A study of self-similar traffic generation for ATM networks

Hung-Ming Chen

UNIVERSITY OF DURHAM

SCHOOL OF ENGINEERING

A Study of Self-Similar Traffic Generation for ATM Networks

Hung-Ming Chen

A THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS OF THE COUNCIL OF THE UNIVERSITY OF DURHAM FOR THE DEGREE OF DOCTOR OF PHILOSOPHY (PH.D.).

DECEMBER 1997

Declaration

I hereby declare that this thesis is a record of work undertaken by myself, that it has not been the subject of any previous appplication for a degree, and that all sources of information have been duly acknowledged.

.....

Hung-Ming Chen, December 1997

Abstract

This thesis discusses the efficient and accurate generation of self-similar traffic for ATM networks. ATM networks have been developed to carry multiple service categories. Since the traffic on a number of existing networks is bursty, much research focuses on how to capture the characteristics of traffic to reduce the impact of burstiness. Conventional traffic models do not represent the characteristics of burstiness well, but self-similar traffic models provide a closer approximation. Self-similar traffic models have two fundamental properties, long-range dependence and infinite variance, which have been found in a large number of measurements of real traffic. Therefore, generation of self-similar traffic is vital for the accurate simulation of ATM networks.

The main starting point for self-similar traffic generation is the production of fractional Brownian motion (FBM) or fractional Gaussian noise (FGN). In this thesis six algorithms are brought together so that their efficiency and accuracy can be assessed. It is shown that the discrete FGN (dFGN) algorithm and the Weierstrass-Mandelbrot (WM) function are the best in terms of accuracy while the random midpoint displacement (RMD) algorithm, successive random addition (SRA) algorithm, and the WM function are superior in terms of efficiency. Three hybrid approaches are suggested to overcome the inefficiency or inaccuracy of the six algorithms. The combination of the dFGN and RMD algorithm was found to be the best in that it can generate accurate samples efficiently and on-the-fly. After generating FBM sample traces, a further transformation needs to be conducted with either the marginal distribution model or the storage model to produce self-similar traffic. The storage model is a better transformation because it provides a more rigorous mathematical derivation and interpretation of physical meaning.

The suitability of using selected Hurst estimators, the rescaled adjusted range (R/S) statistic, the variance-time (VT) plot, and Whittle's approximate maximum likelihood estimator (MLE), is also covered. Whittle's MLE is the better estimator, the R/S statistic can only be used as a reference, and the VT plot might misrepresent the actual Hurst value.

An improved method for the generation of self-similar traces and their conversion to traffic has been proposed. This, combined with the identification of reliable methods for the estimators of the Hurst parameter, significantly advances the use of self-similar traffic models in ATM network simulation.

To Ruby and Beatrice –

For making it possible
and worthwhile

– with all my love

ACKNOWLEDGEMENTS

Firstly, I would like to thank Prof. P. Mars and Mr. J. E. Mellor for all their advice and support throughout the last three years, and their provision of the excellent laboratory facilities which have been used in this work. Without their belief, support, and continuous encouragement I would never arrive at this point. In particular, I am very grateful to Mr. J. E. Mellor for his extreme patience in dealing with almost endless spelling and grammar checks.

For their financial support during my study, I am most appreciative to the Committee of Overseas Research Students Awards Scheme and the Research Awards Sub-Committee at University of Durham for the provision of an ORS and University Studentship awards.

I am extremely grateful to my colleagues (and ex-colleagues) at the Centre for Telecommunication Networks, for all their comments and advice. However, there are some more persons who deserve special mention. Thanks must go to Dr. J. Yao, former research staff of the BATMAN project, for his discussion and suggestion on some research subjects. Thanks must also go to J. Reeve, M. Ritch, and J. Hall, for their advice and grammar check. I am also grateful to Prof. M. Crovella, University of Boston, MA, for providing S-Plus source code of Whittle's approximate MLE and to Dr. D. Wooff, Department of Mathematical Science, University of Durham, for providing the S-Plus package.

Finally, I am deeply indebted to my wife Ruby and my daughter Beatrice (Ying-Ju). Without their support and encouragement, I would have stayed in Taiwan and kept doing programming all day long.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

ATM	Asynchronous Transfer Mode
CBR	Constant Bit Rate
dFGN	discrete Frcation Gaussian Noise
FARIMA	Fractional AutoRegression Moving Average
FBM	Fractional Brownian Motion
fFGN	fast Fractiona Gaussian Noise
FGN	Fractional Gaussian Noise
i.i.d.	identical independently distributed
LRD	Long-Range Dependence/Dependent
NRM	Network Resource Management
nrt-VBR	non-real-time Variable Bit Rate
QoS	Quality of Service
rt-VBR	real-time Variable Bit Rate
RMD	Random Midpoint Displacement
R/S	Rescaled Adjusted Range
SRA	Successive Random Addition
WM	Weierstrass-Mandelbrot
VT	Variance-Time

PUBLICATIONS

[1] H.-M. F. Chen, Z. Tsai, J. Mellor, and P. Mars, "Performance analysis of intelligent network with two inter-connected service centres," in *Third International Workshop on Queueing Network with Finite Capacity* (D. Kouvatsos and Y. Dallery, eds.), (West Yorkshire, UK), pp. 32/1–32/7, July 1995.

[2] H.-M. F. Chen, Z. Tsai, J. Mellor, and P. Mars, "Alternative routing strategy in a private network to provide intelligent network service," in *Second Communication Networks Symposium* (J. M. Senior and R. A. Cryan, eds.), (Manchester, UK), pp. 238–241, July 1995.

[3] H.-M. F. Chen, Z. Tsai, J. Mellor, and P. Mars, "IPP overflow traffic on a two-stage queueing network," in *11th UK Performance Workshop for Computer and Telecommunications Systems*, (Liverpool, UK), Sept. 1995.

[4] J. Mellor and H.-M. F. Chen, "Broadband asynchronous transfer multimedia access node," in *Proc. of IEEE 2nd Communication Networks Sympoium*, (Manchester), pp. 187–190, July 1995.

[5] H.-M. F. Chen, J. Mellor, and P. Mars, "Comparisons of simulation algorithms for self-similar traffic models," in *Proc. 13th IEE UK Teletraffic Symposium*, (Glasgow, U.K.), Mar. 1996.

[6] H.-M. F. Chen, J. Mellor, and P. Mars, "On burstiness of self-similar traffic models," in *ATM, Network and LANs* (D. W. Faulkner and A. L. Harmer, eds.), pp. 146–153, Amsterdam: IOS Press, 1996.

[7] H.-M. F. Chen, J. Mellor, and P. Mars, "Bursty measurement of Poisson and self-similar traffic," in *Third Communication Symposium*, pp. 168–172, July 1996.

[8] J.-G. Yao, J. Mellor, M. Ritch, and H.-M. F. Chen, "Architecture and control mechanism of a high-speed local area ATM switch node," in *ATM, Networks and LANs* (D. W. Faulkner and A. L. Harmer, eds.), pp. 154–161, Amsterdam: IOS Press, 1996.

[9] H.-M. F. Chen, J. Mellor, and P. Mars, "A hybrid approach for generating fractional Brownian motion," in *Proc. GLOBECOM 96*, vol. 1, (London), pp. 591–595, 1996.

[10] H.-M. F. Chen, J. Mellor, and P. Mars, "Evaluations of fractional Brownian motion and fractional Gaussian noise generating algorithms," in *Proc. of 5th IFIP Workshop on Performance Modelling and Evaluation of ATM Network*, (Bradford, Uk), pp. 26/1 – 26/11, July 1997.

[11] J.-G. Yao, H.-M. F. Chen, J. Mellor, M. Ritch, and P. Mars, "Simulation of cell scheduling algorithms using conventional and self-similar traffic models," in *Proc. of 5th IFIP Workshop on Performance Modelling and Evaluation of ATM Network*, (Bradford, Uk), pp. 41/1 – 41/10, July 1997.

Chapter 1

Introduction

The rapid growth of Internet and Intranet usage has led to an increasing number of multimedia applications being integrated into existing networks. This information is no longer restricted to local area networks (LAN), but flows throughout the world. Multimedia applications generate bursty connectionless data that is very different from the telephone traffic in conventional telecommunication networks. To integrate this connectionless data into existing telecommunication networks, Integrated Service Digital Network (ISDN) and its advanced version, Broadband Integrated Service Digital network (B-ISDN), have been proposed [Sta92].

ISDN and B-ISDN are unified network technologies and networking techniques that are necessary in order to provide transmission and switching for diverse applications, with varying traffic characteristics, over interconnected networks. Synchronous Optical NETwork (SONET) and Asynchronous Transfer Mode (ATM) are two technological developments which have helped the implementation of such networks. The former provides very high speed transmission systems while the latter concerns issues of the utilization of networks and of the quality of services (QoSs).

SONET provides the standard for interconnection of heterogeneous devices that may use optical interfaces. One of the main requirements of SONET is to prepare for the future sophisticated service offerings and support the broadband ISDN ATM transmission technique [Sta92]. The main delay occurs in ATM switches rather than in the transmission channel. Moreover, all operation, administration, and maintenance of B-ISDN are based on the design and implementation of management functions of the ATM protocol. The major effort of B-ISDN is focused on the implementation of ATM management functions [IT93a, Onv95].

1.1 ATM Networks and Control Issues

ATM is a method of statistical multiplexing in which the units of information are fixed size entities called *cells*. The switching method is referred to as *cell switching*. Because small fixed-size calls can be used as a common information unit among voice, data, video, and other services, ATM is considered to be able to carry multiple-type traffic, which satisfies the objectives of a high-speed network.

Supported by the cell-based switching, the behavior of traffic in an ATM network can be considered

on several levels: *network* (or *subscription*) *level*; *call level*; *burst level*; and *cell level* as shown in Fig. 1.1 [CPL+94, De 95, GSV95, Gin96]. At the network level, the concern is with the connections and disconnections of traffic sources, as well as the number of calls. At the call level, a call or session lasts for the duration of the connection between the end users. A call in turn can be partitioned into a sequence of alternate burst and silent periods. During the burst period, a stream of ATM cells is emitted at regular intervals, while no cells are launched during the silent periods. An important attribute of each level is its time scale which is governed by the mean interarrival time of entities at that level during an active period in the upper level.

The characteristics of the traffic in ATM networks require to be identified. International Telecommunication Union - Telecommunication Standardization Sector (ITU-T) classifies traffic into four main classes [IT93b]: constant bit rate (CBR) service; variable bit rate (VBR) service; connection-oriented service, and connectionless data transfer. However, this classification does not satisfy the demand for capturing the characteristics of traffic from different applications, especially the VBR traffic. Recently the ATM Forum traffic management working group has completed the TM4.0 specification, according to which five service categories have been identified [For96, Gin96, PP97]. The service categories are distinguished as being either real-time or non-real-time. The real-time traffic is closely related to the original B-ISDN service classes, therefore they are easier to model. Non-real-time traffic more closely reflects the majority data transfer and LAN interconnection requirements. The non-real-time traffic fluctuates and makes the design, operation, maintenance, and management of ATM networks very difficult.

The real-time and non-real-time traffic in an ATM network may be bursty. A bursty source generates cells at a peak rate during its active periods and possibly no cells during its inactivity. Since an ATM network can support a large number of bursty traffic sources, statistical multiplexing can be used to gain bandwidth efficiency, allowing more sources to share the bandwidth. However, if a large number of bursty sources are active simultaneously, network congestion can occur. To provide the end-to-end Quality of Service (QoS) for bursty traffic is one of the most important and challenging issues in the design of broadband networks.

Such different classes of traffic may require deterministic or statistical bounds on various QoS parameters, such as throughput, cell loss, cell delay, and jitter. These parameters represent performance objectives expected from the system for the duration of a connection. The QoS which includes call-blocking probability, cell loss rate, cell delay, and cell-delay variations, is used as a measure of performance of these control functions. In the ATM forum specification six QoS parameters are identified to fit a network performance objective [For96]. Half of them are required to negotiate between the end-systems and the networks, while the others are not necessary.

QoS is one of the main reasons for controlling and monitoring network traffic flow. A network can be decomposed into physical, control, and transmission components or layers. Control layers involve multiplexing, demultiplexing, traffic control, congestion control, and error control. A framework for managing and controlling traffic and congestion in ATM networks is formed by the following functions or their combinations: Network Resource Management (NRM); Connection Admission Control (CAC); Feedback control; Usage/network parameter control (UPC/NPC); and priority control [IT93c]. Other means of traffic and congestion control have been made in the Traffic Management 4.0 document:

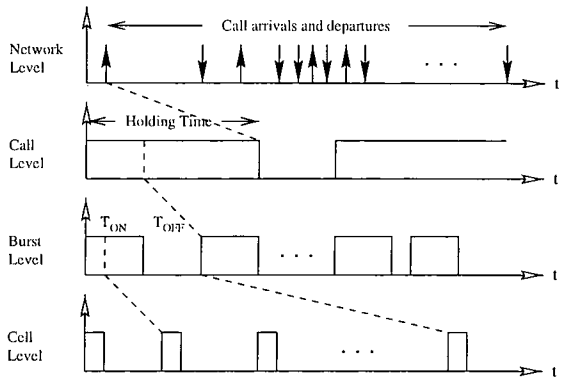


Figure 1.1: ATM call process decomposition.

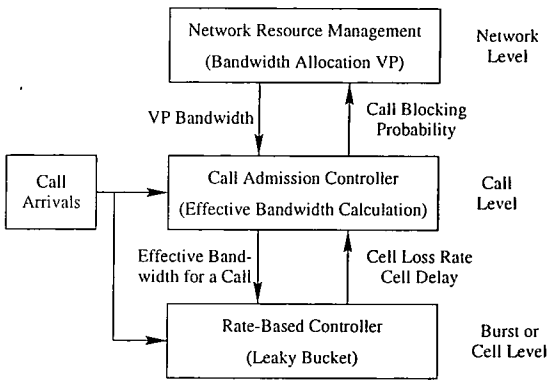


Figure 1.2: Layered traffic control strategy.

Connection Admission Control; Usage parameter Control; Selective Cell Discarding; Traffic Shaping; Explicit Forward Congestion Indication; Resource Management using Virtual Paths; Frame Discard; and Available Bit Rate (ABR) Flow Control [For96].

These traffic control mechanisms can be divided into three categories: NRM; CAC; and Rate-Based Control (RBC) [GSV95]. The traffic control strategy is illustrated in Fig. 1.2. NRM involves virtual path (VP) bandwidth and buffer allocation. CAC includes routing and virtual channel (VC) bandwidth allocation to determine the rejection or acceptance of calls. RBC which covers leaky bucket, window or credit based control, priority control, traffic shaping, and UPC/NPC, is therefore concerned with all the control schemes for preventing or reacting to congestion.

1.2 Motivation and Objective

Engineering and modeling are the two main technologies for designing, evaluating, and managing telecommunication networks [HTL92]. In order to meet performance objectives and to utilize the capital deployed in the network efficiently, these engineering guidelines require to be based on a solid foundation of knowledge about the network and its traffic. Similarly, both analytical and simulation models are currently used to compare alternative implementation strategies for new services and to evaluate network performance. Again, in order to assess network strategies, integrity, and performance reliability, models should be based on realistic assumptions about the network and its traffic.

Modeling is the means to present the characteristics of the traffic, through two types of understanding: concrete causal understanding and abstract statistical understanding [Nor95b]. The former aggregates several elementary models to build a total traffic model, while the latter uses a stochastic model with some essential statistical properties to represent the total traffic. These two understandings are required to represent the characteristics of real traffic.

Non-real-time traffic is difficult to handle because of its burstiness. The integration of different types of traffic into a single network is a challenge for the design of, the operating of, the administration of, and the maintenance of the network. Appropriate control schemes must be implemented to meet the users' QoS requirements and to optimize the utilization of network resources. Traffic modeling is a fundamental issue for designing and evaluating the performance of control mechanisms.

In order to provide a high quality of service and to maximize the utilization of networks, a number of traffic models are used to attempt to capture the characteristics of different types of service. Most of the traffic models are based on Poisson and Markov processes, in which the current state is derived from the previous state. However, a great number of traffic measurements indicate that sampling data exhibits a very strong correlation with previous sampling data, therefore the conventional traffic models are difficult to model the traffic.

Recently, self-similar traffic models, which are initiated by the observation of the Ethernet traffic measurement, have been found suitable for modeling the strong correlated phenomenon. These models are based on the self-similar stochastic processes whose second-order stochastic processes exhibit long-range dependence (LRD). The models use an addition parameter, the Hurst parameter, to identify the degree of correlation whereas the other models require a greater number of parameters as the degree of dependence increases. Not only is self-similar nature discovered in the Ethernet traffic, but it also exists in VBR video and World Wide Web (WWW) transactions. Therefore, self-similar traffic models might be the better method to present all categories of traffic and still use only a small number of parameters.

After understanding self-similar traffic models, its mathematical derivation can be used to analyze the queueing behavior of ATM networks and the generated self-similar traffic traces can be used for ATM network simulations. Therefore, the study of self-similar traffic modeling will be a key foundation into further investigation of ATM networks, in particular traffic control and congestion control. The flow chart of the reasoning behind the use of self-similar traffic models is depicted in Fig. 1.3.

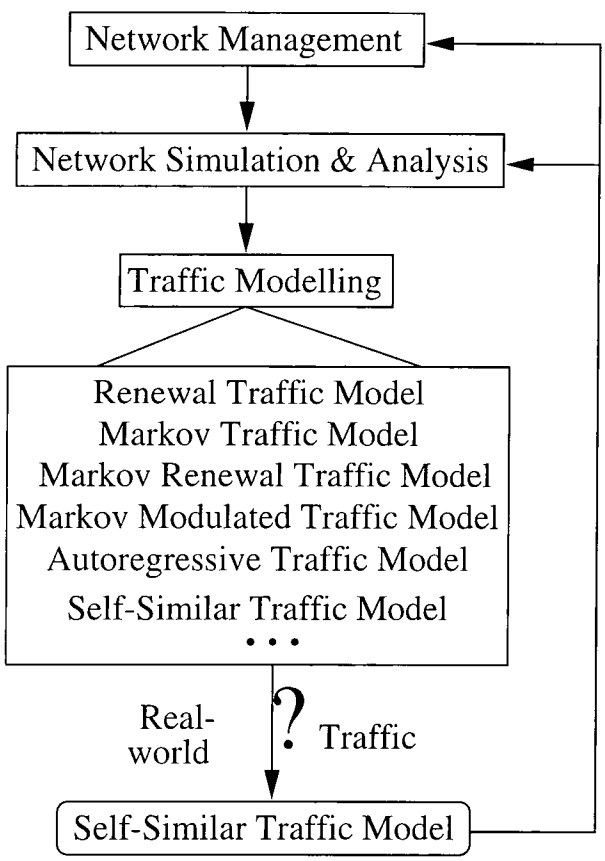


Figure 1.3: Motivation.

Simulation and analysis are the two key disciplines used in the research and development of control mechanisms for ATM networks. Traffic modeling is fundamental to both and has been chosen as the main area of research. Of the traffic models, self-similar traffic models are chosen for further study because they are believed to fit real traffic much better than others.

The thesis documents work which has been undertaken in understanding the mathematical foundation of self-similar traffic. Therefore, the existing algorithms for generating FBM or FGN samples are evaluated, three novel hybrid approaches to produce fractal samples are introduced, and finally the models to transfer the FBM processes to self-similar counting processes are appraised. A thorough study of these issues is required before self-similar traffic can be generated quickly and accurately for further use.

1.3 Outline of the Thesis

The main body of the thesis is divided into 5 chapters. A critical review of the state-of-the-art of traffic modeling in telecommunication networks is covered in Chapter 2. In addition, several traffic measurements are introduced to support the use of self-similar traffic models. Thereafter, six criteria are used to examine self-similar traffic models.

The self-similar stochastic process and its incremental process, the second-order self-similar stochastic process, are the theoretical basis for self-similar traffic modeling. Before introducing the self-similar stochastic process, its deterministic version, fractals, and the properties of self-similarity are introduced. Then, the definition and properties of a random version of self-similarity are discussed. Three models for self-similar phenomena, fractional Brownian motion (FBM) processes, fractional autoregressive integer moving average (FARIMA) processes, and chaotic maps are explained and evaluated. Three Hurst estimators, the R/S statistic, the variance-time (VT) plot, and Whittle's maximum likelihood estimator (MLE), together with a graphical technique for goodness-of-fit, the Q-Q plot, are introduced before using them to examine the quality of fractal samples. All this material is covered in Chapter 3.

After building the basis of the mathematical material for self-similar processes, Chapter 4 introduces six algorithms for generating FBM or its second order process, fractional Gaussian noise (FGN). They are the discrete FGN (dFGN) algorithm, the fast FGN (fFGN) algorithm, the random midpoint displacement (RMD) algorithm, the successive random addition (SRA) algorithm, Whittle's approximate approach, and Weierstrass-Mandelbrot (WM) function. In addition, all these algorithms are appraised by the estimators and the Q-Q plot.

Based on the algorithms introduced in Chapter 4, three hybrid approaches are introduced in Chapter 5. The RMD, SRA algorithms and the WM function are incorporated with the dFGN algorithm to generate FBM or FGN samples on-the-fly, to attempt to reduce the computational effort of the dFGN algorithm and to attempt to improve the accuracy of the RMD and SRA algorithms. The estimators and Q-Q plot are used to examine these approaches and find out which combination scheme of these hybrid algorithms is the most suitable.

After generating the FBM and FGN traces, the transformation from self-similar sample traces to self-similar traffic is investigated in Chapter 6. In this chapter, two different models, the marginal distribution and storage models, are used to generate self-similar traffic traces. Then, the estimators are used again

to examine the models.

Finally, the conclusions of the thesis and further work are stated in Chapter 7. A more complete descriptive survey of traffic models is covered in Appendix A. In Appendix B, the derivation of the dFGN+WM algorithm is discussed. Appendix C is a simple application of the fractal traces in which several definitions of burstiness are used to measure the degree of burstiness of the fractal streams. The last part, Appendix D, tabulates the computational time and estimated Hurst values of algorithms in Chapter 5 and models in Chapter 6.

Chapter 2

Traffic Models for ATM Networks

ATM is designed as a fixed cell length and non-synchronous transmission scheme. The design allows for a very flexible transmission of information at every CBR or VBR up to the link rate. It is important for network engineering and modeling to know the relevant statistical characteristics of a large range of ATM sources.

