp. 1204–1216, Jul. 2000.

[5] C. Fragouli, J.-Y. Le Boudec, and J. Widmer, "Network coding: an instant primer," *ACM SIGCOMM Computer Communication Review*, vol. 36, pp. 63–68, Jan. 2006.

[6] A. Einstein, *Investigations on the Theory of Brownian Motion.* New York: Dutton & Dover, 1926.

[7] E. Gelenbe, "Travel delay in a large wireless ad hoc network," in *Proc. 4th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt'06)*, Boston, MA, USA, Apr. 2006.

[8] ——, "A diffusion model for packet travel time in a random multi-hop medium," *ACM Transactions on Sensor Networks*, vol. 3, no. 2, pp. 1–19, Jun. 2007.

[9] ——, "Search in unknown random environments," *Physical Review E*, vol. 82, no. 6, p. 061112, Dec. 2010.

[10] N. Bisnik and A. A. Abouzeid, "Queuing network models for delay analysis of multihop wireless ad hoc networks," *Ad Hoc Networks*, vol. 7, no. 1, pp. 79–97, 2009.

[11] C.-F. Chiasserini and M. Garetto, "Modeling the performance of wireless sensor networks," in *Proc. INFOCOM '04*, Hong Kong, Mar. 2004, p. 231.

[12] Y.-B. Ko and N. H. Vaidya, "Location-aided routing (LAR) in mobile ad hoc networks," in *Proc. 4th ACM Annual International Conf. on Mobile Computing and Networks (MOBICOM'98)*, 1998, pp. 66–75.

[13] E. Gelenbe, "Steps towards self-aware networks," *Communications ACM*, vol. 52, no. 7, pp. 66–75, Jul. 2009.

[14] Y. Wang, S. Jain, M. Martonosi, and K. Fall, "Erasure-coding based routing for opportunistic networks," in *Proc. ACM SIGCOMM workshop on delay-tolerant networking (WDTN'05)*, Philadelphia, PA, USA, 2005, pp. 229–236.

[15] The Network Simulator (ns-2). http://www.isi.edu/nsnam/ns/.

[16] D. Bertsekas and R. Gallager, *Data Networks*, 2nd ed. NJ, USA: Prentice-Hall, Inc., 1992.

[17] L. Kleinrock, "Message delay in communication nets with storage," Ph.D. dissertation, Massachusetts Institute of Technology, 1962.

[18] P. Gupta and P. Kumar, "The capacity of wireless networks," *IEEE Transactions on Information Theory*, vol. 46, no. 2, pp. 388–404, Mar. 2000.

[19] M. Zorzi and R. R. Rao, "Geographic random forwarding (GeRaF) for ad hoc and sensor networks: Energy and latency performance," *IEEE Transactions on Mobile Computing*, vol. 2, no. 4, pp. 349–365, Oct. 2003.

[20] Y. Wang, M. Vuran, and S. Goddard, "Cross-layer analysis of the end-to-end delay distribution in wireless sensor networks," in *Proc. 30th IEEE Real-Time Systems Symposium (RTSS'09)*, Washington, D.C., USA, Dec. 2009, pp. 138–147.

[21] ——, "Stochastic analysis of energy consumption in wireless sensor networks," in *Proc. 7th IEEE Conference on Sensor Mesh and Ad Hoc Communications and Networks (SECON'10)*, Jun. 2010, pp. 1–9.

[22] C.-F. Chiasserini, R. Gaeta, M. Garetto, M. Gribaudo, D. Manini, and M. Sereno, "Fluid models for large-scale wireless sensor networks," *Performance Evaluation*, vol. 64, no. 7–8, pp. 715–736, 2007.

[23] H. Bhatia, R. B. Lenin, A. Munjal, S. Ramaswamy, and S. Srivastava, "A queuing-theoretic framework for modeling and analysis of mobility in WSNs," in *Proce. 8th Workshop on Performance Metrics for Intelligent Systems (PerMIS'08)*, Gaithersburg, Maryland, 2008, pp. 248–253.

[24] W. Whitt, "The queueing network analyzer," *The Bell System Technical Journal*, vol. 62, no. 9, pp. 2779–2815, Nov. 1983.