In ATM networks voice, data, and video are the three main types of traffic [CPL+94,Onv95]. Because of the variety of cell streams generated by various ATM sources, several proposals for traffic models have emerged with the emphasis on simplicity, realism, and accuracy. Often these traffic models have been used to evaluate network performance or to generate virtual traffic for network simulations. Self-similar traffic models are considered to be the most appropriate ones in this thesis.

This chapter justifies the chosen self-similar traffic models which form the focus for the following research. The main body consists of three parts. Sec.2.1 gives a brief overview of the traffic models, Sec. 2.2 outlines several measurements of real traffic, and Sec. 2.3 is an examination of self-similar traffic models.

2.1 A Brief Review of Traffic Models

Analysis and simulation are the two major techniques used to understand the behavior of a network. Traffic models are the fundamental basis of both techniques. Many traffic models have been proposed to capture the characteristics of different traffic services. This state-of-the-art survey is based on [CPL+94] and [FM94] with a number of additions. Fig. 2.1 illustrates the set diagram of Markov-based processes and is based on the relationship of the stochastic processes characterized by different kinds of dependency among their random variables. The diagram is based on [Kle75], but with the newly developed traffic models added. A fuller survey of traffic models with more complete references can be found in Appendix A.

Point process, counting process, and interarrival time process are three equivalent processes by which a traffic source in continuous time sense can be model-led and described. The point process concerns the time instances when arrivals occur. The counting process deals with the cumulative number of arrivals in a specific time. As its name implies, the interarrival time process is a non-negative sequence that

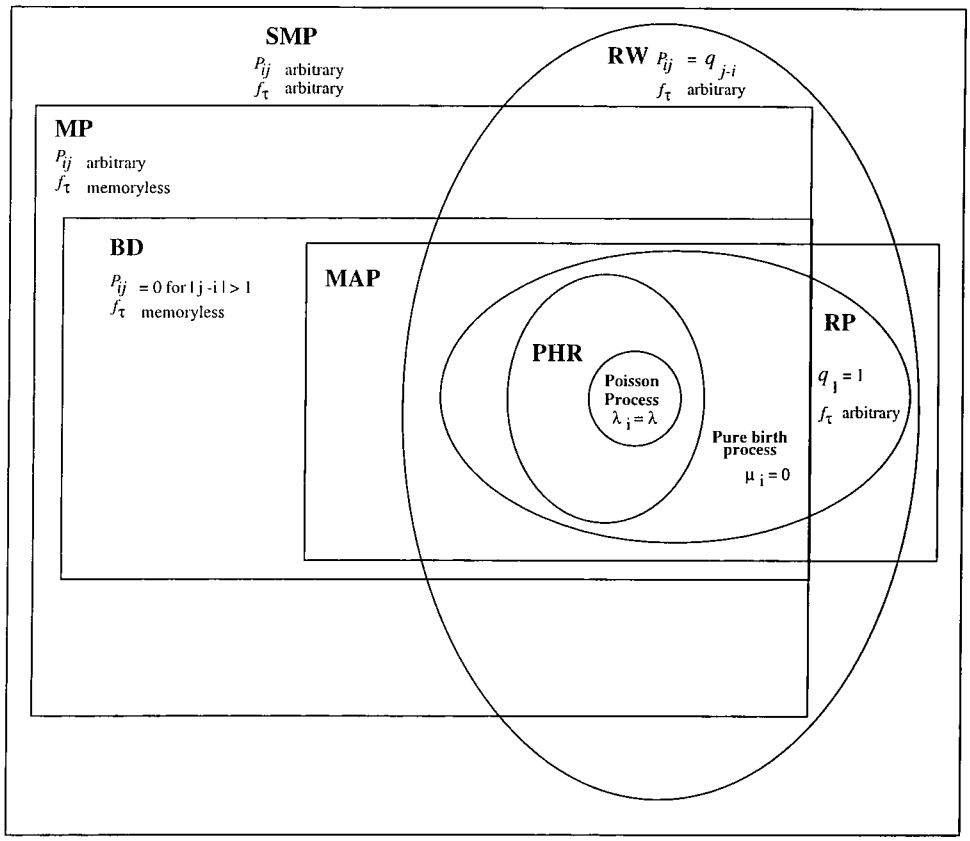


Figure 2.1: Relationships among the interesting random processes. **SMP**: Semi-Markov process; **MP**: Markov process; **RW**: Random walk; **RP**: Renewal process; **BD**: Birth-Death process; **PHR**: Phase-Type Renewal process; **MAP**: Markovian Arrival process.

records the length of time intervals separating a specific arrival from a previous one. Discrete-time traffic processes are applied when the time is slotted.

Poisson and Bernoulli processes are the two simplest renewal models in continuous time and discrete time, respectively. The regeneration property makes the renewal processes popular because the mathematical expressions become simpler and will usually hold in more general situations.

The general modulated deterministic process based on finite state machines describes the behavior of the cell generation [CPL+94]. The ON/OFF model is a more popular model which regularly incorporates other models to describe the behavior of traffic sources.

Markov-modulated models use an auxiliary Markov process which is evolving with time. A Markov-modulate Poisson process (MMPP) is the most commonly used Markov-modulated model [Onv95, Sai94]. It has been applied to voice and video traffic. An interrupted Poisson process (IPP) is the simplest MMPP in which only two states are used. Often, an IPP regards as an ON/OFF model with two independent exponential distributions to alternatively turn on and turn off a queueing system. In the modulated state, with a state transition rather than the state itself, the transition-modulated processes are governed by a transition matrix.

Markov and Markov-renewal processes are used to reflect the characteristics of burstiness because their interarrival time process is correlated. Phase-type (PH) renewal processes have an absorbing state and reach this state within finite value [Neu79, Sai94]. A variation of a phase-type process, a Markovian

arrival process (MAP), uses a probability vector instead of the absorbing state [LMHN90, Sai94]. Semi-Markov processes release the restriction of Markov process in which the state must transfer to another state from the current state [CPL+94, Sai94].

Different from the Poisson- and Markov-based processes, fluid flow models view traffic as a stream of fluid, characterized by a flow rate [CPL+94, GSV95]. These models are restricted to the cases where the individual units are very small in relation to a chosen time scale. This concept is important not only because of its simplicity but also it's benefits in simulations. The fundamental assumption of the fluid-flow models is that the sources are bursty with the ON/OFF type. An extension of the fundamental model is the multi-minisources model in which a single fluid-flow model aggregates many identical independent ON/OFF sources. Moreover, if there are two classes of ON/OFF sources, the superposition of multi-minisources can form a two dimensional Markov chain.

Originally, diffusion processes were used to model the state space of Brownian motion. The discontinuous arrival and service processes in queueing systems can be approximately derived as diffusion processes. Reflection and absorption or jump are the two boundary conditions that have been proposed to model cell streams [RK94, GMF96, GMO97].

Autoregression is a technique developed to predict and estimate time series [SMRA89, Onv95]. Therefore, an autoregressive (AR) model has been proposed to predict the next samples based on current and previous samples in a time series. Simple autoregressive and autoregressive moving average (ARMA) models have been proposed to model video traffic [GCMOO91]. Transform-expand-sample (TES) models have been proposed to fit both marginal and autocorrelations of empirical records for video traffic [FM94].

The packet-train model was proposed to model the measurements of token ring traffic [JR86]. For a single connection of a pair of stations the traffic between them can be viewed as several trains. A preset value is used to distinguish the start of a train. Sometimes, it is also viewed as an ON/OFF model but has a more practical application.

Another measurement-based modeling, self-similar traffic models have been proposed to model Ethernet traffic [LTWW94]. From the measurement it is apparent that the traffic on different time scales exhibits burstiness while conventional models look smooth on a large time scale. Moreover, the same phenomenon has been observed in an analysis of VBR video traces.

Increasing evidence shows that non-real-time traffic is bursty. To identify the degree of burstiness is an important issue for traffic modeling. The mathematical presentation of burstiness is very complicated. The shape of the marginal distribution and the autocorrelation function of interarrival time processes are the two major effects of burstiness [Sai94, EW94, Onv95]. However, the definition of burstiness is not unique. The ratio of peak-to-average traffic rate is adopted by ITU-U as its definition. The coefficient of variance is a more precise measure. In addition, the peakedness measure and the index-of-dispersion measure are two further definitions. Based on the recently discovered self-similar nature of Ethernet traffic the fractal correlation dimension has been proposed.

2.2 Which Traffic Model is More Appropriate?

A number of studies on virtual circuit switch networks, token ring networks, Ethernet, ISDN, and CCSN/SS7 indicate that the conventional Poisson based traffic models are not suitable for modeling real traffic. Twelve of these measurement studies are summarized below:

- Marshall and Morgan [MM85] analyzed a week's worth of data traffic on a DATAKIT Virtual Circuit Switch network at AT&T Bell laboratories. They calculated the interarrival and service time distribution by conventional first two-moment approximations. However, they also argued that the assumption will fail if the arrival process is not a renewal process.

- Gusella [Gus90] analyzed the behavior of the Ethernet local-area network in an environment of diskless workstations running the UNIX operating system. He concluded that a system built around diskless workstations is highly interactive, and short-term network utilization is more likely to affect users' behavior than long-term average utilization. This means that the traffic on Ethernet generated by diskless workstation is highly bursty.

- Fowler and Leland [FL91] analyzed congestion management through the measurement of Ethernet at the Bellcore Morristown Research and Engineering Center [LW91]. They found that the enormous time range over which data traffic is bursty means that it does not have a simple recurring pattern over a period of days, weeks, or months. Furthermore, the comparison of the peak to mean ratio and the index of dispersion of Poisson and observed data are very different.

- Meier-Hellstern *et al* [MHWYH91] monitored eight ISDN data users with Digital Subscriber Line (DSL) and analyzed the traffic data. The researchers argued that the interarrival histogram can be modeled as three different states with different distributions. The first two interarrival states exhibit the gamma distribution and the third state can be modeled with Pareto distribution.

- Heyman *et al* [HTL92] analyzed a 30-minute sequence of real-video teleconference data. In their study the number of cells per frame is a gamma (or negative binomial) distribution, rather than normal distribution. In addition, neither an autoregressive model of order 2 nor a two-state Markov chain model is sufficient because they do not model correctly the occurrence of frames with a large number of cells. The authors suggested a multi-state Markov chain model is sufficiently accurate for use in traffic study.

- Paxson [Pax94] analyzed 2.5 million TCP connections and found that log-extreme or log-normal distributions are better to model the observed data with individual protocols. However, although these models come close to describing the distribution, they are not statistically exact. Therefore, the models fits subsamples rather than ruling out simple hourly, daily, or weekly patterns in the parameters.

- More than 150 million SS7 messages have been collected by Duffy *et al* on SS7 signaling links [DMRW94]. The distributions of call holding times exhibit a heavy-tailed nature. Such distributions can be lognormal, Weibull, and Pareto. Therefore, SS7 traffic at the message level exhibits

LRD and therefore call holding times may be best described using heavy-tailed distributions with possibly infinite variance/mean.

- Leland *et al* [LTWW94] used four data sets collected during the period from 1989 to 1992. The shortest collecting period was more than 20 hours, while the longest one was 47.91 hours. In their paper the authors explored the self-similar phenomenon of the Ethernet traffic. In addition, self-similar traffic models are developed as parsimonious models to capture the bursty feature of data networks.

- Garrett and Willinger [GW94] analyzed a 2-hour long empirical sample of VBR video. They found that tail behavior of the marginal bandwidth distribution can be accurately described by heavy-tailed distributions. The autocorrelation of VBR video sequence decays hyperbolically, i.e., LRD. Therefore, VBR video can be modeled using self-similar processes.

- Paxson and Floyd [PF95] collected and analyzed a larger number of traffic traces from wide-area networks. They found that user-initiated TCP session arrivals are well-model-led by the Poisson process with fixed hourly rates, but the other connection arrivals deviate considerably from Poisson. Thus, they suggest that the Poisson-based modeling of wide-area traffic for all but user session arrivals should be abandoned.

- Beran *et al* [BSTW95] analyzed 20 large sets of actual VBR video data, generated by a variety of different codecs and presented through a wide range of different scenes. They concluded that LRD is an inherent feature of VBR video traffic; that is, VBR video traffic is independent of scene, such as video phone, video conference, motion picture video, and codec.

- Crovella and Bestavros [CB96] collected the network traffic from the World Wide Web in which the traffic arises as the result of file transfers in a closed system. Once the data was tested, it was found that the traffic was self-similar with a Hurst value of 0.82.

From the history of measuring different kinds of traffic, there is strong evidence to indicate that the conventional Poisson-based processes are not adequate to model the traffic on any network. Even in traditional telephone call holding times it has been recognized that the exponential approximation seriously underestimates the actual numbers of very long calls.

Leland *et al* in their pioneering and subsequent papers [LTWW94, WTSW97] illustrate a pictorial diagram to exhibit the self-similar phenomenon, which is cited as Fig. 2.2. The diagram is divided into five rows, each with a different time scale and three columns. The right hand column illustrates actual Ethernet traffic, and the middle and left hand ones are synthetic traces generated with traditional and self-similar traffic models, respectively.

Most of the traffic models described in Sec. 2.1 are related to the family of Poisson or Markov processes which are known as short-range dependent, except for the packet-train and self-similar traffic models. Of the two exceptions the packet-train traffic model is a concept rather than an analytical model. Thus, self-similar traffic models are believed to be the one which can most accurately capture the nature of traffic on networks and also provide a rigorous mathematical derivation.

2.3 Do Self-Similar Traffic Models Meet the Criteria of Traffic Models?

Some important criteria for selecting a traffic model are given in [CPL+94]. These criteria are accuracy, generality, stability, analytical tractability, parameterization, and number of parameters. All of these are based on the work of conventional queueing theory. Self-similar traffic models will be examined and the applicability of the criteria stated.

2.3.1 Accuracy/Close to Reality/Physical Meaning

A traffic model will work as a tool for analysis or simulation of ATM networks when its statistical characteristics are consistent with real traffic. However, this argument is only an ideal concept. There are still several characteristics that have not yet been well defined, including burstiness and QoS, in ATM networks. Self-similar traffic models are extracted from the phenomenon of measurement of real traffic. The measurement shows that traffic exhibits burstiness no matter what time scale is used to collect the data. In conventional traffic models the traffic will become smooth when a large time scale is taken to measure the traffic as in the middle column in Fig. 2.2.

To expand the scope of the criterion previously referred to a 'accuracy' to encompass 'closeness to reality' and 'physical meaning' allows a more significant representation of traffic models to be made. Self-similar traffic models, by derivation, are closed to reality and carry more physical meaning, and therefore an potentially more accurate description of real traffic.

2.3.2 Generality

The majority of conventional queueing models are designed for specific sources. Although a number of traffic models can be extended to fit the requirements of traffic sources, this can cause complicated and intractable problems. Self-similar traffic models are able to cover a large family of traffic streams so that their various characteristics can be preserved. Many traffic measurements exhibit the self-similar phenomenon, thus, self-similar traffic models are more general than other traffic models.

2.3.3 Usability for Simulation/Statistical Stability

To facilitate simulation study, traffic models are required to have statistical stability. Statistical stability is tested through a period of time, which is proportioned not only to the highest level of resolution in time but also to the number of different states of the model. If the highest resolution and the number of states are directly proportional to time, they can be reasonably expected to be statistical stability. Otherwise, model parameters still could be produced if accompanying by measures of statistical certainty. The self-similar stochastic process does not have a linear proportional relationship, but it follow a power law. In many measurements, traffic exhibits heavy-tailed distributions, such as Pareto and Weilbull distributions, rather than Poisson distribution. This feature coincides with the self-similar phenomenon.

2.3.4 Analytical Tractability

An appropriate traffic model should keep the usage of computational resources to a reasonable level. Fast-growing networks or the superposition of sources always requires numerical computation and causes the increases in resource consumption. Due to large volume and complexity with actual traffic, a simple traffic model is hard to obtain. Therefore, approximate methods need to be used to help solve the problems. Approximate methods should retain the consumption of the computational resources to an acceptable level. However, their results need to satisfy the requirement of accuracy. The commonly-used approximate approaches include iteration methods, aggregation methods, and matrix analytical methods.

In the majority of cases, the preservation of analytical tractability of resultant queueing or performance problems dominates the development of traffic models. Therefore, in the statistical sense, few of the resultant models have been tested against real traffic to demonstrate how well they work. Self-similar traffic models do not have the attractive properties which the conventional queueing models have. Until now, little research has covered the analysis of queueing behavior with self-similar traffic models. Moreover, the suitability of evaluating the models with this criterion is still unknown, and may need further investigation, which is beyond the scope of this thesis.

2.3.5 Method of Model Parameterization

All traffic models must be parameterizable to describe their statistical characteristics. Even though the parameters of the models can be directly measured, it is unlikely to be obtained without any bias. To reduce the degree of biased estimation, several equations can be employed, which include the mean, variance, autocovariance, and probability distribution of the models. Autocorrelation is added by some researchers as an extra statistical measure to provide sufficient parameterization information. Others think that it is a key parameter that influence network performance. The Hurst parameter derived from the autocorrelation of self-similar time series is provided as an index to help describe the degree of correlation in self-similar traffic models. A number of Hurst estimators are described in Sec.3.5, although some of them are approximate methods, they still provide ways to obtain the Hurst value of the fractal samples efficiently.

2.3.6 Number of Parameters

The number of the parameters used is directly related to the complexity of the model. More parameters increase the dimension of the problem considerably when the size of the networks or the number of cell streams increase. Limiting the number of parameters is a critical requirement in traffic modeling.

Normally, when the size of the networks or the number of sources increase, the number of parameters of the family of the Poisson or Markov processes will increase significantly. Various iteration algorithms have been proposed to overcome the parameter problem, but the computational time is increased dramatically. For self-similar traffic models, only one more parameter, the Hurst parameter, is added. Therefore, the self-similar traffic model is a parsimonious model.

2.4 Summary

CBR services have been studied since the renowned pioneering telephone engineer A. K. Erlang first investigated telephone traffic. There is no argument about the suitability of the Poisson process in the oldest telecommunications territory. However, technological advances have brought more data traffic into telecommunication networks. ATM is used to integrate various services into telecommunications. Understanding the incoming traffic is fundamental to designing, operating, and maintaining ATM networks. Therefore, traffic models are a fundamental basis for the discrete-event simulations and the analytical modeling of communications systems.

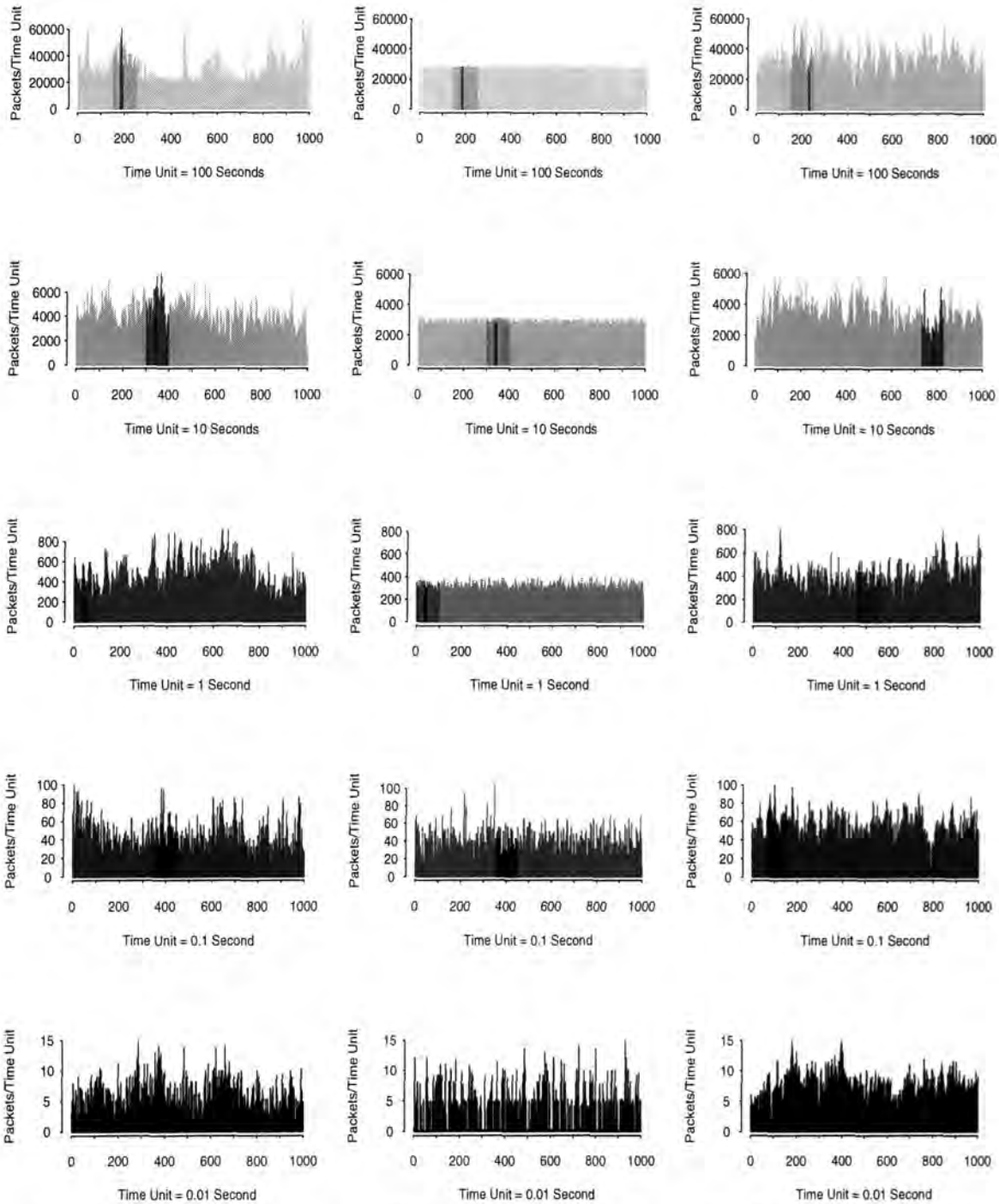
A great number of traffic models have been proposed to attempt to capture their underlying mechanisms. Different types of VBR service have different approaches which can be used to express their characteristics. Traffic burstiness is the most important issue as it destroys the assumption of time independence. Data applications normally use simple or compound Poisson processes to represent traffic behavior. For VBR voice sources IPP, IBP, and MMPP are commonly used to imitate the behavior of VBR voice sources. At present, the majority of research concentrates on compressed video traffic. A great number of traffic models, including the IPP, MMPP, ARMA, have been proposed to model the VBR video traffic.