[25] T. Jun and C. Julien, "Delay analysis for symmetric nodes in mobile ad hoc networks," in *Proc. 4th ACM workshop on Performance monitoring and measurement*

of heterogeneous wireless and wired networks (PM2HW2N'09), Tenerife, Canary Islands, Spain, 2009, pp. 191–200.

[26] J. R. Jackson, "Jobshop-like queueing systems," *Management Science*, vol. 10, no. 1, pp. 131–142, Oct. 1963.

[27] E. Gelenbe and G. Pujolle, *Introduction to Queueing Networks.* New York, NY, USA: John Wiley & Sons, Inc., 1987 and 2000.

[28] E. Gelenbe and E. Ngai, "Adaptive random re-routing for differentiated QoS in sensor networks," *The Computer Journal*, vol. 53, no. 7, pp. 1052–1061, Sep. 2010.

[29] A. Gamal, J. Mammen, B. Prabhakar, and D. Shah, "Throughput-delay trade-off in wireless networks," in *Proc. INFOCOM '04*, Mar. 2004, pp. 464–475.

[30] I. Mabrouki, G. Froc, and X. Lagrange, "On the data delivery delay taken by random walks in wireless sensor networks," in *Proc. 5th International Conference on Quantitative Evaluation of Systems (QEST'08)*, St Malo, France, Sep. 2008, pp. 17–26.

[31] D. Dhanapala, A. Jayasumana, and Q. Han, "Performance of random routing on grid-based sensor networks," in *Proc. 6th IEEE Consumer Communications and Networking Conference (CCNC'09)*, Las Vegas, NV, USA, Jan. 2009.

[32] M. Penrose, *Random Geometric Graphs (Oxford Studies in Probability, 5).* Oxford University Press, 2003.

[33] R. Beraldi, Q. Leonardo, and B. Roberto, "Low hitting time random walks in wireless networks," *Wireless Communications and Mobile Computing*, vol. 9, pp. 719–732, May 2009.

[34] R. Beraldi, "Biased random walks in uniform wireless networks," *IEEE Transactions on Mobile Computing*, vol. 8, no. 4, pp. 500–513, Apr. 2009.

[35] S.-P. Wang and W.-J. Pei, "First passage time of multiple Brownian particles on networks with applications," *Physica A: Statistical Mechanics and its Applications*, vol. 387, no. 18, pp. 4699–4708, 2008.

[36] F. Rojo, J. Revelli, C. E. Budd, H. S. Wio, G. Oshanin, and K. Lindenberg, "Intermittent search strategies revisited: effect of the jump length and biased motion," *Journal of Physics A: Mathematical and General*, vol. 43, no. 34, p. 345001, 2010.

[37] R. Beraldi, "Random walk with long jumps for wireless ad hoc networks," *Ad Hoc Networks*, vol. 7, no. 2, pp. 294 – 306, 2009.

[38] D. Aldous and J. A. Fill, "Reversible Markov chains and random walks on graphs," http://www.stat.berkeley.edu/ aldous/RWG/book.html, 1998.

[39] L. Lovász, "Random walks on graphs: A survey," *Combinatorics, Paul Erdos is Eighty*, vol. 2, no. 1, pp. 1–46, 1993.

[40] M. Sigelle, I. Jermyn, S. Perreau, and A. Jayasuriya, "Lattice green functions and diffusion for modelling traffic routing in ad hoc networks," in *Proc. 7th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOPT'09)*, Seoul, Korea, Jun. 2009.

[41] C.-K. Chau and P. Basu, "Analysis of latency of stateless opportunistic forwarding in intermittently connected networks," *IEEE/ACM Transactions on Networking*, 2011, to appear.

[42] I. M. Macphee, M. V. Menshikov, and A. R. Wade, "Moments of exit times from wedges for non-homogeneous random walks with asymptotically zero drifts," Preprint arxiv.org/abs/0806.4561, 2010.

[43] ——, "Angular asymptotics for multi-dimensional non-homogeneous random walks with asymptotically zero drifts," *Markov Processes and Related Fields*, vol. 16, pp. 351–388, 2010.

[44] G. F. Newell, *Applications of Queueing Theory.* London: Monographs on Applied Probability and Statistics, Chapman and Hall Ltd., 1971.