There is increasing evidence that VBR data is bursty in different applications. Although many researchers accept the use of the Poisson-based processes (or the more generalized Markovian processes) to model the data applications, it is very difficult to capture the characteristic of the traffic in ATM networks. Many sophisticated models, including MMPP, SMP, PH renewal process, and MAP, have been proposed to deal with the characteristics. However, almost all of them are too complicated to handle when the number of sources increases.

The packet train and self-similar models are the two which extract the characteristics of networks from the observations of Ethernet and token ring network. However, the packet train model is a concept, in which the critical value to indicate the start and end of trains is not well-defined, rather than a rigorous mathematic model. Thus, self-similar traffic models are more appropriate not only because the models can match many traffic measurements but also because they have more rigorous mathematical foundations.

Self-similar traffic models have been chosen for further study because they can fit the findings of a number of traffic measurements well. Furthermore, self-similar traffic models have been tested through six commonly-used criteria. Their failure is in one of these, analytical tractability, might harm the users' perception when considering the theoretical derivation. However, to date no evidence has been drawn that such a failure definitely spoils their fit in practical cases. It is noted that this criterion is based on the development of Markov-based processes, so it might not be an important assessment for self-similar models. Therefore, the above evaluation supports the view that self-similar traffic models are more appropriate than the others.

In the next chapter, the mathematical foundation of self-similar traffic models and the models for self-similar phenomena are outlined. The tools for assessing the fractal samples are also introduced.



(a) Actual Ethernet Traffic (b) Traditional Traffic Models (c) Self-Similar Traffic Models

Figure 2.2: Pictorial proof of Self-Similar traffic models - actual Ethernet traffic (left column), synthetic trace generated from an appropriately chosen traditional traffic model (middle column), and synthetic trace generated from an appropriately chosen self-similar traffic model with a single parameter (right column).

Chapter 3

Self-Similar Stochastic processes

Although traffic models based on Poisson processes are believed to work well in representing traffic behavior in telephone systems, they fail to handle the burstiness of multimedia applications. Several models have been introduced to help resolve this problem. In the last chapter, self-similar traffic models have been highlighted as being more appropriate. It is not only for the reason that they fit the actual measurements well, but also that they satisfy most of the criteria for traffic modeling.

This chapter explains the mathematical foundation on which self-similar traffic models are based. It justifies the models as an accurate representation of the self-similar phenomena for the further investigation which follows. The chapter is divided into five main parts. Sec. 3.1 introduces the fundamental issues about fractals and self-similarity. In Sec. 3.2, the derivation of the Hurst parameter and its relationship with random fractals are discussed. In Sec. 3.3, self-similar stochastic processes and their properties are explained. In Sec. 3.4, three models for self-similar phenomena are presented and evaluated. Finally, the discussion is focused on three chosen estimators and the introduction of a visual assessment tool.

3.1 Fractals and Self-Similarity

Before introducing the properties of self-similarity, fractals should be discussed first. Fractal dimension provides a method to measure the size of a fractal or how similar two different fractals are. Once fractal dimension has been discussed, the concept of self-similarity is introduced.

3.1.1 Fractals

The inventor of the term fractal attempted to provide a definition. Benoit B. Mandelbrot offered the following tentative definition of a fractal in 1983 [Man83]:

> A fractal is by definition a set for which the Hausdorff-Besicovitch dimension strictly exceeds the topological dimension.

This definition requires a definition of the terms *set, Hausdorff-Besicovitch dimension*, and *topological dimension*. The latter is always an integer. However, the definition is too strict to fit many newly-discovered fractal structures, especially the chaotic phenomena of dynamic systems. A looser definition

is required, therefore Mandelbrot has produced another tentative definition, proposing the following [Fed88]:

> A fractal is a shape made of parts similar to the whole in some way.

Although Mandelbrot has tried to give a neater and more complete characterization of fractals, a precise definition is still lacking. The second definition contains the fundamental feature that has been noted in experiments, a fractal looks the same whatever the scale.

One of the most famous applications of fractals is in solving the problem of how long the coastline of Britain is. A British map using different rules of measurement produce different lengths for the British coastline. Another well-known example is described in Richarson's article [Ric61]. The lengths of the common frontiers between Spain and Portugal, or Belgium and Netherlands, as reported in their neighbor's encyclopedias, differ by 20%. Richardson had studied empirically the variation of the approximate length, $L(\varepsilon)$, where ε is the yardstick length. In his study the nature of turbulence involves a self-similar cascade. There are two constants, called F and D, such that one needs roughly $F\varepsilon^{-D}$ intervals of length ε, leading up to the length

$$L(\varepsilon) \sim F\varepsilon^{1-D}. \tag{3.1}$$

The value of the exponent D seems to depend on the coastline and the yardstick length and to be independent of the method chosen to estimate the length of the coastline. The discovery gives the idea of scaling although later research has shown that the relation is not so simple.

3.1.2 Self-Similarity

As the latter definition of fractals mention, a fractal is a shape which looks similar to the whole in some way. There should be a way to measure the degree of complexity by evaluating how fast length, surface, or volume increases if it is measured with respect to smaller and smaller scales. Before going further, the term 'self-similar' should be defined first [PJS92]:

> A structure is said to be (strictly) self-similar if it can be broken down into arbitrarily small pieces, each of which is a small replica of the entire structure.

This definition is only a pictorial idea. In order to measure the degree of self-similarity, Mandelbrot defined the self-similar dimension [Man83]:

Definition 1 Given a self-similar structure, there is a relation between the reduction factor s and the number of pieces a into which the structure can be divided; and that is

$$a = 1/s^{D_s} \tag{3.2}$$

or equivalently

$$D_s = \log a/\log(1/s). \tag{3.3}$$

D is called the *self-similarity dimension.* □

However, not all self-similar structures are fractals, such as a line segment, a square, or a cube. Each one can be broken into small copies which are obtained by similarity transformations.

The definition provides a concept rather than a measure. Several measuring methods have been proposed to bring the concept to reality. The compass dimension is the simplest one. The relation between the power law of length measurement using different compass settings and the self-similarity dimension of a fractal curve is very simple. The compass setting method uses different yardsticks to compass a measured polygon. First, simply choose appropriate units of length measurements such that the factor d in power law become unity

$$u = \frac{1}{s^d}, \tag{3.4}$$

where u is the length with respect to compass setting s. Taking logarithms, we obtain

$$\log u = d \cdot \log \frac{1}{s}. \tag{3.5}$$

On the other hand, taking the logarithm on the power law $a = 1/D^s$, where a denotes the number of pieces in a replacement step of the self-similar fractal with scaling factor s, is

$$\log a = D_s \cdot \log \frac{1}{s}, \tag{3.6}$$

The connection between length u and number of pieces a is that at scaling factor $s = 1$ we measure a length $u = 1$. Therefore, when measuring at some other scale s, where the whole object is composed of a small copies each of size s, then a total length of a times s is measured as

$$u = a \cdot s. \tag{3.7}$$

After deriving the relationship of self-similar dimension D_s and the exponent factor d can be expressed as

$$D_s = 1 + d. \tag{3.8}$$

3.2 Fractal Records in Time

Many natural observations are recorded in time series. For example, the records of temperature vary erratically on both the short- and the long-term time scales. In addition, the discharge of rivers, rainfall, and thickness of tree rings can be analyzed by an exponent H – the Hurst parameter [Hur51, Fed88].

Hurst was a civil engineer who spent his lifetime studying the problem of water reservoirs of the Nile. He invented a new statistical method, the rescaled range analysis (R/S analysis), as a result of his recording of the discharges of Lake Albert over 60 years to determine the design of an ideal reservoir. An ideal reservoir never overflows or empties. In any given year t, the reservoir will accept the influx $X(t)$ from the lake, and a regulated volume per year, $E[X(\tau)]$, will be released from the reservoir. The average influx over the period of τ years is

$$E[X_\tau] = \frac{1}{\tau} \sum_{t=1}^{\tau} X(t). \tag{3.9}$$

The average should equal the volume released per year from the reservoir. Let $Y(t)$ be the accumulated departure of the influx $X(t)$ from the mean $E[X_\tau]$

$$Y(t,\tau) = \sum_{u=1}^{t} \{X(u) - E[X_\tau]\}. \tag{3.10}$$

The difference between the maximum and the minimum accumulated influx X is the *range* R. The range is the storage capacity required to maintain the mean discharge throughout the period. The explicit expression for R is

$$R(\tau) = \max_{1 \le t \le \tau} Y(t,\tau) - \min_{1 \le t \le \tau} Y(t.\tau), \tag{3.11}$$

where t is a discrete integer-valued time and τ is the time-span considered. Clearly, the range depends on the time period τ considered and as expected the range R increases with increasing τ.

Hurst investigated many natural phenomena, such as river discharges, mud sediments, and tree rings. He used the dimensionless ratio R/S, where S is the standard deviation. The use of this dimensionless ratio permits the comparison of observed ranges of various phenomena. The standard deviation is estimated from the observations by

$$S = (\frac{1}{\tau} \sum_{t=1}^{\tau} \{X(t) - E[X_\tau]\}^2)^{1/2}. \tag{3.12}$$

Hurst found that observed rescaled range, R/S, for many records in time is very well described by the following empirical relation:

$$R/S = (\tau/2)^H. \tag{3.13}$$

The *Hurst exponent* H is more or less symmetrically distributed about a mean of 0.73 with a standard deviation of about 0.09. Hurst's observation is remarkable considering the fact that in the absence of long-run statistical dependence, R/S should become asymptotically proportional to $\tau^{1/2}$ for records generated by statistically independent processes with finite variances. The formula is given by

$$R/S = (\pi\tau/2)^{1/2}. \tag{3.14}$$

3.3 Self-Similar Stochastic Processes

From the pictorial illustration of measurement of Ethernet traffic in Fig. 2.2, the scale invariant or self-similar feature is drastically different from conventional telephone traffic or the currently considered stochastic models for packet traffic. The use of self-similar stochastic processes is one method to model self-similar features. The following presentation is based almost entirely on the work of Cox and Leland *et al* [Cox84, LTWW93, LTWW94].

Self-similar stochastic processes are invariant in distribution under judicious scaling of time and space. They are important in probability because of their connection to limit theorems and are of considerable interest in modeling. Self-similar phenomena appear in geophysics, hydrology, physics, chemistry, turbulence, economics, and communications. It is related to an α-stable distribution with $0 < \alpha < 2$,

where α is the index of stability of a distribution. An α-stable stochastic process is a random element whose finite-dimensional distributions are α-stable, which are preserved under convolution and approximate the distribution of normalized sums of i.i.d random variables making them useful in modeling the contribution of many small random effects [ST94, WTE96].

3.3.1 Definition

A stochastic process is an uncountable infinity of random variables, one for each sampling time t [BJ76, BD91]. For a specific t, $X(t)$ is a random variable with distribution

$$F(x,t) = P\{X(t) \leq x\}. \tag{3.15}$$

The function $F(x,t)$ is called the first-order distribution of the process $X(t)$. Its derivative with respect to X

$$f(x,t) = \frac{\partial F(x,t)}{\partial x} \tag{3.16}$$

is the first order density of $X(t)$. The second-order distribution of the process $X(t)$ is the joint distribution

$$F(x_1,x_2;t_1,t_2) = P\{X(t_1) \leq x_1, X(t_2) \leq x_2\} \tag{3.17}$$

of the random variables $X(t_1)$ and $X(t_2)$. The corresponding density equals

$$f(x_1,x_2;t_1,t_2) = \frac{\partial^2 F(x_1,x_2;t_1,t_2)}{\partial x_1 \partial x_2}. \tag{3.18}$$

For the determination of the statistical properties of a stochastic process, knowledge of the function $F(x_1,\cdots,x_n;t_1,\cdots,t_n)$ is required for every x_i,t_i and n. However, for many applications, only certain averages are used.

Let $X = \{X_t : t = 0,1,2,\cdots\}$ be a discrete-time wide-sense stationary stochastic process. Its autocovariance at lag k is given by $\gamma_k = Cov[t+k,t]$, and variance $\sigma^2 = \gamma_0$, so autocorrelation at lag k is $\rho_k = \gamma_k/\gamma_0$. The spectral density (power spectrum) function of this process is

$$S(\omega) = \frac{1}{2\pi} \sum_{k=-\infty}^{\infty} \gamma_k e^{-jk\omega}, \tag{3.19}$$

$$\rho_k = \int_{-\pi}^{\pi} e^{jk\omega} S(\omega) d\omega. \tag{3.20}$$

Therefore, the autocovariance and the spectral density function are equivalent specifications of second-order properties.

For each $m = 1,2,\cdots$, let $X^{(m)}$ denote a new time series obtained by averaging the original series X over non-overlapping blocks of size m, replacing each block by its mean. This is, for each $m = 1,2,\cdots$, $X^{(m)}$ is given by

$$X_k^{(m)} = (X_{km-m+1} + \cdots + X_{km})/m, \quad k = 1,2,3,\cdots \tag{3.21}$$

Note that the new time series defines a second-order wide-sense stationary process with autocovariance function $\gamma_{(k)}^m$, variance $\sigma_m^2 = \gamma_0^{(m)}$, and associated variance sequence $[\sigma_k^2]^{(m)} = Var_k^{(m)}$.

A well known internal correlation has been proved by Yule [Yul45] that with $\sigma_1^2 = \sigma^2$

$$\sigma_m^2 = \frac{\sigma^2}{m} + \frac{2}{m^2}\sum_{k=1}^{m}(m-k)\gamma_k = \frac{\sigma^2}{m} + \frac{2}{m^2}\sum_{s=1}^{m-1}\sum_{k=1}^{s}\gamma_k. \tag{3.22}$$

Directly we can obtain

$$\gamma_k = \frac{1}{2}\cdot\nabla^2(k^2\sigma_k^2), \tag{3.23}$$

where ∇^2 denotes the central second difference operator applied to a function $f(k)$, i.e., $\nabla^2(f(k)) = f(k+1) - 2f(k) + f(k-1)$. Thus a specification of the sequence X is equivalent to one of $\{\gamma_k\}$ or of $S(\omega)$. The autocovariance can be calculated as

$$\gamma_k^{(m)} = 1/2\nabla^2(k^2 Var_k^{(m)}) \tag{3.24}$$

which can be re-expressed in terms of the $\{\gamma_k\}$ by Eq. (3.22). In particular, the autocorrelation $\rho_k^{(m)}$ of the average process can be deduced from Eq. (3.24). The most important special case is the lag one autocorrelation

$$\rho_1^{(m)} = 2Var_2^{(m)}/Var^{(m)} - 1. \tag{3.25}$$

Processes, called stochastic processes with *short-range dependence*, satisfy the following essentially equivalent properties:

S-1 $\sum_{k=0}^{\infty}\gamma_k$ is convergent;

S-2 $S(0) < \infty$;

S-3 σ_m^2 is for large m asymptotically of the form σ^2/m;

S-4 the average process $\{X_t^{(m)}\}$ tends to second order pure noise as $m \to \infty$.

These properties are forms of second-order strong mixing, and moving average processes, which means these kind of processes have rational power spectral densities.

Conversely, processes with LRD are expressed by the following properties:

L-1 $\sum\gamma_k$ is diverge;

L-2 $S(\omega)$ is singular near $\omega = 0$;

L-3 $m\sigma_m^2 \to \infty$ as $m \to \infty$.

More precisely, these properties can be represented as follows, with $0 < \beta < 1$:

L'-1 as $k \to \infty$, $\gamma_k \sim \gamma' k^{-\beta}$;

L'-2 as $\omega \to 0$, $S(\omega) \sim S'\omega^{-(1-\beta)}$;

L'-3 as $m \to \infty$, $\sigma_m^2 \sim V' m^{-\beta}$.

γ', S', and V' are connected by simple relations.

In addition, at least in theory there are other possibilities to study processes which contrast with those with short-range dependence. The insertion of slowly varying functions into the asymptotic forms would have a relatively minor effect. Another possibility arises from a local behavior of $S(\omega)$ near $\omega = 0$ of the form

$$S_2'\omega^{-\beta_2} + 1/2[1 + \cos(k/\omega)]S_1'\omega^{-\beta} \tag{3.26}$$

with $0 < \beta_1, \beta_2 < 1, \beta_1 \neq \beta_2, S_1', S_2', k > 0$. It follows from Eq. (3.24) and (3.25).

L'-4

$$\rho_k^{(m)} \rightarrow 1/2\nabla^2(k^{2-\beta}) \simeq 1/2(2-\beta)(1-\beta)k^{-\beta}, \tag{3.27}$$

$$\rho^{(m)} \rightarrow 2^{1-\beta} - 1 \neq 0, \text{ as } m \rightarrow \infty. \tag{3.28}$$

For a large k, Eq. (3.27) derives from the asymptotic equivalence of differencing to differentiation.

All processes satisfying properties L'-1 - L'-4 are termed *stochastic processes with LRD of index* β. From a direct statistical point of view, their salient feature is perhaps that the variance of a mean decreases more slowly than the reciprocal of sample size. From an heuristic point, the averaged process $\{X_t^{(m)}\}$ takes a non-degenerated correlated structure as $m \rightarrow \infty$.

The properties of L'-4 can be called *asymptotic second-order self-similarity.*

Definition 2 The process $\{X_t\}$ is called *exactly second order self-similar* if, for $k = 1, 2, \cdots$,

$$\rho_k = 1/2\nabla^2(k^{2-\beta}) \tag{3.29}$$

where $\rho_k^{(m)} = \rho_k$ and $\sigma_m^2 = \sigma^2 m^{-\beta}$, i.e., the process and the average process $\{X_t^{(m)}\}$ have an identical correlated structure. □

However, the definitions of self-similarity do not apply to continuous time because the relation $\sigma_k^2 = \sigma^2 k^{-\beta}$ cannot hold for arbitrarily small k if the process itself is to have finite variance. If a process merely concerns a large value of k, i.e., the asymptotic version of self-similarity, failures in the self-similarity property as $k \rightarrow 0$ are of no concern. If the exact self-similarity is concerned, it is in effect necessary to regard $\{X(t)\}$ as the (improper) derivative of a process $\{Y(t)\}$, i.e., to work with

$$Y(t) = \int_{t_0}^{t_0+t} X(u)du. \tag{3.30}$$

For example, if $\{X(t)\}$ is the formal derivative of Brownian motion, so that $\{Y(t)\}$ itself is Brownian motion, the self-similarity of $\{Y(t)\}$ is expressed through the property that $\{m^{-1/2}Y(mt)\}$ and $\{Y(t)\}$ have the same variance. For the more general definition $\{m^{\frac{1}{2}\beta-1}Y(mt)\}$ and $\{Y(t)\}$ should have the same variances. Here $Y(t)/t$ is one value, starting at t_0, of the process $\{X(t)\}$ averaged over the period t. The parameter $1 - \frac{1}{2}\beta$ is termed the *Hurst coefficient* and denoted by H where the parameter $1 + \frac{1}{2}\beta$ is called the *Hausdorff-Besicovitch dimension* and denoted by D.

3.3.2 Properties of Self-Similar Processes

A stochastic process with LRD must satisfy L'-4, i.e., it can be characterized by an autocorrelation that decays hyperbolically as the lag increases. Thus, it implies $\sum_k \gamma_k = \infty$. This result implies the fact, while high-lag correlations are all individually small, their cumulative effect is of importance and gives rise to features which are drastically different from those of conventional stochastic processes.

In the frequency domain, LRD manifests itself in a spectral density that obeys a power-law behavior near the origin. From property L'-2, LRD implies that $\sum_k \gamma_k$ is divergent (property L'-1). This means its spectral density tends to $+\infty$ as the frequency ω approaches 0 ($1/f$-noise). Heuristically, LRD manifests itself in the presence of cycles of all frequencies and in the power law of magnitude, and displays features suggestive of non-stationarity. In addition, self-similar processes provide a theoretical explanation and interpretation of the empirical law, Hurst effect.

From a statistical point of view, the most salient feature of the self-similar process is that the variance of the arithmetic mean decreases more slowly than the reciprocal of the sample size, i.e., it behaves like $k^{-\beta}$ for some $\beta \in (0,1)$, instead of like k^{-1} for the processes whose aggregated series converge to second-order pure noise. From property L'-3, V' is slowly varying at infinity, that is, $\lim_{t\to\infty} V'(tx)/V'(x) = 1$ for all $x > 0$. The consequences of the slowly decaying variances for classical statistical tests and confidence or prediction intervals can be disastrous, since usual standard errors are wrong by a factor that tends to infinity as the sample size increases.

The majority of conventional stochastic models have the property that their aggregated processes $\{X^{(m)}\}$ tend to second-order pure noise (property S-1). The result is that the covariances will eventually decrease exponentially, the continuity of the spectral density function at the origin will eventually show up, the variances of the aggregated processes will eventually decrease as m^{-1}, and the rescaled adjusted range will eventually increase as $n^{0.5}$. For finite sample sizes, distinguishing between those which are asymptotic and those corresponding to self-similar process is problematic. For time series with hundreds of thousands of observations, parsimonious modeling becomes a necessity due to the large number of parameters needed when trying to fit a conventional process to a truly self-similar model. Although it is possible for traditional models to represent self-similar model, the number of parameters tends to infinity as the sample size increases. By using FGN and $FARIMA(p,d,q)$ processes, the number of parameters is at most four.

3.4 Models for Self-Similar Phenomena

To model the self-similar phenomena, there are two approaches, stochastic models which are based on self-similar processes and deterministic models which use nonlinear chaotic maps. A finite approximation of a continuous sum of the Gauss-Markov process is often suggested to manifest the presence of a multilevel hierarchy of the underlying mechanisms to account for the self-similarity. However, in practice it is very hard to represent the multilevel hierarchies of the underlying mechanisms as well as to show why its presence should contribute to self-similar behavior. Therefore formal mathematical models have been used to represent the self-similar phenomena. Two such models are discussed below: one is

the exactly self-similar FGN; the other is the asymptotically self-similar FARIMA process. In addition, capturing the deterministic self-similarity from fractals with the usage of chaotic maps to model traffic is introduced.

3.4.1 Fractional Brownian Motion

The exactly self-similar FGN or FBM is very important because it is the unique self-similar Gaussian process with the self-similar index H, also because the various α-stable self-similar processes with index H reduce to it when setting $\alpha = 2$, and because it has been widely applied, particularly in the context of LRD. FBM has two different representations: moving-average and harmonizable. These two representations are inherent in the properties of α-stable processes [ST94].

A FGN $X = \{X_k : k = 0, 1, 2, \cdots\}$ is a stationary Gaussian process with mean $\mu = E[X_k]$, variance $\sigma^2 = E[(X_k - \mu)^2]$, and autocorrelation

$$\rho_k = 1/2(|k+1|^{2H} - |k|^{2H} + |k-1|^{2H}), \quad k = 1, 2, 3, \cdots \tag{3.31}$$

From the form of Eq. (3.27) above formula is very easy to be expressed as

$$\rho_k \sim H(2H-1)|k|^{2H-2} \tag{3.32}$$

as $k \to \infty$ with $0 < H < 1$. The resulting aggregated processes $X^{(m)}(m = 1, 2, 3, \cdots)$ all have the same distribution as X, for all $0 < H < 1$. Thus, FGN is exactly second-order self-similar with Hurst parameter, H, as long as $1/2 < H < 1$.

FBM is one of the models to represent self-similar processes that have been proposed to model the real-world traffic. Its definition and properties are stated below.

Definition

When considering a one-dimensional Brownian motion in which a particle moves on a line, the moving step-length η of the particle is given by a Gaussian or normal probability distribution [MN68, Man83, Fed88, ST94]. In practice it is impossible to observe a Brownian motion with infinite resolution. Rather than observing the particle in continuous time, the particle position can be observed only at intervals $b\tau$, where b is an arbitrary number. The increments of the particle position constitute an independent Gaussian process with $E[\eta] = 0$ and a variance of $E[\eta^2] = 2Dt$ with $t = b\tau$, where D is the *diffusion coefficient*. This results in the Brownian record looking 'the same' under a change of resolution. Such a result is termed a *scale invariance* or *symmetry* of the Brownian record.

The scaling property of Brownian motion changes the probability density as follows:

$$p(\hat{\eta} = b^{1/2}\eta, \hat{\tau} = b\tau) = b^{-1/2}p(\eta, \tau). \tag{3.33}$$

This equation shows that the Brownian random process is invariant in distribution under a transformation that changes the time scale by b and the length scale by $b^{1/2}$. The probability distribution for the particle position $B(t)$ is also found by the methods used above and can be written as

$$P\{b^{1/2}[B(bt) - B(bt_0)]\} = b^{-1/2}P\{B(t) - B(t_0)\}. \tag{3.34}$$

The position of a Brownian particle is a random function of time t. Consider a normalized independent Gaussian random process $\{\eta\}$. Let the increment in the position of the Brownian particle be given by

$$B(t) - B(t_0) \sim \eta |t - t_0|^H, \quad t \geq t_0, \tag{3.35}$$

for any two time epochs, t and t_0. Here $H = 1/2$ for ordinary Brownian motion.

The generalization of the random function $B(t)$ can be easily derived by changing the parameter $H = 1/2$ to any real number in the range $(0, 1)$. In [MN68], Mandelbrot and Ness proposed to designate FBM, $B_H(t)$, a family of Gaussian random functions, in which $B(t)$ is ordinary Brownian motion and H is a parameter satisfying $0 < H < 1$. The exponent H is a moving average of the past increments of $B(t)$ which is weighted by the kernel function $(t-s)^{H-1/2}$. The fundamental concept of FBM is that the span of interdependence between their increments can be said to be infinite.

Let ω be the set of all values of a random function and t represent time with $0 < t < \infty$. The ordinary Brownian motion, $B(t, \omega)$, is a real random function with independent Gaussian increments such that $B(t_2, \omega) - B(t_1, \omega)$ has zero mean and variance $|t_2 - t_1|$. The stochastic process FBM, $B_H(t, \omega)$, is defined as follows:

Definition 3 Let H be such that $0 < H < 1$, and let b_0 be an arbitrary random number. The following random function $B_H(t, \omega)$ is called a *reduced fractional Brownian motion* with parameter H and starting value b_0 at time 0. For $t < 0, B(t, \omega)$ is defined by

$$B_H(0, \omega) = b_0, \tag{3.36}$$

$$B_H(t, \omega) - B_H(0, \omega) = \frac{1}{\Gamma(H + 1/2)} \int_{-\infty}^{t} K(t - t') dB(t', \omega), \tag{3.37}$$

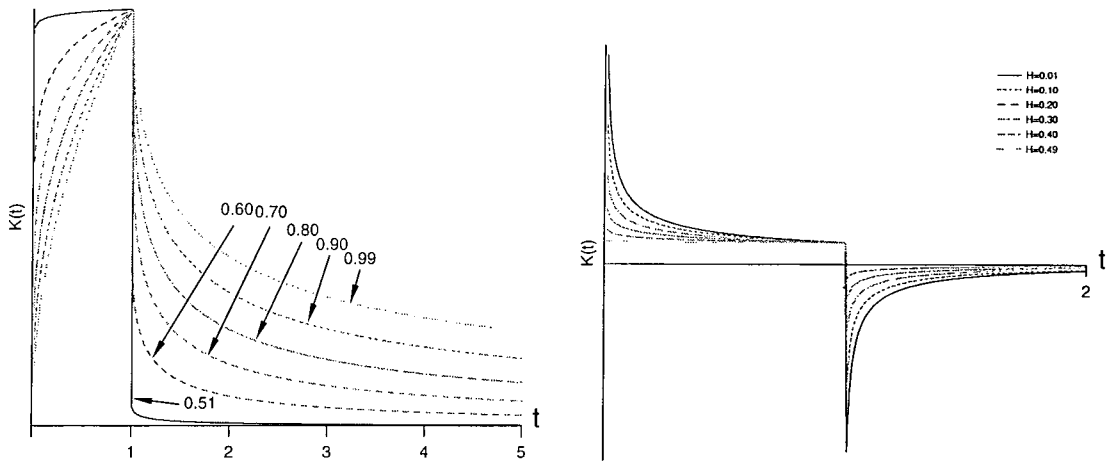
(and similarly for $t < 0$), where the function $K(u)$ is defined as follows:

$$K(t - t') = \begin{cases} (t - t')^{H-1/2} - (-t')^{H-1/2} & \text{if } t' < 0, \\ (t - t')^{H-1/2} & \text{if } 0 \leq t' \leq t. \end{cases} \tag{3.38}$$

□

The kernel vanishes quickly as $t' \to -\infty$ to allow the expression to properly define the random function $B_H(t, \omega)$. It is more convenient to use $B_H(t)$ to represent $B_H(t, \omega)$. The kernel function $K(t)$ is depicted in Fig. 3.1. In this diagram Fig. 3.1 (a) depictes the kernel function with $\frac{1}{2} < H < 1$ and Fig. 3.1 (b) illustrates a number of the chosen Hurst values between 0 and $\frac{1}{2}$.

In Fig. 3.2, the FBM $B_H(t)$ and its increments, FGN, $G_H(t)$, are shown as a function of time, using $B_H(0) = 0$. Different Hurst values are illustrated in the diagrams. In the first row the trace is generated with $H = 0.5$, while the second and third rows are $H = 0.7$ and $H = 0.9$, respectively. The left column shows sample traces of FBM, the right column shows the corresponding sample traces of FGN processes The data shown in this figure is generated by the discrete fractional Gaussian noise (dFGN) algorithm which is introduced in Chapter 4.



(a) $\frac{1}{2} < H < 1$ (b) $0 < H < \frac{1}{2}$

Figure 3.1: The kernel function of FBM with different Hurst values.

Properties

Because it is a model that represents self-similar stochastic processes, FBM and its increments inherit the properties of self-similar stochastic processes, i.e., the process has (i) its autocorrelation function decaying hyperbolically as the lag increases, (ii) its spectral density tending to $+\infty$ as the frequency approaches 0 ($1/f$-noise), and (iii) its variance of the arithmetic mean decreasing more slowly than the reciprocal of the sample size. As a result of these properties Mandelbrot and Wallis refer to this phenomenon as the *Joseph effect*, an illusion to the biblical figure Joseph who faced long periods of plenty followed by long periods of famine [Man69].

In addition, FBM is derived from ordinary Gaussian noise so its FGN inherits the scaling property. When the time scale changes by a factor b, the kernel function is shown as $K(bt - b\hat{t}) = b^{H-1/2}K(t - \hat{t})$, where $b\hat{t} = t'$. Thus, the following can be obtained

$$G_H(t) = B_H(bt) - B_H(0) = b^H[B_H(t) - B_H(0)]. \tag{3.39}$$

Besides the preceding properties, there are several other remarkable and important properties [MN68]. First, as FGN is a stationary stochastic process the standard deviation of $G_H(t)$ can be stated as follows:

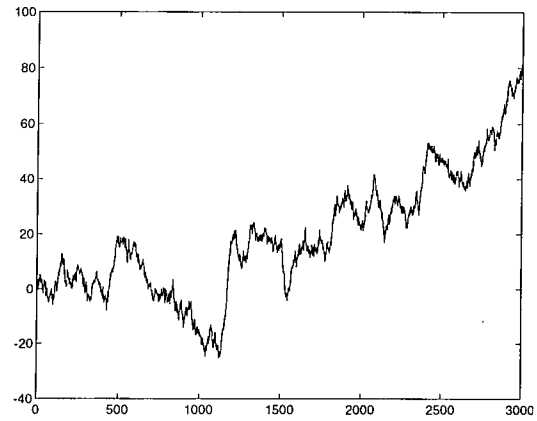
$$E[G_H(t+T)] = E[B_H(t+T) - B_H(t)]^2 = T^{2H}V_H, \tag{3.40}$$

where

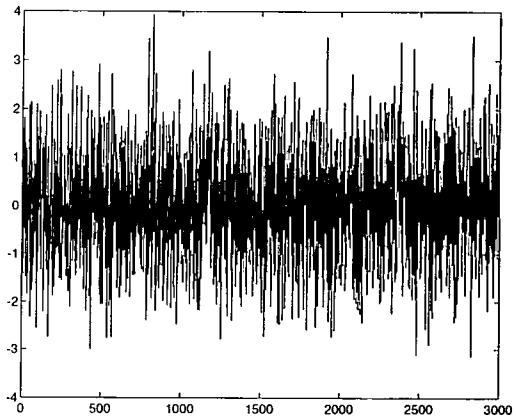
$$V_H = [\Gamma(H+1/2)]^{-2}\{\int_{-\infty}^{0}[(1-s)^{H-1/2} - (-s)^{H-1/2}]^2 ds + \frac{1}{2H}\}. \tag{3.41}$$

Although Eq. (3.40) produces the result that $B_H(t)$ is mean square continuous, the characteristics of the sample paths are still unknown. Mandelbrot and Ness show that almost all sample paths of $B_H(t)$, $0 < H < 1$ are continuous and almost surely not differentiable. The ordinary Brownian motion also has the same difficulty of no derivative. Thus, FGN is evolved to give meaning to the concept of the derivative of FBM.

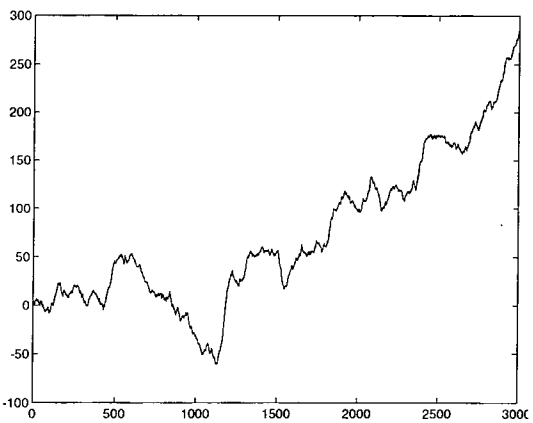
Furthermore, FGN is a stationary Gaussian sequence with mean zero and variance $E[G_H^2(t)] = E[B_H^2(1)] = \sigma^2$. Let $\gamma(k) = E[G_H(0)G_H(k)], k \in Z$, denote its autocovariance function and $\{S(\lambda), -\pi \leq$



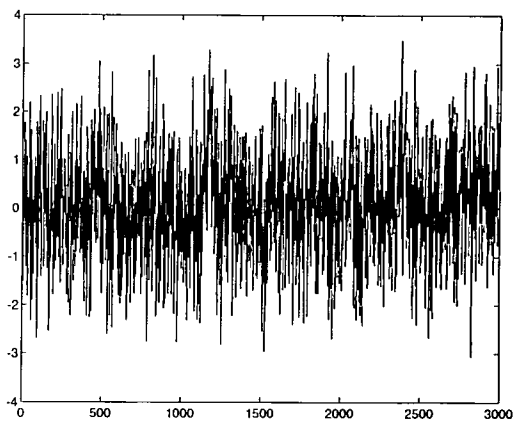
(a) FBM (H=0.5)



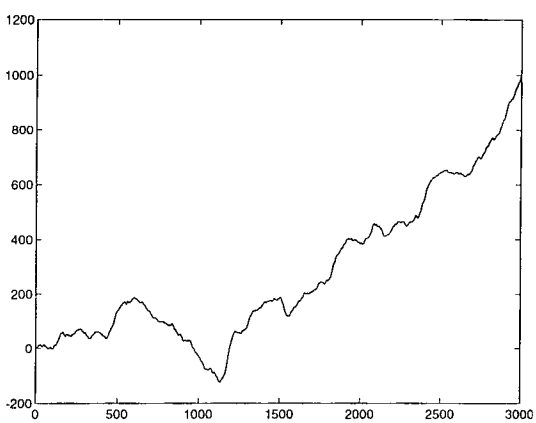
(b) FGN (H=0.5)



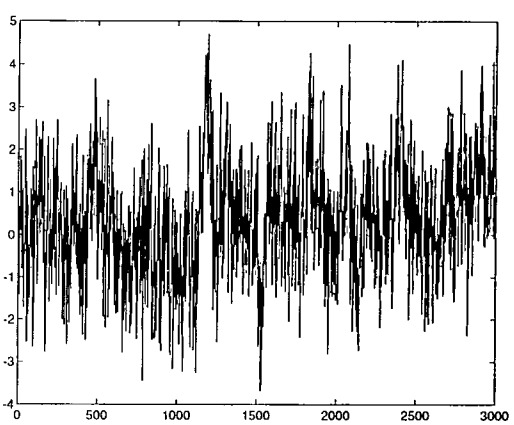
(c) FBM (H=0.7)



(d) FGN (H=0.7)



(e) FBM (H=0.9)



(f) FGN (H=0.9)

Figure 3.2: FBM and FGN generated by the dFGN algorithm with $M = 700, n = 10$.

$\lambda \leq \pi\}$ denote its spectral density function. Its autocovariance function and spectral density function can then be expressed as follows [MN68, Cox84, LTWW94, ST94]

$$\gamma(t) = \frac{\sigma_0^2}{2}(|t+1|^{2H} - 2|t|^{2H} + (t-1)^{2H}),\ t \in Z \tag{3.42}$$

$$S(\lambda) = \frac{\sigma_0^2}{C_2^2(H)}|e^{i\lambda} - 1|^2 \sum_{k=-\infty}^{\infty} \frac{1}{|\lambda + 2\pi k|^{2H+1}},\ -\pi \leq \lambda \leq \pi, \tag{3.43}$$

where $C_2(H) = (\frac{\pi}{H\Gamma(2H)\sin(H\pi)})^{1/2}$. Thus, for $H \neq 1/2$,

$$\gamma(k) \sim \sigma_0^2 H(2H-1)k^{2H-2} \text{ as } k \to \infty, \tag{3.44}$$

$$S(\lambda) \sim \sigma_0^2 C_2^{-2}(H)|\lambda|^{1-2H} \text{ as } \lambda \to 0. \tag{3.45}$$

From these two approximations when $1/2 < H < 1$ the $\gamma(k)$ tends to zero so slowly that $\sum_{k=-\infty}^{\infty}\gamma(k)$ diverges (LRD) and $S(\lambda)$ diverges at the origin ($1/f$-noise). Therefore, as expected, FGN has the fundamental properties of self-similar stochastic processes. Thus, FGN can be view as an idealization or as a reasonable first approximation of a more complex structure.

3.4.2 Fractional Autoregressive Integer Moving Average Model

Autoregressive moving average (ARMA) processes are often used for modeling empirical time series which can be defined in terms of linear difference equations with constant coefficients. When a given set of observations exhibits non-stationary or has not a rapidly decreasing autocorrelation function, autoregressive integrated moving-average (ARIMA) processes transform the data to a new series with the properties. A long memory process can be approximated by an ARMA process, however, the order of polynomials required to achieve a reasonably good approximation may be so large as to make parameter estimation extremely difficult. Thereafter, Fractional ARIMA (FARIMA) was introduced by [GJ80] and [Hos81], independently. The new version of the ARIMA processes extend the order of transformation polynomial from a natural number to a positive real number.

A process $\{X_k : k = 0, 1, 2, \cdots\}$ is said to be an $ARIMA(p,d,q)$ process, where p and q are nonnegative integers, with $d \in (-0.5, 0.5)$ if $\{X_k\}$ is stationary and satisfies the difference equations,

$$\phi(B)\nabla^d X_k = \theta(B)Z_t, \tag{3.46}$$

where $\{Z_t\}$ is white noise and ϕ, θ are polynomials of degrees p, q respectively. The $ARIMA(p,d,q)$ process with $d \in (-0.5, 0.5)$ is also called $FARIMA(p,d,q)$. In Eq. (3.46), ∇^d is the d^{th} order difference operator. The first order operator is defined as follows:

$$\nabla = X_k - X_{k-1} = (1-B)X_t, \tag{3.47}$$

where B is the backward shift operator,

$$BX_k = X_{k-1}. \tag{3.48}$$

Polynomials in B and ∇ are manipulated in precisely the same way as polynomial functions of real variables. Thus,

$$\nabla^d = (1-B)^d X_k = \sum_k \binom{d}{k} (-B)^k \tag{3.49}$$

with $\binom{d}{k}(-1)^k = \Gamma(-d+k)/(\Gamma(-d)\Gamma(k+1))$. The polynomials of ϕ and θ with the p^{th} and q^{th} degrees are written as

$$\phi(z) = 1 - \phi_1 z - \cdots - \phi_p z^p, \tag{3.50}$$

and

$$\theta(z) = 1 + \theta z + \cdots + \theta_q z^q. \tag{3.51}$$

For a $FARIMA(p,d,q)$ process it has been shown that if the solution $\{X_k\}$ is causal and invertible then its autocorrelation function $\{\rho_k\}$ satisfies, for $d \neq 0$,

$$\rho_k \sim Ck^{2d-1} \text{ as } k \to \infty, \tag{3.52}$$

where $C > 0$ [BD91].

$FARIMA(p,d,q)$ processes are more flexible with regard to the simultaneous modeling of the short-term and long-term behavior of a time series than FGN, mainly because the latter, with only three parameters μ, σ^2, and H, is not capable of capturing the wide range of low-lag correlation structures which are encountered in practice. However, FARIMA processes are highly dependent on the orders of polynomials. It is unlike FBM which has more physical meaning and several fast approximations. Moreover, the two additional parameters, p and q, make this model more flexible but also difficult to be used in practice because two more parameters need to be estimated. In addition, many of the asymptotic relations stated in Sec. 3.5 holds for finite sample sizes because FGN is the increment of the self-similar process FBM [TTW95].

3.4.3 Chaotic Maps

Chaos can be found in the trajectories of a dynamic system. There is no general definition for chaos, but mathematicians agree that for the special case of iteration of transformations there are three common characteristics of chaos: sensitive dependence on initial condition; mixing; and dense periodic points [Dev87, PJS92, Bar93, KJ94, ESP94, BL97, RBL97]. The link between fractals and chaos theory is that fractals are simple rules for generating infinite complexity, but a complexity which is bounded within its attractor. Chaotic maps are simple dynamic systems whose time evolution is described by a knowledge of the initial state and a set of dynamic laws.

Let $f(\cdot)$ be a one-dimensional nonlinear systems in which the state variable x_n evolves over time according to the iterative function, $x_{n+1} = f(x_n)$. If the initial value is decided, then the trajectories of the dynamic system can be determined. When using the system to model traffic sources, assume an arrival (or a batch of arrivals) is generated while the state x_n is in an active period which is decided by a pre-given

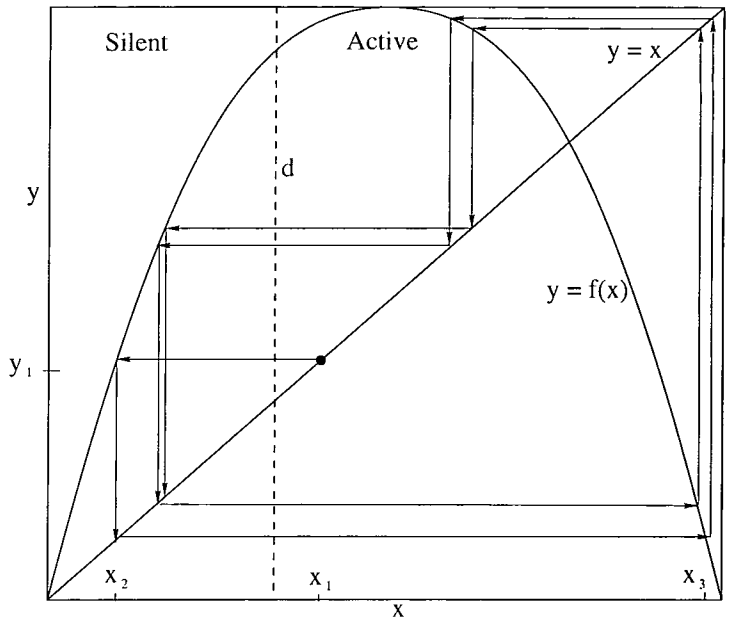


Figure 3.3: A simple example of a chaotic map.

threshold d. For every iteration this dynamic system will issue an arrival (or batch of arrivals) while x_n locates in the active interval. Fig. 3.3 illustrates an example of how the iterative dynamic system works. In this diagram the initial state x_1 hits the dynamic system $F(x_1)$ on y_1. Then, let $y_1 = x_2$ and repeat the iteration. When x_n is less than the threshold value d, an arrival is generated. The multi-threshold scheme is also considered as an alternate to the simplified active/silent approach. In this proposal every interval decides it own rule on how many arrivals will be generated.