[45] E. Gelenbe, "Probabilistic models of computer systems. part II: Diffusion approximations, waiting times and batch arrivals," *Acta Informatica*, vol. 12, pp. 285–303, 1979, 10.1007/BF00268317.

[46] E. Gelenbe, X. Mang, and R. Onvural, "Diffusion based call admission control in ATM," *Performance Evaluation*, vol. 27 & 28, pp. 411–436, 1996.

[47] S. Shakkottai, "Asymptotics of search strategies over a sensor network," *IEEE Transactions on Automatic Control*, vol. 50, no. 5, pp. 594–606, May 2005.

[48] T. Czachórski, K. Grochla, and F. Pekergin, "Diffusion approximation model for the distribution of packet travel time at sensor networks," in *Proc. EuroNGI Workshop on Wireless Systems and Mobility in Next Generation Internet*, ser. Lecture Notes in Computer Science, L. Cerdà-Alabern, Ed., vol. 5122. Springer Berlin / Heidelberg, 2008, pp. 10–25.

[49] D. R. Cox and H. D. Miller, *The Theory of Stochastic Processes.* Wiley, New York, 1965.

[50] G. F. Lawler, O. Schramm, and W. Werner, "Values of Brownian intersection exponents, I: Half-plane exponents," *Acta Mathematica*, vol. 187, pp. 237–273, 2001.

[51] ——, "Values of Brownian intersection exponents, II: Plane exponents," *Acta Mathematica*, vol. 187, pp. 275–308, 2001.

[52] ——, "Values of Brownian intersection exponents III: Two-sided exponents," *Annales de l'Institut Henri Poincare (B) Probability and Statistics*, vol. 38, no. 1, pp. 109 – 123, 2002.

[53] C.-f. Hsin and M. Liu, "Hitting time analysis for a class of random packet forwarding schemes in ad hoc networks," *Ad Hoc Networks*, vol. 7, no. 3, pp. 500–513, May 2009.

[54] H. Fukś, A. T. Lawniczak, and S. Volkov, "Packet delay in models of data networks," *ACM Transactions on Modeling and Computer Simulation*, vol. 11, pp. 233–250, Jul. 2001.

[55] M. Moreau, O. Bénichou, C. Loverdo, and R. Voituriez, "Dynamical and spatial disorder in an intermittent search process," *Journal of Physics A: Mathematical and Theoretical*, vol. 42, no. 43, p. 434007, 2009.

[56] G. M. Viswanathan, S. V. Buldyrev, S. Havlin, M. G. E. da Luz, E. P. Raposo, and H. E. Stanley, "Optimizing the success of random searches," *Nature*, vol. 401, no. 6756, pp. 911–914, Oct. 1999.

[57] R. Metzler, T. Koren, B. van den Broek, G. J. L. Wuite, and M. A. Lomholt, "And did he search for you, and could not find you?" *Journal of Physics A: Mathematical and Theoretical*, vol. 42, no. 43, p. 434005, 2009.

[58] D. W. Sims *et al.*, "Scaling laws of marine predator search behaviour," *Nature*, vol. 451, no. 7182, pp. 1098–1102, 2008.

[59] N. E. Humphries *et al.*, "Environmental context explains Lévy and Brownian movement patterns of marine predators," *Nature*, vol. 465, no. 7301, pp. 1066–1069, 2008.

[60] G. Viswanathan, E. Raposo, and M. da Luz, "Lévy flights and superdiffusion in the context of biological encounters and random searches," *Physics of Life Reviews*, vol. 5, no. 3, pp. 133 – 150, 2008.

[61] J. M. Kleinberg, "Navigation in a small world," *Nature*, vol. 406, no. 845, 2000.

[62] M. Franceschetti and R. Meester, "Navigation in small-world networks: A scale-free continuum model," *Journal of Applied Probability*, vol. 43, no. 4, pp. 1173=–1180, 2006.

[63] G. Li, S. D. S. Reis, A. A. Moreira, S. Havlin, H. E. Stanley, and J. S. Andrade, "Towards design principles for optimal transport networks," *Physical Review Letters*, vol. 104, no. 1, p. 018701, Jan 2010.

[64] L. Hu, A. Y. Grosberg, and R. Bruinsma, "First passage time distribution for the 1D diffusion of particles with internal degrees of freedom," *Journal of Physics A: Mathematical and Theoretical*, vol. 42, no. 43, p. 434011, 2009.

[65] T. Noguchi, T. Matsuda, and M. Yamamoto, "Performance evaluation of new multicast architecture with network coding," *IEICE Transactions on Communications*, vol. 86, pp. 1788–1795, 2002.

[66] N. Cai and R. Yeung, "Secure network coding," in *Proc. IEEE International Symposium on Information Theory (ISIT'02)*, 2002, pp. 323–.

[67] J. Vilela, L. Lima, and J. Barros, "Lightweight security for network coding," in *IEEE International Conference on Communications (ICC'08)*, May 2008, pp. 1750–1754.

[68] T. Ho, B. Leong, R. Koetter, M. Medard, M. Effros, and D. Karger, "Byzantine modification detection in multicast networks using randomized network coding," in *Proc. International Symposium on Information Theory (ISIT'04)*, Jun. 2004, pp. 144–.

[69] D. Charles, K. Jain, and K. Lauter, "Signatures for network coding," in *Proc. 40th Annual Conference on Information Sciences and Systems*, Mar. 2006, pp. 857–863.

[70] T. Ho, M. Medard, R. Koetter, D. Karger, M. Effros, J. Shi, and B. Leong, "A random linear network coding approach to multicast," *IEEE Transactions on Information Theory*, vol. 52, no. 10, pp. 4413 –4430, Oct. 2006.

[71] R. Koetter and M. Medard, "An algebraic approach to network coding," *IEEE/ACM Transactions on Networking*, vol. 11, no. 5, pp. 782–795, Oct. 2003.

[72] D. S. Lun, M. Mdard, R. Koetter, and M. Effros, "On coding for reliable communication over packet networks," *Physical Communication*, vol. 1, no. 1, pp. 3–20, 2008.

[73] P. Chou, Y. Wu, and K. Jain, "Practical network coding," in *Proc. 41st Annual Allerton Conference on Communication, Control and Computing*, Oct. 2003.

[74] B. Shrader and A. Ephremides, "A queueing model for random linear coding," in *Proc. IEEE Military Communications Conference (MILCOM'07)*, Orlando, FL, USA, Oct. 2007, pp. 1–7.

[75] D. Lun, P. Pakzad, C. Fragouli, M. Medard, and R. Koetter, "An analysis of finite-memory random linear coding on packet streams," in *Proc. 4th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks (WiOpt'06)*, Boston, MA, USA, Apr. 2006.

[76] R. Cogill and B. Shrader, "Queue length analysis for multicast: Limits of performance and achievable queue length with random linear coding," in *Proc. 47th Annual Allerton Conference on Communication, Control, and Computing*, Allerton House, UIUC, Illinois, USA, Sep. 2009, pp. 462–468.

[77] J. K. Sundararajan, D. Shah, and M. Medard, "On queueing in coded networks - queue size follows degrees of freedom," in *Proc. IEEE Information Theory Workshop on Information Theory for Wireless Networks (ITW'07)*, Bergen, Norway, Jul. 2007.

[78] ——, "ARQ for network coding," in *Proc. IEEE International Symposium on Information Theory (ISIT'08)*, Ontario, Canada, Jul. 2008, pp. 1651–1655.

[79] D. E. Lucani, M. Medard, and M. Stojanovic, "Random linear network coding for time-division duplexing: Queueing analysis," in *Proc. IEEE International Symposium on Information Theory (ISIT'09)*, Seoul, Korea, Jul. 2009, pp. 1423–1427.

[80] Y. Sagduyu and A. Ephremides, "On joint MAC and network coding in wireless ad hoc networks," *IEEE Transactions on Information Theory*, vol. 53, no. 10, pp. 3697–3713, Oct. 2007.

[81] A. Eryilmaz and D. S. Lun, "Control for inter-session network coding," in *Proc. 3rd Workshop on Network Coding, Theory and Applications (Netcod'07)*, San Diego, CA, USA, Jan. 2007.