The above example illustrates that the selection of dynamic systems will insensitively affect the results. Two of the simplest classes of chaotic maps, piecewise linear segments and intermittency maps, have been discussed in [ESP94]. The Bernoulli shift is a particularly simple dynamic system with piecewise linear segments, which is defined as follows:

$$x_{n+1} = f(x_n) = \begin{cases} \frac{x_n}{d}, & 0 < x_n < d, \\ \frac{x_n - d}{1-d}, & d \leq x_n < 1. \end{cases} \tag{3.53}$$

This dynamic system has been used to show that the invariant density of this map is uniform. Let the state x_n exceed the threshold as the active interval, then the probability of generating an arrival is $1-d$. Therefore, the active and silent periods are geometrically distributed. While the piecewise segments form a linear system, the intermittency maps are nonlinear. Let the system be defined as follows:

$$x_{n+1} = f(x_n) = \begin{cases} x_n + cx_n^m + \varepsilon, & 0 < x_n \leq d, \\ \frac{x_n - d}{(1-d)}, & d < x_n < 1, \end{cases} \tag{3.54}$$

where $c = (1 - \varepsilon - d)/d^m$. The long term process of this deterministic nonlinear case has been shown to be related to the heavy-tailed property of fractals.

Although chaotic maps can be used to generate self-similar traffic directly, there are still several problems associated with them. First, different chaotic maps leads to different traffic streams. In some cases the trajectories will converge to a certain value, called an *attractor*. Sometimes, the trajectories are

periodic. Then, the trajectories are seemingly random rather than stochastic. When the initial value is decided, the whole trajectories can be determined. Finally, the relation between the similarity dimension and the key parameter of stochastic self-similarity, the Hurst parameter, is still vague. Although in some cases the similarity dimension is equal to the reciprocal of the Hurst parameter, the similarity dimension is not uniquely defined.

3.5 Statistical Analysis Tools for Long-Range Dependence

To estimate the self-similar parameter or the intensity of LRD in a time series, various methods, such as the VT plot, differencing the variance method, absolute values of the aggregated series, Higuchi's method, residuals of regression, the R/S statistic, periodogram method, modified periodogram method, correlogram method, variogram, and Whittle's approximated maximum likelihood estimator (MLE), are available [Ber94, TTW95].

Almost all of the estimators are heuristic and make use of the samples themselves, their variance, correlation, and periodogram. The R/S statistic, the VT plot, periodogram method, and Whittle's approximate MLE are the most commonly used estimators. The former three estimators are all heuristic approaches while the last one uses a more rigorous mathematical derivation. In practice, the periodogram method can not use whole samples for the regression in the frequency domain because the approximation only holds for frequencies close to the origin, referring to the L'-2 property of self-similar stochastic processes. Whittle's approximate MLE is also based on periodogram and obtains the Hurst values by minimizing a log-likelihood function. Therefore, the periodogram method is not included in this thesis. In addition to previously discussed estimators, the Q-Q plot is used to visually assess the goodness-of-fit of sample traces.

3.5.1 The Q-Q Plot

The Q-Q plot is a graphical technique that is commonly used to visually assess goodness-of-fit and estimate location and scale parameters. The method is based on the following sample observation [MS86]: Let

$$Y_{(1)} \leq Y_{(2)} \leq \cdots \leq Y_{(n)} \tag{3.55}$$

be the order statistics from n identically independent distributed (i.i.d.) observations and let $F(x|\mu,\sigma)$ be the corresponding accumulated probability function (cdf) where μ and σ are unknown location and scale parameters, respectively.

Since μ and σ are location and scale parameters, the cdf can be expressed as

$$F(x|\mu,\sigma) = G(\frac{x-\mu}{\sigma}) = G(z), \tag{3.56}$$

where $z = (x-\mu)/\sigma$ is referred to as the standardized variable and $G(z)$ is the cdf of the standardized random variable. Thus, the expectation of the ith order observation is

$$E[X_{(i)}] = \mu + \sigma E[Z_{(i)}] = \mu + \sigma m_i \tag{3.57}$$

where $Z_{(i)}$ is the ith order statistic from the standardized distribution and $m_i = E[Z_i]$. In the probability case, the n i.i.d. observations are uniformly distributed on $[0,1]$. For a fixed $p \in (0,1)$, the pth population quantile for X is $Q_F(p)$, where

$$Q_F(u) = F^{-1}(u|\mu,\sigma) = \inf\{x : F(x|\mu,\sigma) \geq u\}, \quad 0 < u < 1, \tag{3.58}$$

and F^{-1} is the inverse function of F. Hence,

$$Q_F(p) = \mu + \sigma Q_G(p) = \mu + \sigma G^{-1}(p). \tag{3.59}$$

The order statistic $X_{(i)}$ can be regarded as an estimate of its mean, or of the pth quantile of $F(x|\mu,\sigma)$, where p is an appropriate probability.

In constructing a probability plot, the $X_{(i)}$ is on the y-axis and its mean m_i is on the x-axis. When the sample is extracted from the cdf $F(x|\mu,\sigma)$, the points will tend to fit on a straight line with intercept μ and slope σ. The quantiles solve the difficulty of computing the mean of the order statistics because $F(x|\mu,\sigma)$ is easy to invert.

3.5.2 The R/S Statistic

The R/S statistic was developed by the famous hydrologist Hurst who investigated the question of how to regulate the flow of the Nile river [Hur51, Fed88, Ber94]. The problem he studied was how to determine the design of an ideal reservoir based upon the given record of observed discharges from Lake Albert. Suppose a capacity of an idea reservoir for the time span between t and $t+k$. To simplify matters, assume the time is discrete and no storage losses. The capacity of the reservoir at time $t+k$ is the same as the initial time t. Let X_i denote the inflow at time i and $Y_j = \sum_{i=1}^{j} X_i$ the cumulative inflow up to time j. The ideal capacity can be shown to be equal to

$$R(t,k) = \max_{0\leq i\leq k}\left[Y_{t+i} - Y_t - \frac{i}{k}(Y_{t+k} - Y_t)\right] - \min_{0\leq i\leq k}\left[Y_{t+i} - Y_t - \frac{i}{k}(Y_{t+k} - Y_t)\right]. \tag{3.60}$$

$R(t,k)$ is called the *adjusted range*. In order to study the properties that are independent of the scale, $R(t,k)$ is normalized by

$$S(t,k) = \sqrt{k^{-1}\sum_{i=t+1}^{t+k}(X_i - \bar{X}_{t,k})^2}, \tag{3.61}$$

where $\bar{X}_{t,k} = k^{-1}\sum_{i=t+1}^{t+k} X_i$. The ratio

$$R/S = \frac{R(t,k)}{S(t.k)} \tag{3.62}$$

is called the *rescaled adjusted range* or *R/S statistic* where $R(t,k)$ is the *adjusted range* and $S(t,k)$ is the standard deviation.

If an FGN process intends to estimate its degree of the Hurst parameter, the R/S analysis is a heuristic graphical approach. However, Eq. (3.60) should then be modified as

$$R(t,k) = \max_{0\leq i\leq k}\left[\sum_{j=1}^{i} X_{t+j} - i\bar{X}_{t,k}\right] - \max_{0\leq i\leq k}\left[\sum_{j=1}^{i} X_{t+j} - i\bar{X}_{t,k}\right] \tag{3.63}$$

As $k \to \infty$ the rescaled adjusted range represents the relation

$$E[R(t,k)/S(t,k)] \sim ck^H. \tag{3.64}$$

Given a sample of n observations $\{X_i\}$, subdivide the whole sample into m_k non-overlapping blocks and compute the rescaled adjusted range for each block. Thus, for a given value of k there are as many as m_k R/S-statistic values. Next, plotting $\log(R(t,k)/S(t,k))$ against $\log k$, the R/S plot is formed. Use the plotting samples draw a straight line to fit all the data. The slope drawn is the estimated Hurst parameter.

3.5.3 The Variance-Time Plot

From a statistical point of view, the most salient feature of self-similar processes is that the variance of the arithmetic mean decreases more slowly than the reciprocal of the sample size [LTWW94, Ber94]. This fact means that its behavior has $n^{-\beta}$ for some $\beta \in (0,1)$ rather than n^{-1} for the processes whose aggregated series converge to second-order pure noise, where $H = 1-\beta/2$. In fact, a specification of the autocorrelation function satisfying the assumptions of LRD processes has the property

$$Var[\bar{X}_n] \sim c\, n^{2H-2}, \tag{3.65}$$

where $c > 0$.

Therefore, the following method for estimating the Hurst parameter is suggested: first divide the sample path $\{X_i\}$ into a sufficient number of sub-series of length k, said $\bar{X}_j(k)$, where $j = 1,2,\cdots,m_k$. The length of sub-series could be any integer in the range $2 < k < n/2$. $\bar{X}_j(k)$ is the sample mean of the jth sub-series and the overall mean

$$\bar{X} = m_k^{-1} \sum_{j==1}^{m_k} \bar{X}_j(k). \tag{3.66}$$

Then, for each k calculate the sample variance of the sample means $\bar{X}_j(k)(j = 1,2,\cdots,m_k)$:

$$\sigma^2(k) = (m_k - 1)^{-1} \sum_{j=1}^{m_k} (\bar{X}_j(k) - \bar{X}(k))^2. \tag{3.67}$$

Using $\log\sigma^2(k)$ against $\log k$ the VT plot is obtained. Finally, drawing a least squares line to fit the sample points, the slope is β and the value of the Hurst parameter is obtained.

3.5.4 Whittle's Approximate MLE

While the R/S statistic and the VT plot tests focus on the time domain, the Whittle's approximate estimation examines the properties in the frequency domain. Maximum likelihood estimates (MLE) are based on the periodogram. Suppose all parameters except $Var(X)$ and H of a sample of a self-similar process X are known. Let $f(\lambda;\theta)$ be the power spectrum of X when normalized to have unit variance, and $I(\lambda)$ be the periodogram of X. To estimate H, find θ that minimizes the following equation:

$$g(\theta) = \int_{-\pi}^{\pi} \frac{I(\lambda)}{f(\lambda;\theta)} d\lambda, \tag{3.68}$$

where θ is a vector, $(\sigma^2, H, \theta_3, \cdots, \theta_k)$ in which $\theta_3, \cdots, \theta_k$ model the short-range dependence structure of the process. After deriving Eq. (3.68), an approximate MLE of θ is obtained by minimizing the log-likelihood function [Ber94]

$$L_W(\theta) = \frac{1}{2\pi}\int_{-\pi}^{\pi} \log f(\lambda,\theta)d\lambda + \frac{\mathbf{x}^t\mathbf{A}(\theta)\mathbf{x}}{n} \tag{3.69}$$

with respect to θ. In Eq. (3.69) $\mathbf{x}$ is the vector of random variables of a joint distribution function, $\mathbf{x}^t$ is the transpose of the vector $\mathbf{x}$, and $\mathbf{A}$ is the covariance matrix of the distribution function.

For statistical robustness Whittle's MLE faces two problems: one is caused by its derivation from Gaussianity; and the other is due to its assumption of a commonly encountered model spectrum. The first can be overcome by transforming the data to obtain the desired marginal distribution. There are several proposals to cope with the second problem, such as estimating H from the periodogram ordinates at low frequencies or bounding the periodogram at high frequencies. In practical usage for large data sets, the method of aggregation is an alternative and efficient way to tackle the problem. It has been proved in [Ber94, LTWW94] that FGN can be modeled by the aggregated process $X^{(m)}$ $(m \geq 1)$ defined as

$$X_j^{(m)} = m^{-H}L^{-1/2}(m) \sum_{i=(j-1)m+1}^{mk} (X - i - E[X_i]),\ j \in \{1, 2, \cdots, [n/m]\}, \tag{3.70}$$

for a sufficiently large m. Therefore, the reasonable aggregated size m is chosen such that the sample size of the corresponding series $X^{(m)}$ is not less than about 100.

3.6 Summary

Self-similar traffic models are based on an essential property, self-similarity, which is an aspect of fractals. A fractal is a structure that has a non-integer value of dimension. There are a considerable number of practical interpretations which attempt to capture the fractal phenomenon. Self-similarity is a very important phenomenon which illustrates deterministic fractals. It is often explained as being equivalent to the existence of a multilevel hierarchy of underlying mechanisms.

The Hurst parameter plays the key role in moving fractals from a deterministic version to a stochastic version. Hurst measured the difference between the maximum and minimum accumulated influx of a reservoir during a sampling period. After normalizing by standard deviation, the rescaled statistic value, R/S, is found to be proportional to a power law with an exponent, the Hurst exponent H.

By using the Hurst exponent to represent the degree of self-similarity, self-similar stochastic processes attain three important properties: its autocorrelation function decays hyperbolically as the lag increases; its spectral density appears as a pulse at the origin; and its variance of the arithmetic mean decays more slowly than the reciprocal of the sample size.

In order to deal with fractal properties of packet traffic (i.e., with properties such as self-similarity) LRD, infinite variance, and fractal dimensions observed in actual packet traffic measurements, two major approaches, self-similar processes and deterministic models using nonlinear chaotic maps, have been considered. Both methods emphasize the need for models with a small number of parameters, where

every parameter can be given a physically meaningful interpretation, and the ability to generate synthetic traffic easily and quickly.

Self-similar stochastic processes can be represented by exactly self-similar models, such as fraction Gaussian noise, and asymptotically self-similar models, such as FARIMA processes. These two processes can be used to fit hour-long traces of the Ethernet traffic very accurately. Associated with FGN is FBM, which is simply the integrated version of FGN, i.e., a FBM process is simply the sum of FGN. FBM supports many fundamental properties and it can more easily interpret the physical meanings. To generate fractal traffic, it is necessary to understand the foundation of FBM/FGN and generate FBM/FGN samples. At times, there are indications of a particular short-range dependence structure in the traffic measurements, in which case asymptotically self-similar models are more appropriate. Asymptotically self-similar processes, such as the FARIMA processes, are more suitable for the prediction of bursty incoming data on networks but are more difficult to use because of their additional parameters. Unlike FBM and FARIMA, chaotic maps are used to generate traffic directly. However, the usage of dynamic systems tend to generate deterministic results rather than stochastic ones.

In order to measure the Hurst parameter, three Hurst estimators have been introduced for use in the accurate assessment of FBM samples and fractal traffic. There are a great number of Hurst estimators. Most of them are derived from the fundamental properties of the self-similar stochastic process. The three most commonly used estimators have been described. The R/S statistic makes use of the scaling property in self-similarity. The VT plot derives from the properties of LRD. Whittle's MLE takes account of the optimization of the logarithm of the periodogram. In addition, a visual assessment tool has been presented to test whether the generated fractal samples are Gaussian.

In the next chapter, the investigation focuses on how the algorithms generate FBM sample traces and their evaluation.

Chapter 4

Algorithms for Generating Fractional Brownian Motion

The self-similar stochastic process is a mathematical model which is used to represent the characteristics of the various types of traffic. This process is very difficult to analyze due to its continuous and non-differentialable characteristics. In a similar way to ordinary Brownian motion and Gaussian noise, the increments of the self-similar process form a stationary process with LRD. The FBM and the FARIMA models can be used to represent these processes with LRD. It has been proved that FBM processes are more appropriate because they need fewer parameters and convey more physical meaning.

This chapter discusses the algorithms which generate FBM samples. It consists of five major parts. Firstly, six algorithms whose function is to generate FBM sample traces are introduced. Secondly, assessment of their efficiency and accuracy is carried out using time consumption, the three Hurst estimators, and the Q-Q plot, respectively. Thirdly, comparisons of these algorithms are provided. Fourthly, a further look at the effect of changing the parameters of the dFGN algorithm are studied. Finally, the discussion is focused on the suitability of using the VT plot.

4.1 Simulation Algorithms

A great number of algorithms have been proposed for generating FBM or FGN samples. Most of them are derived from the definition and properties of the FBM processes. Some are incorporated with other techniques, such as the important sampling technique [HDLK95a] and the wavelet transformation [Fla92]. The number of customers in an $M/G/\infty$ queue asymptotically represent the self-similar process [Cox84, LTWW94, PF95]. Different approximation of the definition of the FBM processes has used to generate more accurate simulations with $H < 0.5$ [RP94].

In this section six algorithms, which has been divided into three types, are chosen for further investigations. The discrete FGN (dFGN) and fast FGN (fFGN) algorithms directly are derived from the definition of the FBM processes. The random midpoint displacement (RMD) and successive random addition (SRA) algorithms generate interpolated samples using the scaling property. The Weierstrass-Mandelbrot (WM) function and Whittle's approximate approach generate samples in the frequency domain rather

than in the time domain.

4.1.1 The dFGN Algorithm

From the definition of FBM the discrete version of FBM (dFBM) can be approximately expressed as [MW69]

$$B_H(k) - B_H(0) \simeq \frac{1}{\Gamma(H+1/2)} \sum_{i=-\infty}^{nt} K(k-\frac{i}{n}) G(i,1/n), \tag{4.1}$$

where $G(i,1/n)$ is a discrete random Gaussian variable with mean zero and variance $1/n$. The kernel function K is given in Eq. (3.38). Using the scaling property the Gaussian random variable can be expressed as $G(i,1/n) = n^{-1/2}G(i)$. Here $G(i)$ is a discrete time random variable with mean zero and unit variance. Actually, for every unit time interval a dFBM process divides the time unit into n subintervals. The subsamples of the whole time scale are still Gaussian random variables with mean zero but the variance changes from unity to $1/n$. Nevertheless, the dFBM process makes the sum go to infinity. A slight modification to the kernel is used to make the sum convergent.

Due to the long-memory property of the FBM process, any calculation of B_H must use a finite number of terms. The increment of FBM is considered and the sums can cover a range M in the integer time t. Thus, the discrete approximation is

$$B_H(k) - B_H(k-1) = \frac{1}{\Gamma(H+1/2)} \sum_{i=n(t-M)}^{nk-1} K(k-\frac{i}{n}) n^{-1/2} G(i), \tag{4.2}$$

This equation is referred to as the *discrete-time fractional Gaussian noise (dFGN)*. Every calculation of the dFGN is effected by the kernel function in which the last n Gaussian random variables multiply the factor $(k-\frac{i}{n})^{H-1/2}$, whereas the Gaussian random variables with indexes from $n(k-M)$ to $n(k-1)$, multiplying the factor $\{(k-\frac{i}{n})^{H-1/2} - (-\frac{1}{n})^{H-1/2})\}$. With a change of the summation variable and a rearrangement of terms the discrete FBM increments are given by

$$\begin{aligned} G_H(k) \equiv \Delta B_H(k) &= B_H(k) - B_H(k-1) \\ &= \frac{n^{-H}}{\Gamma(H+1/2)} \left\{ \sum_{i=1}^{n} (i)^{H-1/2} G(nk-i) \right. \\ &\qquad \left. + \sum_{i=1}^{n(M-1)} \{(n+i)^{H-1/2} - (i)^{H-1/2}\} G(nk-n-i) \right\}. \end{aligned} \tag{4.3}$$

With the procedure given in Eq. (4.3) a sequence of the dFGN may be generated from a sequence of Gaussian random variables. This is a sliding average over the Gaussian process with a power-law weight function. Since only M integer time steps are included in the summation it follows that for integer times $t >> M$, the increments will become independent Gaussian processes. It is clear that the algorithm is inefficient since the sums of nM terms have to be evaluated for each increment in B_H.

4.1.2 The Fast Fractional Gaussian Noise Algorithm

The dFGN algorithm is based on finite weighted moving averages in which the number of Gaussian variables to average is roughly proportional to the size of the desired sample. It is obvious that this

approximation produces a practical drawback. Mandelbrot proposed a different faster FGN generator in 1971 [Man71]. In this algorithm he made use of Markov-Gaussian processes to construct FGN sample traces. The procedure is derived from Eq. (3.45), the autocovariance function of FGN.

The fFGN process $\vec{G}_H(t)$ consists of a finite sum of Markov-Gauss processes approximating the autocovariance function of FGN. A sample trace $\vec{G}_H(t)$ with sample size n can be constructed as the sum of a high frequency Markov-Gauss term $X_h(t,H)$ and a low frequency term $X_l(t,H)$ which is a weighted sum of a number $N(T)$ of independent Markov-Gauss processes. In this algorithm two new parameters, the base $B > 1$ and the quality factor Q are introduced. The low frequency term is defined as

$$X_l(t,H) = \sum_{k=1}^{N(n)} W_k X(t,H|MG), \tag{4.4}$$

where $X(t,H|MG)$ is the Markov-Gauss process of variance 1 and covariance c_n^2 with $c = \exp(-B^{-n})$. W_k is the weighting function given as

$$W_k^2 = \frac{H(2H-1)(B^{1-H} - B^{-1+H})}{\Gamma(3-2H)} B^{-2(1-H)k}, \tag{4.5}$$

where $\Gamma(x)$ is the gamma function. The term $N(n)$ satisfies

$$N(n) = \lceil \log(Qn)/\log B \rceil, \tag{4.6}$$

in which $\lceil \cdot \rceil$ is the ceiling function. To define the Markov-Gauss process a sequence of ordinary Gaussian variables $G_k(t)$, with mean zero and univariance, are required to generate a sample path from $t = 1$ to $t = n$. The Markov-Gauss process is defined as follows:

$$\begin{aligned} X(1,c_k|MG) &= G_k(1), \\ X(2,c_k|MG) &= c_k X(1,c_k|MG) + (1-x_k^2)^{1/2} G_k(2), \\ &\vdots \\ X(t,c_k|MG) &= c_k X(t-1,c_k|MG) + (1-x_k^2)^{1/2} G_k(t), \\ &\vdots \end{aligned} \tag{4.7}$$

Each sample point requires $N(n)$ terms of weighted Markov-Gauss processes. As $B \to 1$ and $Q \to \infty$, the value of $N(n)$ increases and the quality of the low frequency approximation improves. To minimize its approximating error Mandelbrot compiled a table to evaluate the quality factor Q. From this it was found as $H = 0.95$ it is better to choose the quality factor $Q > 14.6$. In addition, it was recommended to use a base, B, of 2, 3, or 4.

The high frequency term $X_h(t,H)$ is defined as

$$X_h(t,H) = 2^{2H-2} - 1 + \sum_{k=1}^{N(n)} W_k(1-c_k) - \frac{B^{-(1-H)} H(2H-1)}{\Gamma(3-2H)}. \tag{4.8}$$

This term uses the recommended Markov-Gauss process, its variance, depending on H and B,and its correlation of lag. However, when $H \to 1$, Eq. (4.8) can be reduced to

$$X_h(t,H) = 1 - \frac{H(2H-1)}{\Gamma(3-2H)} B^{-(1-H)}. \tag{4.9}$$

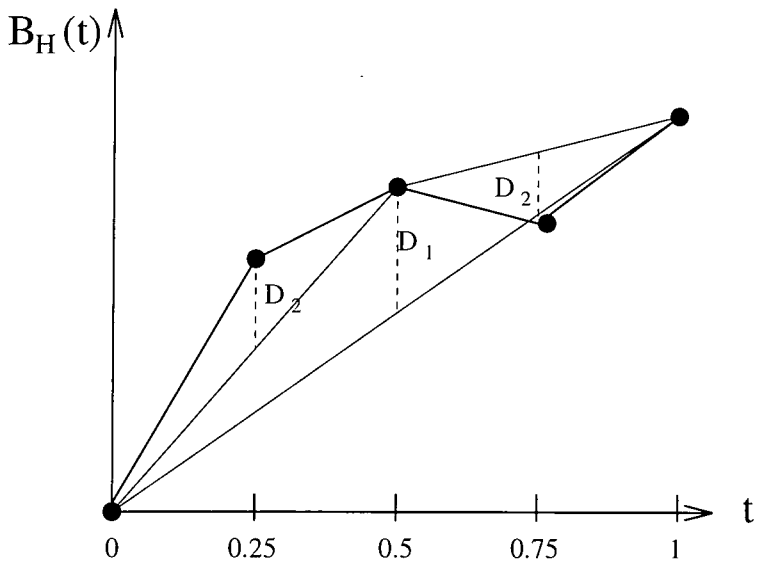


Figure 4.1: A two-stage example of the RMD algorithm.

4.1.3 The Random Midpoint Displacement Algorithm

The random midpoint displacement (RMD) method was originally developed for computer graphics [FFC82, Vos85a, Vos85b, Vos88, Sau88, PJS92, LEWW95]. The idea is derived from a generating technique that N. Wiener applied to generate ordinary Brownian motion. One of the important properties of the RMD algorithm is that it can be generalized to several dimensions usefully. Two facts help in the design of this approximation algorithm. The first is that FBM is self-similar. The second is that a formula exists for the conditional expectation of $B_H(t), 0 \leq t \leq 1$, knowing $B_H(0) = 0$ and $B_H(1) = 1$,

$$E[B_H(t)|B_H(1)] = 1/2(t^{2H} + 1 - |t-1|^{2H}). \tag{4.10}$$

When $t = 1/2$, the right-hand side becomes $1/2$ independent of H. The concept of the RMD method is to recursively subdivide the interval and construct the values of the process at the midpoints from the values at the end points.

If a process Y_t is to be computed n times within $t = [0,1]$, then the computation starts by setting $Y_0 = 0$ and sets Y_1 as a sample of a Gaussian random variable. Next $Y_{\frac{1}{2}}$ is constructed as the average of Y_0 and Y_1, i.e. $\frac{1}{2}(Y_1 + Y_0)$ plus a displacement. Fig. 4.1 shows the first two stages of implementation of this algorithm. Executing this procedure recursively n times, a sample trace with 2^n points is obtained. If Y_t were truly a FBM process, then the midpoint displacement $Y_{\frac{a+b}{2}} - \frac{Y_a+Y_b}{2}$ for the interval $[a,b]$ would be independent of the increment $Y_b - Y_a$ over the whole interval. The displacement would be a Gaussian random variable with mean zero and its variance would follow the scaling property.

Considering a simple FBM its mean square increment for points separated by a time $\Delta t = 1$ is σ^2. Applying the scaling property the value can be expressed as $E[|B_H(t) - B_H(0)|^2] = t^{2H} \cdot \sigma^2$. For convenience let $B_H(0) = 0$, the positions at $t = \pm 1$ are chosen as samples of a Gaussian random variable with variance σ^2. Given these initial conditions, one defines the midpoints at

$$B_H(1/2) = \frac{1}{2}[B_H(0) + B_H(1)] + D_1, \tag{4.11}$$

where D_1 is a Gaussian random offset with zero mean and variance σ_1^2 in order to satisfy the condition that the increments from 0 to $1/2$ must follow the expression,

$$E[|B_H(t_2) - B_H(t_1)|^2] \propto |t_2 - t_1|^{2H}. \tag{4.12}$$

Therefore,

$$\sigma_1^2 = \frac{\sigma^2}{2^{2H}} - \frac{\sigma^2}{4} = \frac{\sigma^2}{2^{2H}}[1 - 2^{2H-2}]. \tag{4.13}$$

The first term describes the variance of Eq. (4.12) while the second term represents the fluctuations already in $B_H(1) - B_H(0)$ because of the previous stage. As $H \to 1$, $\sigma_1^2 \to 0$, no new fluctuations are added at smaller stages, and $B_H(t)$ remains a collection of smooth line segments connecting the starting points. At the second stage, $B_H(1/4) = 1/2[B_H(0) + B_H(1/2)] + D_2$ and $B_H(3/4) = 1/2[B_H(1/2) + B_H(1)] + D_2'$ where the displacements D_2 and D_2' have the same statistical characteristics. Their variance can be expressed as

$$\sigma_2^2 = \frac{\sigma^2}{4^{2H}} - \frac{\sigma_1^2}{4} = \frac{\sigma^2}{4^{2H}}[1 - 2^{2H-2}]. \tag{4.14}$$

Therefore, at the nth stage the length scale has decreased to $1/2^n$ and a random Gaussian offset D_n is added to the midpoints of the $(n-1)$th stage with variance

$$\sigma_n^2 = \frac{\sigma^2}{(2^n)^{2H}}[1 - 2^{2H-2}] = 2^{-2H}\sigma_{n-1}^2 \tag{4.15}$$

As expected for a FBM, at length scale $r = 1/2^n$ one adds randomness with mean square variations varying as r^{2H}. After rearranging Eq. (4.15) we can obtain

$$\sigma_n = 2^{-H}\sqrt{1 - 2^{2H-2}}\sigma = 2^{-H}\sigma_{n-1} \tag{4.16}$$

Following the relation it is more convenient to set the initial value

$$\sigma_0 = \sqrt{1 - 2^{2H-2}}\sigma. \tag{4.17}$$

The first two stages in the random midpoint technique are depicted in Fig. 4.1 where $B_H(1)$ is generated by ordinary Gaussian noise and $B_H(0)$ is chosen as 0. This concept is easy to implement using a recursive method which accelerates the simulation speed. Therefore, to generate N points of sample traces requires only $O(N)$ operations.

Although the process does produce a fractal, Mandelbrot criticized the sample paths generated by the RMD algorithm as not stationary for all H [Man82, Man83]. When a point at time t_i is fixed, its value does not change in any later stage. The following stages which occur after the stage determining time t_i produces two independent parts: one is $t > t_i$ and the other is $t < t_i$. This violates the basic property that the requirement of FBM is that the samples should be correlated, rather than independent, with $H \neq 1/2$. Consequently, points generated at different stages have different statistical properties in their neighborhood. In spite of its mathematical failings, the speed of midpoint displacement, and its ability to interpolate an existing curve make it a useful fractal algorithm for a number of applications.

4.1.4 The Successive Random Addition Algorithm

For the RMD algorithm, when the value at a point is determined it remains fixed. Only half of the sample points are determined more accurately at each successive stage. To overcome the shortcoming of the RMD algorithm, R. Voss proposed the successive random addition (SRA) algorithm in which offsets are added to all the points which have been generated, not just the midpoints [Vos85a, Vos85b, Vos88, Sau88, Fed88].

Assuming that $B_H(0) = 0$ and $B_H(1)$ can have a value that is generated by Gaussian random variable with variance σ^2, then the midpoint sample $B_H(1/2)$ can be expressed as

$$B_H(1/2) - B_H(0) = \frac{1}{2}(B_H(1) - B_H(0)) + D_{(1)}. \tag{4.18}$$

In continuous time the whole sample trace can be divided into infinite bisections, so the displacement $D_{(1)}$ is aggregated by infinite displacements which are Gaussian random variables with zero mean and variances that are derived in the RMD algorithm. Therefore, the offset $D_{(1)}$ can be expressed as

$$D_{(1)} = \sum_{k=1}^{\infty} D_k, \tag{4.19}$$

where D_k are Gaussian random variables with mean zero and variance σ_k^2. The variance of $D_{(1)}$ is

$$\sigma_{(1)}^2 = \sigma^2(1 - 2^{2H-2})2^{-2H}. \tag{4.20}$$

From Eq. (4.19) the relation of the variance can be expressed as

$$\sigma_{(1)}^2 = \sum_{k=1}^{\infty} \sigma_k^2. \tag{4.21}$$

From the same procedure the variance of the nth stage is

$$\sigma_{(n)}^2 = \sum_{k=n}^{\infty} \sigma_k^2 = \sigma^2(1 - 2^{2H-2})2^{-2nH}. \tag{4.22}$$

From Eqs. (4.21) and (4.22) the variance σ_n^2 can be expressed as

$$\sigma_n^2 = 2^{-2H}\sigma_{n-1}^2. \tag{4.23}$$

Therefore, it is more convenient to set the initial value as

$$\sigma_0^2 = \sigma^2(1 - 2^{2H-2})(1 - 2^{-2H}). \tag{4.24}$$

In some texts, the above equation is simplified to

$$\sigma_0^2 = \sigma^2\frac{1}{2}(1 - 2^{2H-2}). \tag{4.25}$$

The process of the SRA algorithm is illustrated in Fig. 4.2.

In addition, Voss also generalized the algorithm with a factor $0 < r < 1$. In the RMD and SRA methods the resolution improved by the factor $r = 1/2$ in each stage. For the generalized algorithm between two endpoints there are $1/r$ sample points generated. At nth stage a random displacement D_n

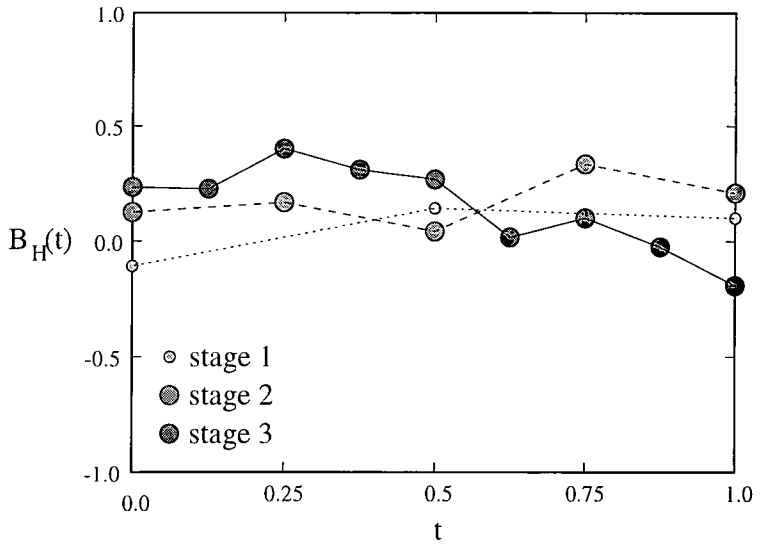


Figure 4.2: The process of successive random additions.

would be added to all interpolated points. The variance of the displacement D_n is proportional to $(r^n)^{2H}$. The additional parameter r will change the appearance of the fractal. The feature of the fractal controlled by r has been termed the *lacunarity*.

Based on the ideas of the RMD algorithm, the aggregated displacement must be the same as Eq. (4.15) above, i.e.

$$\sigma^2_{(n)} = \frac{1}{2}(1 - r^{2-2H}) \sum_{k=n}^{\infty} (r^k)^{2H}. \tag{4.26}$$

However, when the derivation of the generalized algorithm is carefully examined, the variance of the displacement D_n is oversimplified. Even at the first stage the variance of $B_H(kr) - B_H(0)$ should be related to the factor k, that is, the kth points counted from the origin will be generated in this stage. The displacement of higher stages will be more complicated because they relate to the lower stage. For example, at the second stage the sample $B_H(r^2)$ can be expressed as

$$B_H(nr + r^2) = kr[B_H((n+1)r) + B_H(nr)] + D_2. \tag{4.27}$$

As the factor of $B_H(nr)$ cannot totally be eliminated; the generalized algorithm is not considered to be of use in this study.

4.1.5 The Weierstrass-Mandelbrot Function

Because of the scale-invariant property of FBM, its Fourier transform should be scale-invariant. Furthermore, its Fourier coefficients should satisfy the appropriate power law in its statistical aspects. In a midpoint displacement process generated by the RMD or SRA method, every stage increases the highest spatial frequency by a factor $1/r = 2$. Thus, both are related to the WM function which is non-differentiable [BL80, Man83, Vos85a, Fed88, Vos88, Kor92, HS93, Ali96]. This function deals with the frequency domain property of FBM.

Brownian motion may also be considered as the cumulative displacement of a series of independent jumps. A pulse at time t_i causes a jump of magnitude A_i (a Gaussian random variable) in ordinary

Brownian motion $B(t)$. The response to each pulse, $A_iP(t-t_i)$, has the form of a step function with $P(t) = 1$ for $t > 0$ and $P(t) = 0$ for $t < 0$. Thus, $B(t)$ can also be written as the sum of the independent cuts at random (Poisson distributed) times t_i

$$B(t) = \sum_{i=-\infty}^{\infty} A_iP(t-t_i). \tag{4.28}$$

The time coordinate t may be replaced with angular position θ on a unit circle. The Brownian motion must be periodic, i.e. $B(\theta) = B(\theta+2\pi)$. In this case $B(t) = \int_{\theta-\pi}^{\theta} G(\theta')\,d\theta'$. The $P(\theta)$ corresponds to half circles, $P(\theta) = 1$ for $0 < \theta < \pi$ and $P(\theta) = 0$ for $\pi < \theta < 2\pi$, and Brownian motion on a circle becomes the summation of independent 'half-circles' uniformly distributed in θ_i. So, $B(\theta)$ can be expressed as

$$B(\theta) = \sum_{i=-\infty}^{\infty} A_iP(\theta-\theta_i). \tag{4.29}$$

Consider a collection of independent pulses occurring at random times t_i corresponding to an average rate $1/\tau$. Each pulse produces the profile $A_iP(t-t_1)$. The random function, $B(t) = \sum A_iP(t-t_i)$, will then have a spectral density

$$S(\omega) \propto \frac{1}{\tau}E[A^2]|P(\omega)|^2, \tag{4.30}$$

where $P(\omega)$ is the Fourier transform of $P(t)$, and $P(\omega) = \int P(t)\exp\{i\omega t\}\,dt$. For ordinary Brownian motion, each excitation by the white noise $G(t)$ produces the step function response $P(t)$ with $P(t) \propto 1/\omega$ and $S(\omega) \propto 1/\omega^2$. With a suitable power-law choice of $P(t)$, one can generate an approximation to FBM with any H.

An alternative way to directly generate FBM is to construct a random function with the desired spectrum. Since ordinary Brownian motion and FBM is scale-invariant, their Fourier transforms should also be scale-invariant. Thus the statistics of their Fourier coefficients should satisfy the appropriate power law. It has been shown that the distributions of the Fourier coefficients are both independent and Gaussian complex-valued random variables [HS93]. In addition, the phases of the Fourier coefficients are uniformly distributed over the interval $[0, 2\pi]$.

The WM function is such a continuous function which is non-differentiable and possesses no scale. The WM function is defined as

$$W(t) = \sum_{n=-\infty}^{\infty} b^{nH}(1-e^{i2\pi b^{-n}t})e^{i\phi_n}, \tag{4.31}$$

where $0 < H < 1$ is the Hurst parameter, b is a parameter with the range $(0, 1)$, and ϕ_n is an arbitrary phase. The function was extended by Mandelbrot based on the original Weierstrass function. The original version does not include the terms for $n < 0$ which add small spatial frequencies and large scale fluctuations. Whereas the frequencies b^{-n} form a 'Weierstrass spectrum' spanning the range zero to infinity in a geometric progression, Eq. (4.31) involves a linear progression of frequencies. The phases ϕ_n can be chosen to make W exhibit deterministic or stochastic behavior.

For the stochastic case, let ϕ_n be a set of random variables of the range 0 to 2π. Then, $W(t)$ is the sum to infinity of many contributions which have random phases, and so is a Gaussian random

function. However, the WM function involves many summations which makes the algorithm hard to use in practice. Although upper and lower thresholds can be set to eliminate this difficulty, this algorithm is still very time consuming. A better solution is to make use of the Fourier transform. Thus, an alternative form of the WM function can be expressed as

$$W(t) = \sum_{n=-\infty}^{\infty} A_n b^{nH} \exp(2\pi i b^{-n} t + i\phi_n), \tag{4.32}$$

where A_n is a Gaussian random variable with the same variance for all n. In this alternative version $W(t)$ is related to the random midpoint displacement and successive random addition methods when $b = 1/2$.

In order to generate FBM traces the infinity many summation should be truncated. Therefore, another form of the WM function to present FBM processes can be written as:

$$B_H(t) = \sum_{n=0}^{N} B_n \exp(i\phi_n), \tag{4.33}$$

where B_n are Gaussian random variables with mean zero and variance σ_n^2. The variance σ_n^2 can be obtained as follows. Let $S_{B_H}(f)$ be the spectral density of $B_H(t)$ and $B_H(f,T)$ be the Fourier transform of a specific sample of $B_H(t)$ for $0 < t < T$,

$$B_H(f,T) = \frac{1}{T}\int_0^T B_H(t)\exp(2\pi i f t)\,dt, \tag{4.34}$$

then

$$S_{B_H}(f) \propto T|B_H(f,T)|^2 \text{ as } T \to \infty. \tag{4.35}$$

An alternative characterization of the time correlations of $R_{B_H}(\tau)$ is given by the 2 point autocorrelation function

$$R_{B_H}(\tau) = E[B_H(t+\tau)B_H(t)] - E[B_H^2(t)]. \tag{4.36}$$

$R_{B_H}(\tau)$ provides a measure of how the fluctuations at two times separated by τ are related. In many cases $R_{B_H}(\tau)$ and $S_{B_H}(f)$ are related by the Wiener-Khintchine relation

$$R_{B_H}(\tau) = \int_0^{\infty} S_{B_H}(f)\cos(2\pi f\tau)\,df. \tag{4.37}$$

For certain power laws for $S_{B_H}(f)$, $R_{B_H}(\tau)$ can be calculated exactly. Thus, for

$$S_{B_H}(f) \propto \frac{1}{f^{\beta}} \text{ with } 0 < \beta < 1,\ R_{B_H}(\tau) \propto \tau^{\beta-1}. \tag{4.38}$$

Moreover, $R_{B_H}(\tau)$ is directly related to mean square increments of FBM

$$E[|B_H(t+\tau) - B_H(t)|^2] = 2\{E[B_H^2(t) - R_{B_H}(\tau)]\}. \tag{4.39}$$

$S_{B_H} \propto 1/f^{\beta}$ approximately corresponds to $R_{B_H}(\tau) \propto \tau^{1-\beta}$ and $2H = \beta - 1$ based on Eqs. (4.39) and (4.12). From the above derivation, it is known that

$$S_{B_H} \propto 1/f^{\beta} = 1/f^{2H+1} = n^{-(2H+1)}. \tag{4.40}$$

Thus, the variance σ_n^2 of B_n is proportional to the spectral density $S_{B_H}(f)$.

Based on above derivation, FBM sample traces can be easily generated by making use of discrete Fourier transform through the following procedure:

1. Generate N independent random phases $\phi_1, \phi_2, \cdots, \phi_N$ from uniform distributions with range $[0, 2\pi]$.
2. Generate the amplitude $|B_n|$ from normal distributions with mean 0 and variance proportional to n^{-1-2H} (negative values are acceptable here, or alternatively take the absolute value of the normal variates).
3. Form the spectral $\{|B_n| \exp(i\phi_n)\}$.
4. Apply inverse Fourier transform to the above spectrum to obtain a complex-valued discrete-time FBM process, $B_H(n)$.
5. Finally take the real part of $B_H(n)$.

In this algorithm, it is better if the size of sample traces is 2^n in order to make use of the fast Fourier transform. Thus, its computational complexity is $O(N \log N)$.

4.1.6 Whittle's Approximate Approach

This algorithm is derived from Whittle's MLE [Ber94, Pax95]. Suppose the power spectrum of an FGN process $f(\lambda)$ is known, then a sequence of complex numbers, $Z[i]$, corresponding to the power spectrum can be constructed. $Z[i]$ is the sample trace of the FGN in the frequency domain. The inverse discrete Fourier transform (DFT) can be used to obtain $G_H[i]$. As $G_H[i]$ has the power spectrum of FGN in which the autocorrelation and the power spectrum form a Fourier pair, $G_H[i]$ is guaranteed to have the autocorrelation properties of an FGN process.

For an FGN process the difficulty with this approximate method is to obtain an accurate $f(\lambda)$ [Ber94]. The spectral density of X_t is given by

$$f(\lambda) = 2c_f(1 - cos\lambda) \sum_{j=-\infty}^{\infty} |2\pi j + \lambda|^{-2H-1}, \lambda \in [-\pi, \pi] \tag{4.41}$$

with $c_f = c_f(H, \sigma^2) = -\sigma^2 \Gamma(2H+1)(2\pi)^{-1} sin(\pi H)$ and $\sigma^2 = Var[G_H(t)]$. After rearranging this equation can be expressed as

$$f(\lambda) = A(\lambda)[|\lambda|^{-2H-1} + B(\lambda)] \tag{4.42}$$

for $0 < H < 1$ and $-\pi \leq \lambda \leq \pi$, where

$$A(\lambda) = 2\sigma^2 sin(\pi H)\Gamma(2H+1)(1 - cos\lambda), \tag{4.43}$$

$$B(\lambda) = \sum_{j=1}^{\infty} [(2\pi j + \lambda)^{-2H-1} + (2\pi j - \lambda)^{-2H-1}]. \tag{4.44}$$

Without loss of generality, let the variance of the FGN process be unity.