[82] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medard, and J. Crowcroft, "XORs in the air: Practical wireless network coding," *IEEE/ACM Transactions on Networking*, vol. 16, no. 3, pp. 497–510, Jun. 2008.

[83] Y. E. Sagduyu, D. Guo, and R. Berry, "Throughput and stability of digital and analog network coding for wireless networks with single and multiple relays," in *Proc. 4th Annual International Conference on Wireless Internet (WICON'08)*, Maui, HI, USA, Nov. 2008, pp. 1–9.

[84] X. He and A. Yener, "On the energy-delay trade-off of a two-way relay network," in *Proc. 42nd Annual Conference on Information Sciences and Systems (CISS'08)*, Princeton, NJ, USA, Mar. 2008, pp. 865–870.

[85] W. Chen, K. B. Letaief, and Z. Cao, "Opportunistic network coding for wireless networks," in *Proc. IEEE International Conference on Communications (ICC'07)*, Glasgow, Scotland, 2007, pp. 4634–4639.

[86] D. Umehara, T. Hirano, S. Denno, M. Morikura, and T. Sugiyama, "Wireless network coding in slotted aloha with two-hop unbalanced traffic," *IEEE Journal on Selected Areas in Communications*, vol. 27, no. 5, pp. 647–661, Jun. 2009.

[87] J. Goseling, R. Boucherie, and J.-K. van Ommeren, "Energy consumption in coded queues for wireless information exchange," in *Proc. Workshop on Network Coding, Theory, and Applications (NetCod'09)*, Jun. 2009, pp. 30–35.

[88] E. Gelenbe, "Queueing networks with negative and positive customers," *Journal of applied probability*, vol. 28, pp. 656–663, 1991.

[89] ——, "G-networks with instantaneous customer movement," *Journal of applied probability*, vol. 30, no. 3, pp. 742–748, 1993.

[90] Y. Sagduyu and A. Ephremides, "Cross-layer optimization of MAC and network coding in wireless queueing tandem networks," *IEEE Transactions on Information Theory*, vol. 54, no. 2, pp. 554–571, Feb. 2008.

[91] J. M. Harrison, "Assembly-like queues," *Journal of Applied Probability*, vol. 10, no. 2, pp. 354–367, June 1973.

[92] E. H. Lipper and B. Sengupta, "Assembly-like queues with finite capacity: bounds, asymptotics and approximations," *Queueing Systems: Theory and Applications*, vol. 1, no. 1, pp. 67–83, 1986.

[93] W. J. Hopp and J. T. Simon, "Bounds and heuristics for assembly-like queues," *Queueing Systems: Theory and Applications*, vol. 4, no. 2, pp. 137–155, Jun. 1989.

[94] U. N. Bhat, "Finite capacity assembly like queues," *Queueing Systems: Theory and Applications*, vol. 1, no. 1, pp. 85–101, 1986.

[95] F. Bonomi, "An approximate analysis for a class of assembly-like queues," *Queueing Systems: Theory and Applications*, vol. 1, no. 3, pp. 289–309, 1987.

[96] D. G. Kendall, "Some problems in the theory of queues," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 13, no. 2, pp. 151–185, 1951.

[97] Y. Ma, W. Li, P. Fan, and X. Liu, "Queuing model and delay analysis on network coding," in *Proc. IEEE International Symposium on Communications and Information Technology (ISCIT'05)*, Beijing, China, Oct. 2005, pp. 112–115.

[98] O. H. Abdelrahman and E. Gelenbe, "Queueing performance under network coding," in *Proc. IEEE Information Theory Workshop (ITW'09)*, Volos, Greece, 10–12 June 2009, pp. 135–139.

[99] S. Chieochan, E. Hossain, T. Issariyakul, and D. Niyato, "Opportunistic network coding and dynamic buffer allocation in a wireless butterfly network," in *Proc. Globecom'09*, Honolulu, HI, USA, Dec. 2009.

[100] P. Parag and J.-F. Chamberland, "Queueing analysis of a butterfly network," in *Proc. IEEE International Symposium on Information Theory (ISIT'08)*, Ontario, Canada, Jul. 2008, pp. 672–676.

[101] A. Mahmino, J. Lacan, and C. Fraboul, "Guaranteed packet delays with network coding," in *Proc. 5th IEEE Annual Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON'08)*, San Francisco, CA, USA, June 2008, pp. 1–6.

[102] K. Wu, W. Jia, Y. Yuan, and Y. Jiang, "Performance modeling of stochastic networks with network coding," in *Proc. Workshop on Network Coding, Theory, and Applications (NetCod'09)*, Lausanne, Switzerland, Jun. 2009, pp. 6–11.

[103] A. Eryilmaz, A. Ozdaglar, M. Medard, and E. Ahmed, "On the delay and throughput gains of coding in unreliable networks," *IEEE Transactions on Information Theory*, vol. 54, no. 12, pp. 5511–5524, Dec. 2008.

[104] R. Cogill, B. Shrader, and A. Ephremides, "Stability analysis of random linear coding across multicast sessions," in *Proc. IEEE International Symposium on Information Theory (ISIT'08)*, Ontario, Canada, Jul. 2008, pp. 31–35.

[105] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," *IEEE Transactions on Automatic Control*, vol. 37, no. 12, pp. 1936–1948, Dec. 1992.

[106] F. Baccelli, W. A. Massey, and D. Towsley, "Acyclic fork-join queuing networks," *Journal of the ACM*, vol. 36, no. 3, pp. 615–642, 1989.

[107] E. Gelenbe and R. R. Muntz, "Probabilistic models of computer systems-part I (exact results)," *Acta Informatica*, vol. 7, pp. 35–60, 1976.

[108] J. Medhi, *Stochastic Processes*, 2nd ed. New Delhi: Wiley Eastern Limited, 1994.

[109] T. Luan, L. Cai, and X. Shen, "Impact of network dynamics on user's video quality: Analytical framework and QoS provision," *IEEE Transactions on Multimedia*, vol. 12, no. 1, pp. 64 –78, Jan. 2010.

[110] P. Chandran, "Inverse Laplace transforms of a class of non-rational fractional functions," *International Journal of Mathematical Education in Science and Technology*, vol. 32, no. 1, pp. 136–140, Jan. 2001.

[111] A. Di Crescenzo, V. Giorno, B. Krishna Kumar, and A. Nobile, "A double-ended queue with catastrophes and repairs, and a jump-diffusion approximation," *Methodology and Computing in Applied Probability*, pp. 1–18, 2011.

[112] H. A. David, *Order Statistics*, 1st ed., ser. Wiley series in probability and mathematical statistics. New York: John Wiley & Sons, Inc., 1970.

[113] R. J. Serfling, *Approximation Theorems of Mathematical Statistics*, ser. Wiley series in probability and mathematical statistics. New York: John Wiley & Sons, Inc., 1980.

[114] E. Gelenbe, "Travel time and energy in wireless networks with duplicate packets," in *Proc. 5th International Symposium on Modeling and Optimization in Mobile, Ad Hoc and Wireless Networks and Workshops (WiOpt'07)*, Limassol, Cyprus, Apr. 2007.

[115] K. J. Hollenbeck, "INVLAP.M: a MATLAB function for numerical inversion of Laplace transforms by the de Hoog algorithm," http://cambridge.org/us/engineering/author/nellisandklein/software.htm, 1998.

[116] F. R. de Hoog, J. H. Knight, and A. N. Stokes, "An improved method for numerical inversion of Laplace transforms," *SIAM Journal on Scientific and Statistical Computing*, vol. 3, no. 3, pp. 357–366, 1982.

[117] P. P. Valkó and J. Abate, "Numerical inversion of 2-D Laplace transforms applied to fractional diffusion equations," *Applied Numerical Mathematics*, vol. 53, pp. 73–88, Apr. 2005.

[118] J. Abate and P. P. Valkó, "Multi-precision Laplace transform inversion," *International Journal for Numerical Methods in Engineering*, vol. 60, pp. 979–993, 2004.

[119] J. S. Marron and M. P. Wand, "Exact mean integrated squared error," *The Annals of Statistics*, vol. 20, no. 2, pp. 712–736, 1992.

[120] D. V. Lindley, "The theory of queues with a single server," *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 48, pp. 277–289, 1952.

[121] E. Gelenbe and A. Ghanwani, "Approximate analysis of coupled queueing in ATM networks," *IEEE Communications Letters*, vol. 3, no. 2, pp. 31–33, Feb. 1999.