The main problem with using Eq. (4.42) to compute the power spectrum is the infinite summation in the expression of $B(\lambda)$, for which no closed form is known. Paxson developed a general method for approximating this summation [Pax95]. The term $B(\lambda)$ can be approximated as

$$B(\lambda) \simeq a_1^d + b_1^d + a_2^d + b_2^d + a_3^d + b_3^d + \frac{a_3^{d'} + b_3^{d'} + a_4^{d'} + b_4^{d'}}{8\pi H}, \tag{4.45}$$

where

$$\begin{aligned} a_k &= 2\pi k + \lambda, \quad b_k = 2\pi k - \lambda, \\ d &= -2H - 1, \quad d' = -2H. \end{aligned}$$

When using the approximate method there are several things that should be noted. The magnitude of $Z[i]$ is an independent exponential random variable with a mean of one, while its phase is a uniform random variable. Because the sample paths in the time domain are real numbers, the real part of power spectrum is an even function while the imaginary part is an odd function. Thus, when generating the sample trace in the frequency domain, the number of samples should be even. All that is required is to generate half of the samples as seeds, called $Z'[k]$ for $k = 1, 2, \cdots n/2$ if the whole sample path has n samples. Then, expand the sequence to the whole sample path following the rule

$$Z[i] = \begin{cases} 0, & \text{if } i = 0, \\ Z'[i], & \text{if } 0 < i \leq n/2, \text{ and} \\ \overline{Z'}[n-j], & \text{if } n/2 < i < n. \end{cases} \tag{4.46}$$

Applying the inverse DFT to $\{Z[i]\}$ the sample trace in time domain is obtained. To make use of the inverse FFT it is more convenient to let the size of samples be an integer of power of 2.

4.2 Results of Computation and Estimation

Based on the algorithms stated in the previous section which generate FBM or FGN traces, the quality of their generated self-similar samples is another important issue. The Q-Q plot is used to visually assess their goodness-of-fit and the three estimators, the R/S statistic, the VT plot, and Whittle's MLE, are applied to estimate the Hurst value of the generated traces. In addition, the computational speed is the key issue for simulation performance. The comparison of their computational speed is covered in Sec. 4.2.1.

In Sec. 4.2.2 to 4.2.7 the numerical results of the algorithms are examined using the Q-Q plot, the R/S statistic, the VT plot, and Whittle's MLE, individually. All the sample traces are generated with a selected random seed but with different algorithms and Hurst parameters.

Because all the generating algorithms are derived from the Gaussian random variables or their frequency version, the exponential distribution for magnitude and the uniform distribution for phases, the marginal distribution of the sample paths must be normal or near normal. Using the Q-Q plot to evaluate the sample paths the probability plot should fit a straight line with the normal distribution.

In each subsection the same sample traces used to illustrate the Q-Q plot are employed again to plot the pox plots of the R/S statistic, the VT plot, and Whittle's MLE with different aggregation levels. For the Q-Q plot, the dots are calculated by the quantile and the straight line is the quantile values of the associated normal distributions. For each pox plot of the R/S statistic all the dots are calculated results, the solid line is the regression of the dots, and the dashed line is the target results. Similarly, the dots in the diagrams of the VT plots are the calculated results and the solid and dashed lines are the regression and reference lines whose slopes indicate the Hurst values. For the plots using Whittle's MLE with

different aggregation levels, which ranges from 1 to 200 in steps of 10, the solid line is the results and the dashed lines along the solid line are the upper and lower 95% confidence intervals. The dots lines in the plots of Whittle's MLE with aggregation levels are targeted Hurst values.

For each algorithm four different Hurst parameters, $H = 0.5, 0.7, 0.9$, and 0.99 are chosen to examine its goodness-of-fit. Although all these three estimators can extract the Hurst parameter from FGN samples, two of them, the VT plot and the R/S statistic, are point estimators. Only Whittle's MLE incorporates an interval estimate. Therefore, a set of twenty different random seeds is chosen to examine the range of the Hurst values. Table 4.1 - 4.6 list the results of algorithms with three different estimators. For each algorithm three different Hurst parameters are chosen to generate FBM or FGN traces. The sample traces are estimated by the R/S statistic, the VT plot, and Whittle's approximate MLE. Twenty sample traces generated by the chosen Hurst parameter and algorithm are tested. This provides their mean Hurst value and 95% confidence interval. Also, the maximum and minimum Hurst values of the twenty traces are listed in each table. In addition, for each algorithm, a figure of the results is illustrated with the random seed.

The following subsections will discuss the numerical results of the algorithms of Sec. 4.1, individually. In addition, the computational effort of the algorithms is compared.

4.2.1 Computational Effort

In network simulation, especially the high speed or large scale networks, computational efforts is a very important issue. It could take a whole day to simulate several seconds of real-world network traffic. To simulate such a system, traffic generation might therefore be a problem. Hence, the complexity of the algorithms will be a very important benchmark for them. It has been shown that the complexity of the dFGN algorithm is the worst with $O(N^2)$. The complexity of other algorithms are listed as follows: the fFGN algorithm is $O(N \log N)$, the RMD algorithm is $O(N)$, the SRA algorithm is $O(N)$, the WM function is $O(N \log N)$, and Whittle's approximate approach is $O(N \log N)$. As the last two algorithms generate random values in the frequency domain, their major efforts are the Fourier transformation. Among the techniques of Fourier transform, the fast Fourier transform is the best choice because of its complexity equal to $O(N \log N)$, however, the sample size should be restrict as a power of 2.

Despite the complexity of the algorithms having been fully revealed, the different approaches will cause slightly different computational efforts. Their mean computational times are depicted in Fig. 4.3. The computational time is measured by generating sample paths with $2^{15} = 32768$ sample points on a SUN SPARC-5 clone. Every algorithm has been run more than 80 times to obtain a mean computational time. For the dFGN algorithm, the truncated parameters M and n are set to be 700 and 10, respectively. In other words, every time interval is subdivided into ten sub-intervals and every sub-interval is a Gaussian random variable with zero mean and variance $1/n$. The current sample point is dependent on 700 previous samples. For the fFGN algorithm, the base B is equal to 2 and the quality factor Q is 20.

From Fig. 4.3, it is obvious that the dFGN algorithm is the worst case. The RMD which has the best performance is five hundred times faster than the dFGN algorithm. The SRA algorithm is a little bit slower than the RMD algorithm although they have the same complexity order. All sample points on

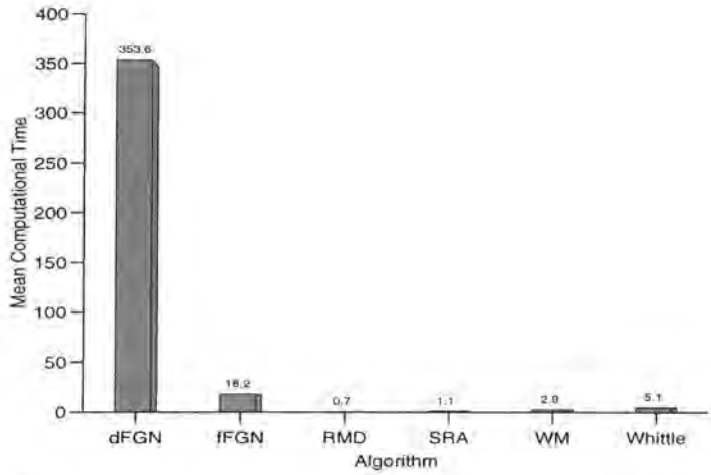


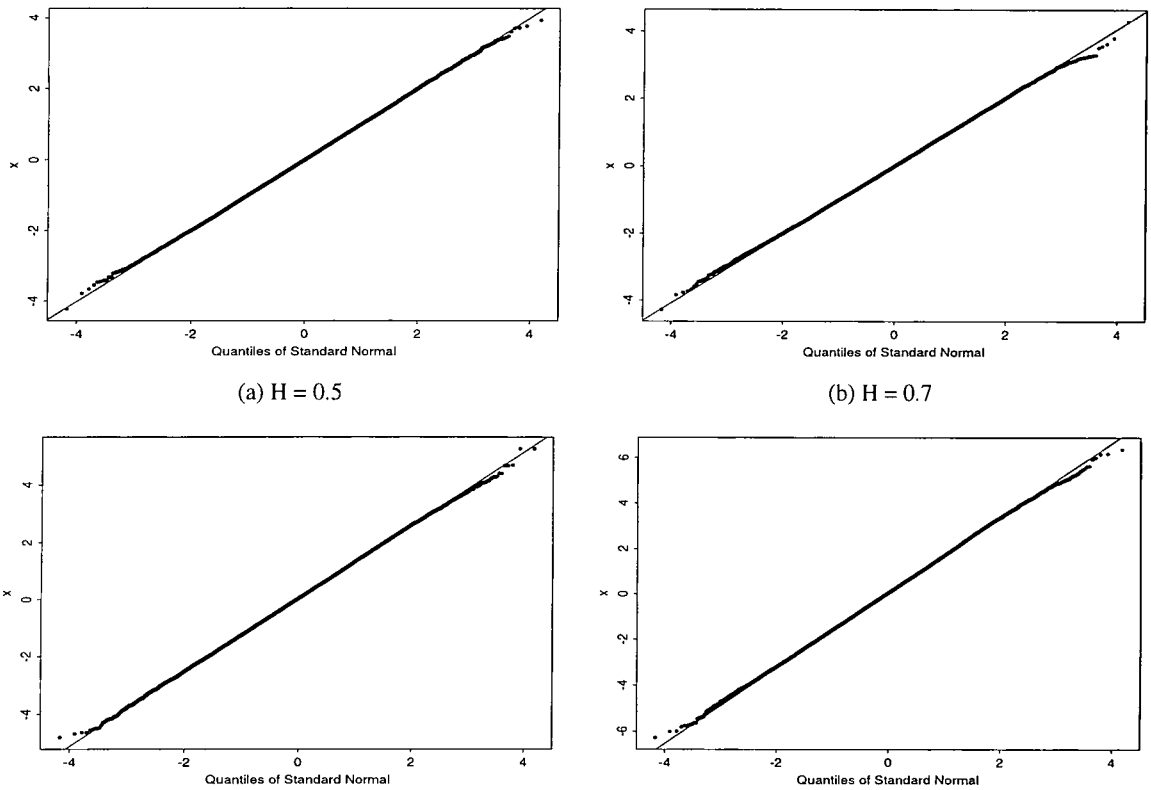
Figure 4.3: Mean computational time of algorithms to generate 32768 sample points.

stage n in the SRA algorithm have to add displacements while the sample points in the RMD algorithm just add one displacement to newly generate samples. The third best result is the WM function in which the major computational effort is to apply the fast Fourier transform. Although Whittle's approximate approach also uses the fast Fourier transform to generate sample traces, it needs two random distributions to generate the spectrum, an exponential distribution for magnitude and a uniform distribution for phase. The fFGN algorithm is marginally better than the dFGN algorithm, although its complexity order is the same as the fast Fourier transform.

In addition, a very important issue should be mentioned here. In network simulation normally people are unaware how many events will occur during the simulation period before running it. Therefore, it is more convenient to generate sample points on-the-fly, i.e. when the simulation system requires an event, it requests an event from the traffic generator. However, most of the algorithms need to decide the size of sample paths in advance. One exception is the the dFGN algorithm. The RMD and SRA algorithms bisect the time interval, recursively. The size of sample points must be power of 2 to determine how many times bisection occurs. For the WM function and Whittle's MLE approach the sample points are generated in the frequency domain, before making use of the Fourier transform. Hence, they are restricted by the Fourier transform. Furthermore, the best performance is obtained by using the fast Fourier transform which constrains the size to an integer of a power of 2. Even using the discrete Fourier transform the size also needs to be determined in advance. The fFGN algorithm requires a number $N(T)$ of independent Markov-Gauss processes, where $N(T) = \lceil \log(QT)/\log B \rceil$ is dependent on the size of sample paths, T. An alternative way is to cascade several generating sub-traces. However, it will violate the basic concept of LRD because any two sub-traces are independent. Thus, only the dFGN algorithm can generate the sample points on-the-fly.

4.2.2 The dFGN Algorithm

For the dFGN algorithm two more parameters, the cover range M and subintervals n, must be set as well as the Hurst parameter. Here, the cover range is set to 700, while the subintervals are 10. In Fig. 4.4 (a)-(d) the Q-Q plots show that the samples of the dFGN algorithm fit the straight line very well. Therefore,



(a) H = 0.5 (b) H = 0.7

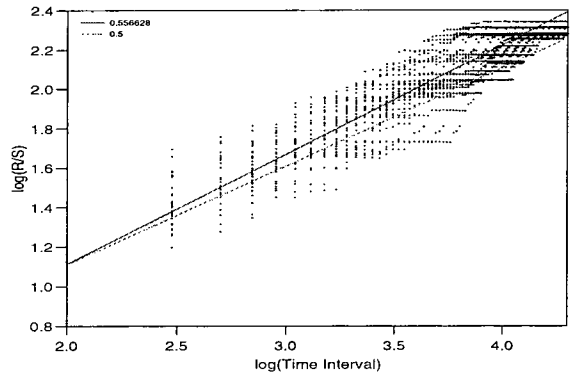
(c) H = 0.9 (d) H = 0.99

Figure 4.4: The Q-Q plots of the sample traces generated by the dFGN algorithm.

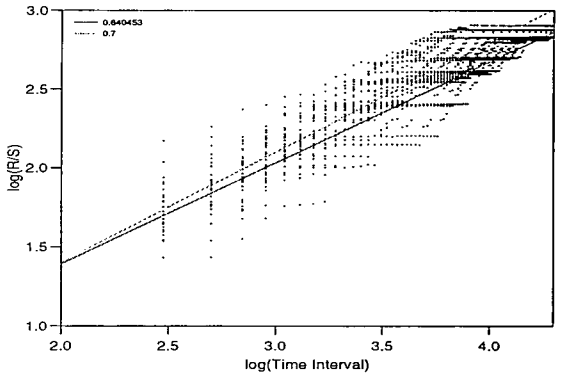
the samples of the dFGN algorithm shows a goodness-of-fit to the normal distribution, with no significant difference when the Hurst value is changed. For the pox plots of the R/S statistic in Fig. 4.5, the values are all lower than the expected Hurst value except for the short-range dependent case, $H = 0.5$. The solid line in each diagram indicates the Hurst value with its slope, whereas the dashed line is the slope of the target Hurst parameter. These four diagrams shows that the higher the hurst parameter, the worse the results. The VT plot of Fig. 4.6 exhibits very poor results when the Hurst parameter is set to $H = 0.9$ and 0.99. Similar to the lines drawn in the pox plots of the R/S statistic the solid line refers to the value while the dotted line exhibits the reference value.

However, the results using Whittle's MLE with aggregation levels in Fig. 4.7 (a) - (d) leads to a very different conclusion. For all four sub-diagrams the results (solid lines) fit the target Hurst values (dotted lines) very well. The two dashed lines either side of the results are the upper and lower bounds of the 95% confidence interval. As expected the larger the aggregation level, the wider the confidence interval. This effect is caused by the size of every subsample trace. When a sample trace is divided into more sections, then they become aggregated. Therefore, the Hurst values are extracted from an increasing smaller size of samples. When comparing these four diagrams the aggregation level seems to be less affected by the results for higher parameters. In the short-range dependent case, $H = 0.5$, the Hurst value gradually parts from the reference line, which is the target Hurst value. When $H = 0.9$ the line of result perfectly matches the reference line, even when the aggregation level increases.

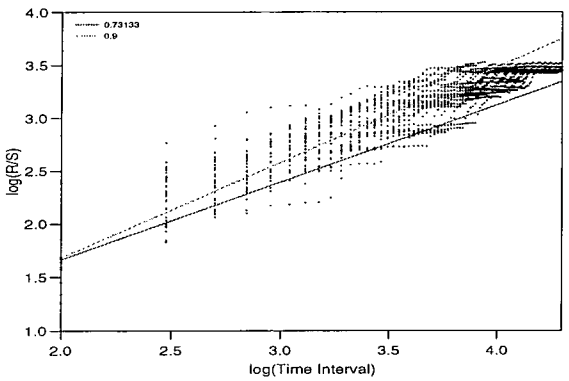
When investigating the change of Hurst parameter under the selected random seed, Fig. 4.8 shows that the results using Whittle's MLE fit the target relatively accurately, where the Hurst parameter varies



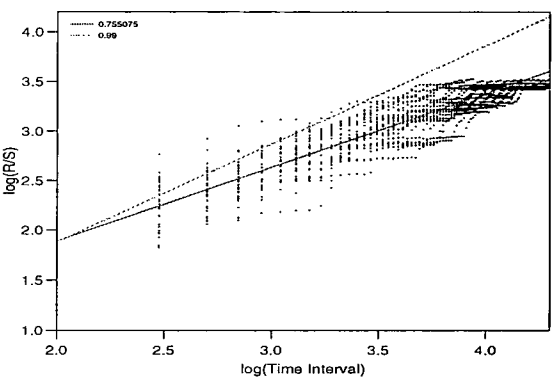
(a) H = 0.5



(b) H = 0.7

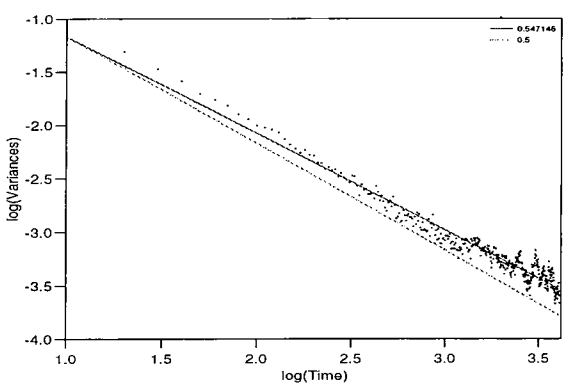


(c) H = 0.9

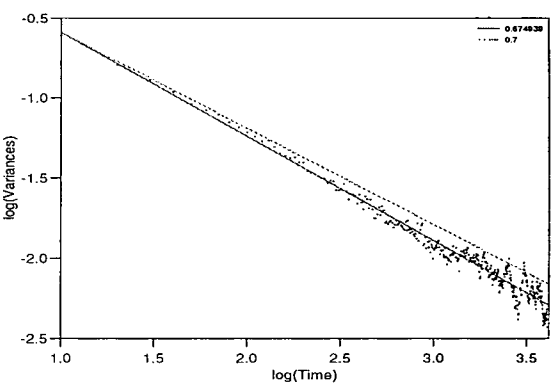


(d) H = 0.99

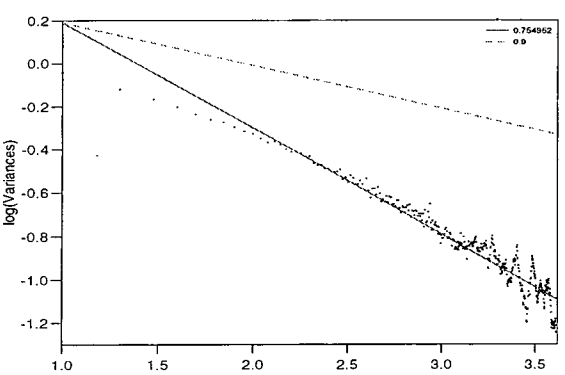
Figure 4.5: The pox plots of the R/S statistic of the sample traces generated by the dFGN algorithm.



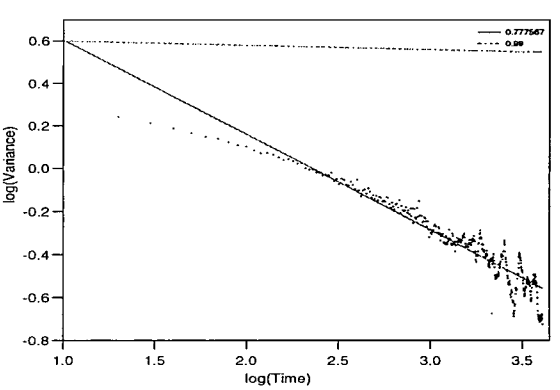
(a) H = 0.5



(b) H = 0.7



(c) H = 0.9



(d) H = 0.99

Figure 4.6: The VT plots of the sample traces generated by the dFGN algorithm.

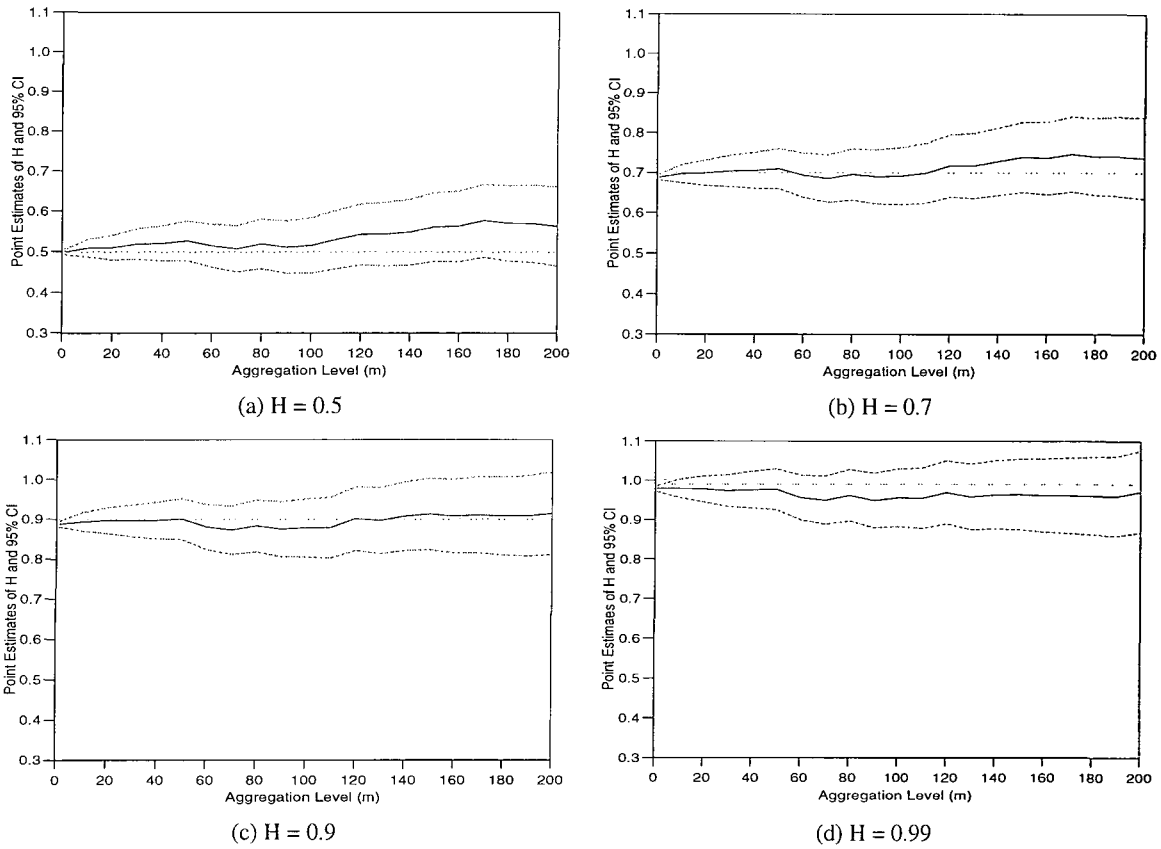


Figure 4.7: The Hurst values using Whittle's MLE of the sample traces generated by the dFGN algorithm.

from 0.5 to 0.99. In this diagram the two dashed lines along the 'Whittle' line are the upper and lower bounds of the 95% confidence interval. The results using Whittle's MLE always are slightly smaller than the target Hurst values except for the values near $H = 0.5$. The results using the R/S statistic and VT plot with varying Hurst parameter are higher than the target value when the Hurst value is below $H = 0.65$. When the Hurst parameter increases, the results using the R/S statistic and the VT plot are significantly worse than the target values. When the Hurst parameter increases, the results using the VT plot increase more slowly than they ought to do. Certainly this is just one selected case, therefore it does not mean that every different random seed produces the same results. However, it is a good representation with which to understand the results of the three estimators.