[122] E. Gelenbe and I. Mitrani, *Analysis and Synthesis of Computer Systems*, 2nd ed. London, UK: Imperial College Press, 2010.

[123] E. Gelenbe and R. Iasnogorodski, "A queue with server of walking type (autonomous service)," *Annales de l'institut Henri Poincar (B) Probabilits et Statistiques*, vol. 16, no. 1, pp. 63–73, 1980.

[124] O. H. Abdelrahman and E. Gelenbe, "Approximate analysis of a round robin scheduling scheme for network coding," in *Proc. 6th European Performance Engineering Workshop (EPEW'09)*, ser. Lecture Notes in Computer Science, vol. 5652. London, UK: Springer-Verlag, Berlin Heidelberg, 9–10 July 2009, pp. 212–217.

[125] E. Gelenbe, "On approximate computer system models," *Journal of the ACM*, vol. 22, pp. 261–269, April 1975.

[126] J. Aweya, "IP router architectures: an overview," *International Journal of Communication Systems*, vol. 14, no. 5, pp. 447–475, Jun. 2001.

[127] G. Fayolle and R. Iasnogorodski, "Two coupled processors: The reduction to a Riemann-Hilbert problem," *Probability Theory and Related Fields*, vol. 47, no. 3, pp. 325–351, Jan. 1979.

[128] J. W. Cohen and O. J. Boxma, *Boundary Value Problems in Queueing System Analysis.* Amsterdam ; New York: North-Holland Pub. Co.; Elsevier Science Pub. Co., 1983.

[129] J. S. H. van Leeuwaarden and J. A. C. Resing, "A tandem queue with coupled processors: Computational issues," *Queueing Systems*, vol. 51, no. 1, pp. 29–52, Jan. 1979.

[130] A. M. Haghighi and D. P. Mishev, "Analysis of a two-node task-splitting feedback tandem queue with infinite buffers by functional equation," *International Journal of Mathematics in Operational Research*, vol. 1, pp. 246–277, Jan. 2009.

[131] K. Fall, "A delay-tolerant network architecture for challenged internets," in *Proc. SIGCOMM '03*, Karlsruhe, Germany, 2003, pp. 27–34.

[132] L. Pelusi, A. Passarella, and M. Conti, "Opportunistic networking: data forwarding in disconnected mobile ad hoc networks," *IEEE Communications Magazine*, vol. 44, no. 11, pp. 134–141, Nov. 2006.

[133] R. D. Poor, "Gradient routing in ad hoc networks," MIT Media Lab, Tech. Rep., 2000, http://www.media.mit.edu/pia/Research/ESP/texts/poorieeepaper.pdf.

[134] J. Faruque, K. Psounis, and A. Helmy, "Analysis of gradient-based routing protocols in sensor networks," in *Distributed Computing in Sensor Systems*, ser. Lecture Notes in Computer Science. Springer Berlin / Heidelberg, 2005, vol. 3560, pp. 466–466.

[135] Q. Lv, P. Cao, E. Cohen, K. Li, and S. Shenker, "Search and replication in unstructured peer-to-peer networks," in *Proc. 16th international conference on Supercomputing (ICS '02)*, New York, New York, USA, 2002, pp. 84–95.

[136] E. Gelenbe and Y. Cao, "Autonomous search for mines," *European Journal of Operational Research*, vol. 108, no. 2, pp. 319–333, Jul. 1998.

[137] E. Gelenbe and G. Hébrail, "A probability model of uncertainty in data bases," in *Proc. 2nd International Conference on Data Engineering (ICDE '86)*, LA, CA, USA, Feb. 1986, pp. 328–333.

[138] J. Medhi, *Stochastic Models in Queueing Theory*, 2nd ed. Burlington, MA: Academic Press, 2003.

[139] H. Risken, *The Fokker-Planck Equation: Methods of Solutions and Applications*, 2nd ed., ser. Springer Series in Synergetics. Berlin Heidelberg: Springer-Verlag, 1996.

[140] K. Jacobs, *Stochastic Processes for Physicists : Understanding Noisy Systems.* Cambridge University Press, 2010.