In Table 4.1 the analytical results using R/S statistic show the Hurst values which tend to approach the middle value, that is, the Hurst values are slightly larger than the lower target Hurst value of 0.5 and less than the higher Hurst value of 0.9. The analytical results using the VT plot indicate that all the mean Hurst values are always lower than the target values. The higher the target Hurst value, the bigger the difference between the estimated and target values. At the minimum VT plot the values of a number of the traces are incredibly low. Significantly better results are shown when Whittle's MLE is applied for testing. Although slightly lower mean values are highlighted in the table when the Hurst parameter increases, the spanning range is very small.

From Table 4.1 it is apparent that the results using Whittle's approximate MLE of sample traces with different random seeds are very close, while the results of the other two estimators have wider range. Thus, the 'Whittle' line of the other diagrams which generate by traces with different random seeds are

Hurst Parameter		Mean	Maximum	Minimum	95% CI
R/S Statistic	0.5	0.5229874	0.571484	0.473914	0.04725129
	0.7	0.6963354	0.762712	0.640453	0.07079152
	0.9	0.8145762	0.920307	0.731274	0.1004692
VT Plot	0.5	0.4636087	0.577534	0.273516	0.1519283
	0.7	0.581953	0.689953	0.374229	0.1563825
	0.9	0.6556407	0.757554	0.447867	0.1537915
Whittle's MLE	0.5	0.5010981	0.5071886	0.4935096	0.007292475
	0.7	0.6902205	0.6973568	0.681748	0.007543267
	0.9	0.8902595	0.8979258	0.8809194	0.007938308

Table 4.1: Analytical results of the dFGN algorithm with $M = 700$ and $n = 10$.

very close to the 'Whittle' line in Fig. 4.8. However, this situation is not suitable for the R/S statistic and the VT plot. In addition, the results of these twenty random seeds with the three estimators are depicted in Fig. 4.9. The meaning of the curves is shown on the legend beside the diagram. It is apparent that the results using Whittle's MLE form an almost straight line, while the results of the other two estimators fluctuate more, especially, the results using the VT plot. Some of the results are much lower than the target values.

4.2.3 The fFGN Algorithm

In the simulations of the fFGN algorithm two additional parameters, the base and the quality factor, are chosen above the necessary minimum values to reduce the approximate errors to an acceptable level. Here, the base is set to $B = 2$ and the quality factor is $Q = 20$. The diagrams of Fig. 4.10 are the graphical results of estimators and goodness-of-fit test. Examining the Q-Q plot of the fFGN algorithm visually, all diagrams look very impressive, but when the Hurst value increases the results become less accurate. The results using the R/S statistic in Fig. 4.11 and the VT plot in Fig. 4.12 are always less than the target Hurst values. The results using Whittle's approximate MLE are shown in Fig. 4.13. Unlike the results in Fig. 4.7 the results with different aggregation level change quite significantly. For $H = 0.9$ and 0.99 even the bounds of the 95% confidence intervals do not cover the target Hurst values.

When examining the values of the selected random seed, the R/S statistic and the VT plot exhibit nonlinear phenomena shown in Fig. 4.14 with different Hurst parameter ranging from 0.5 to 0.99. In the VT plot of Fig. 4.14, the results initially decline, before increasing very quickly, and finally showing a decrease in the speed of growth. The results using the VT plot become concave with the minimum value near $H = 0.53$. The curve of the R/S statistic begins with a very low result, then quickly approaches the target Hurst value, but never goes beyond the target value. Thereafter, the speed of growth decreases. In this case all the results of the three estimators are always lower than the target values. In comparison the results using the Whittle's MLE are the best.

Table 4.2 lists the mean results and the maximum and minimum values of the results with the set of random seeds. For the R/S statistic the mean values are higher than the target values when the Hurst

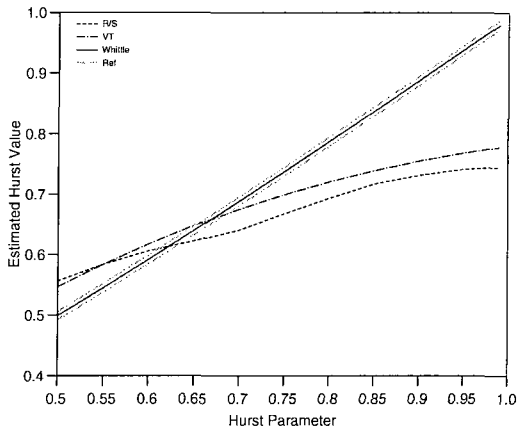


Figure 4.8: The estimated results of the dFGN algorithm with different Hurst parameters.

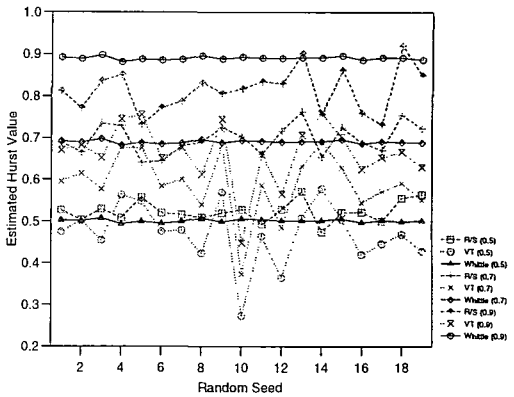
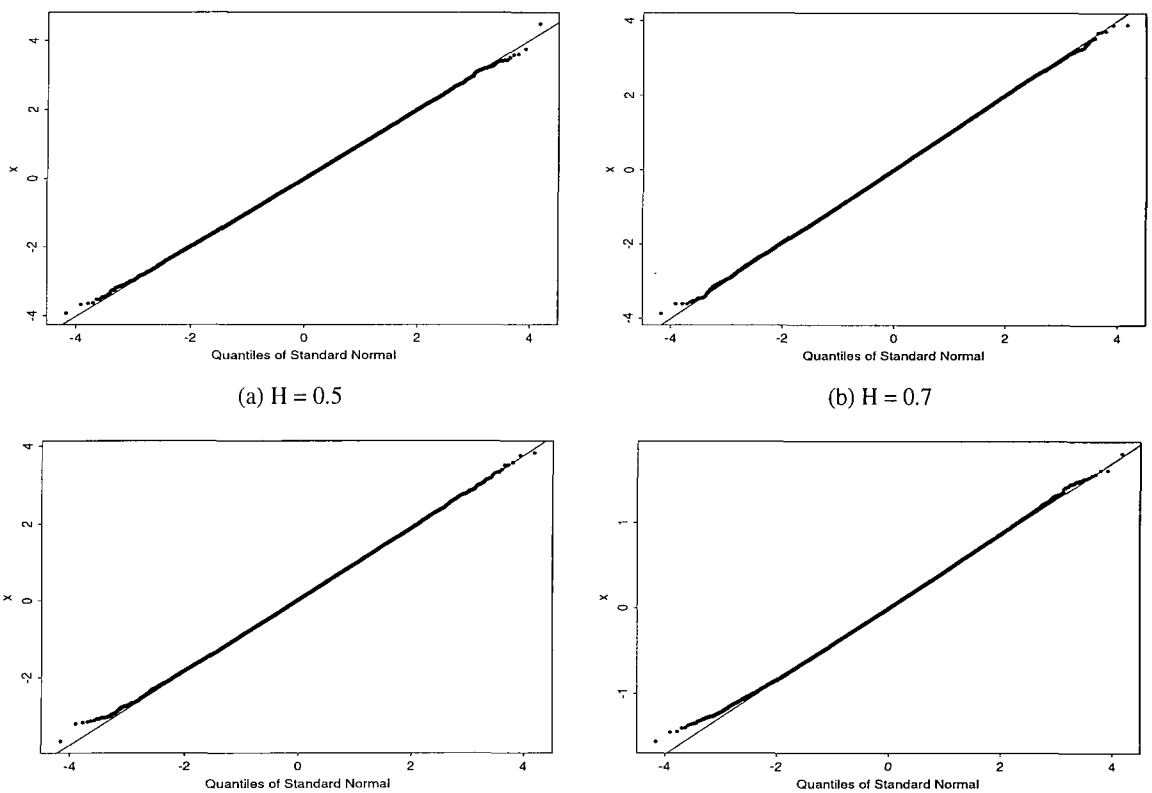


Figure 4.9: The estimated results of the dFGN algorithm with different random seeds.



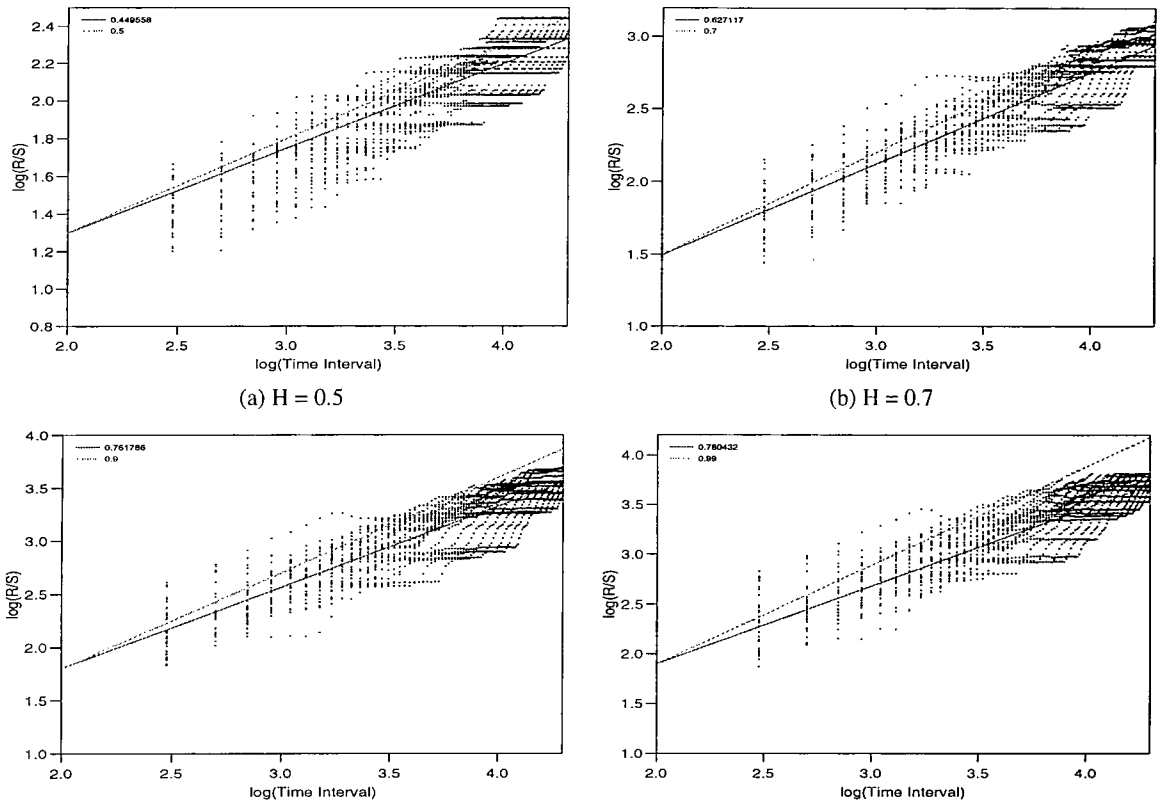
(a) H = 0.5

(b) H = 0.7

(c) H = 0.9

(d) H = 0.99

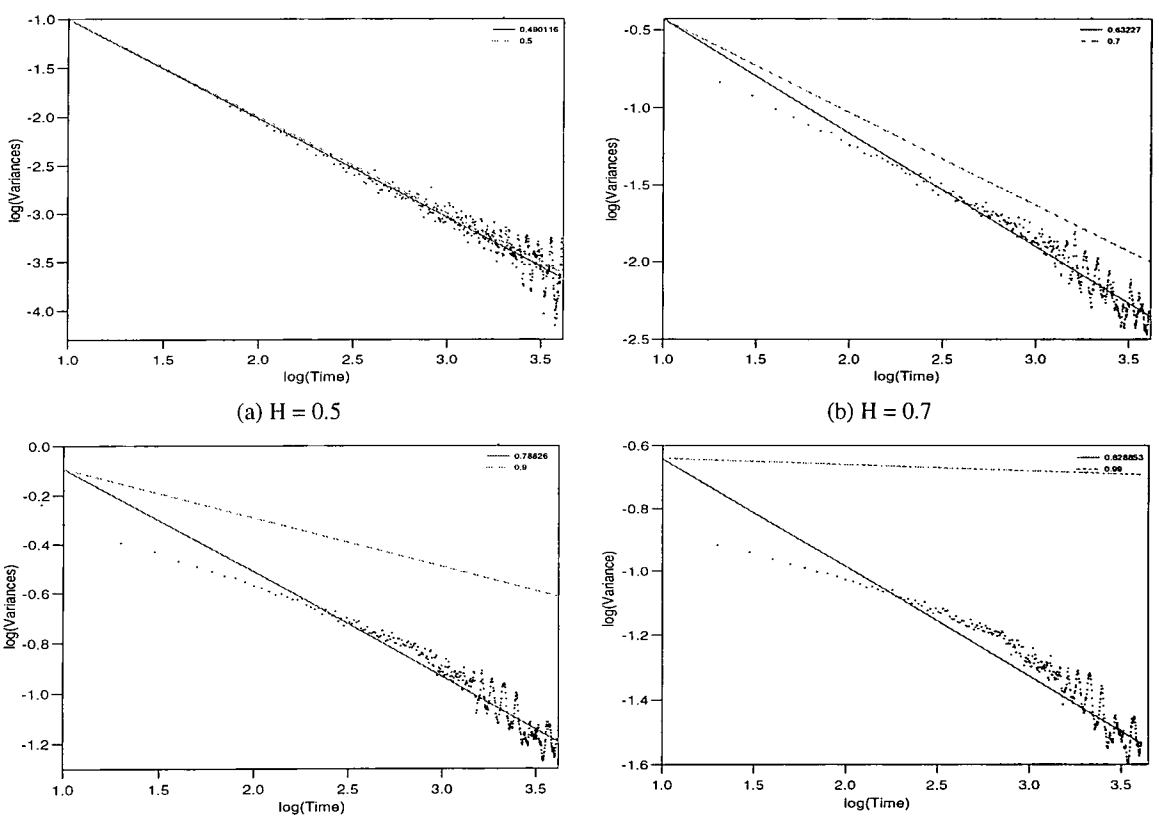
Figure 4.10: The Q-Q plots of the sample traces generated by the fFGN algorithm.



(a) H = 0.5 (b) H = 0.7

(c) H = 0.9 (d) H = 0.99

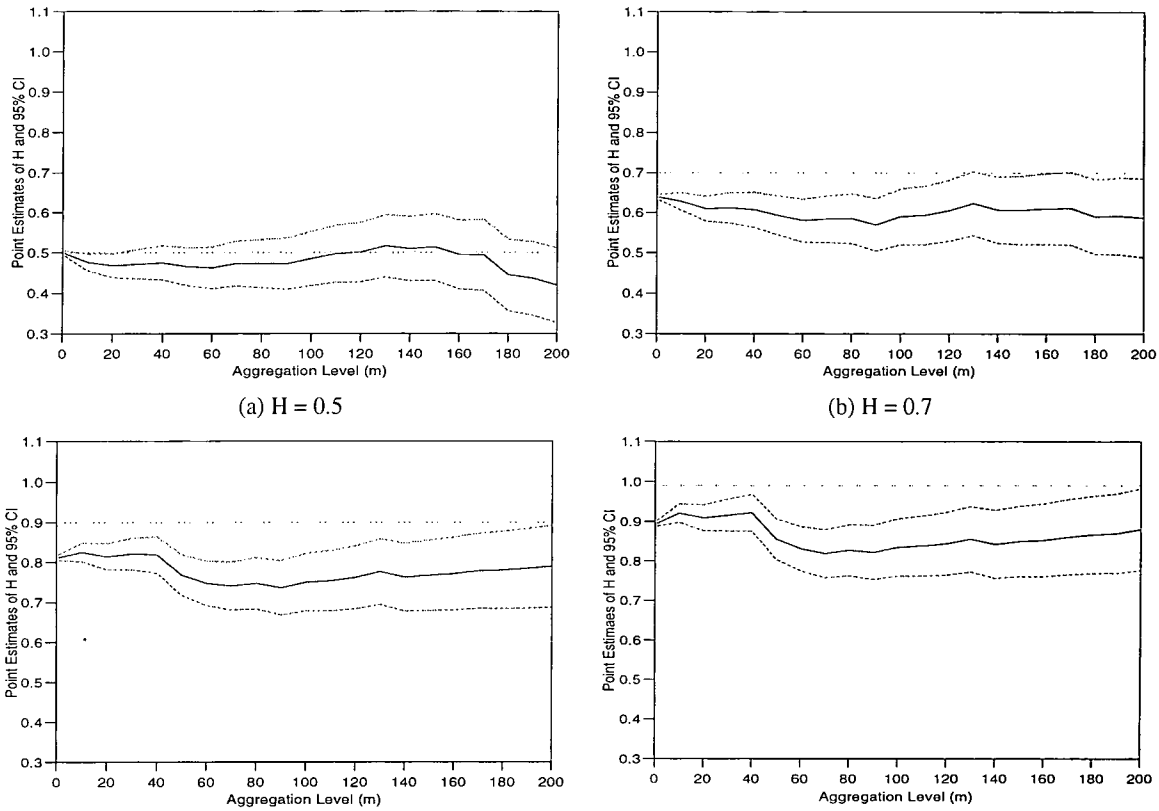
Figure 4.11: The pox plots of the R/S statistic of the sample traces generated by the fFGN algorithm.



(a) H = 0.5 (b) H = 0.7

(c) H = 0.9 (d) H = 0.99

Figure 4.12: The VT plots of the sample traces generated by the fFGN algorithm.



(a) H = 0.5 (b) H = 0.7

(c) H = 0.9 (d) H = 0.99

Figure 4.13: The Hurst values using Whittle's MLE of the sample traces generated by the fFGN algorithms.

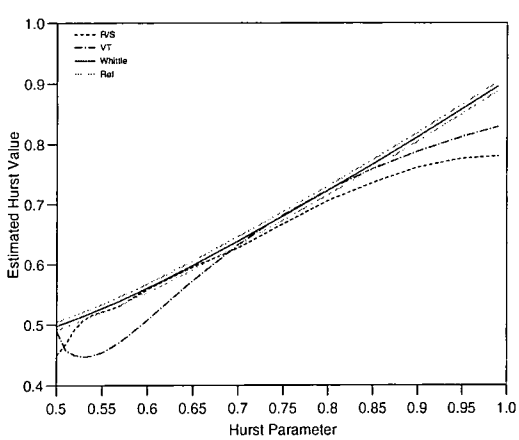


Figure 4.14: The estimated values of the fFGN algorithm with different Hurst parameters.

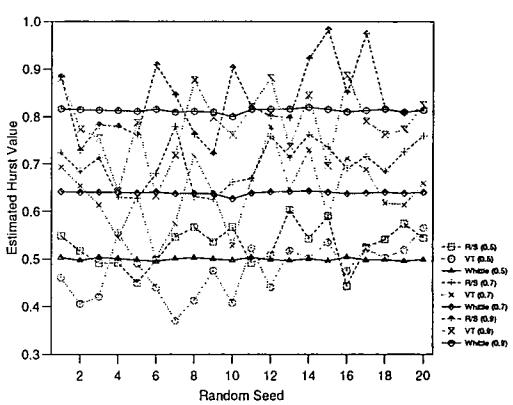


Figure 4.15: The estimated results of the fFGN algorithm with different random seeds.

Hurst Parameter		Mean	Maximum	Minimum	95% CI
R/S Statistic	0.5	0.5284779	0.603229	0.442435	0.08410934
	0.7	0.6986941	0.779765	0.625321	0.09483915
	0.9	0.8337986	0.982923	0.721598	0.1467674
VT Plot	0.5	0.4767081	0.564843	0.370747	0.1077152
	0.7	0.6399548	0.779318	0.501531	0.138547
	0.9	0.7824212	0.887265	0.632559	0.1453562
Whittle's MLE	0.5	0.4992075	0.5042979	0.4951112	0.005738461
	0.7	0.6388327	0.6431055	0.626492	0.006628082
	0.9	0.8135772	0.8204282	0.8008241	0.008007577

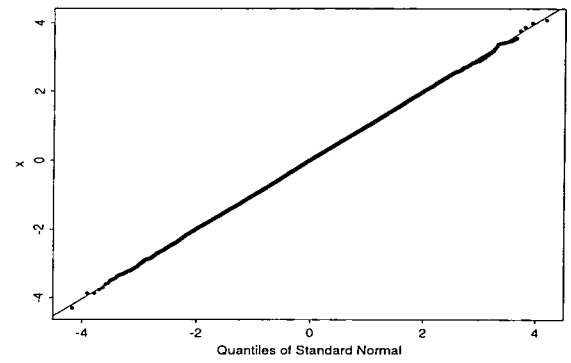
Table 4.2: Analytical results of the fFGN algorithm.

parameter is set to 0.5. For $H = 0.7$ the mean result using the R/S statistic is almost the same as the target value. The mean result with $H = 0.9$ is smaller than the target value. The mean result using the VT plot with different Hurst parameters is always lower than the target value. When examining the mean results using Whittle's MLE, they are all lower than the target values. However, the mean value is very close to the target value when $H = 0.5$. The results illustrated in Fig. 4.15 are similar to those of the dFGN algorithm. The varying range of the R/S statistic and the VT plot are very wide. The values using Whittle's MLE almost form straight lines with three different Hurst values.

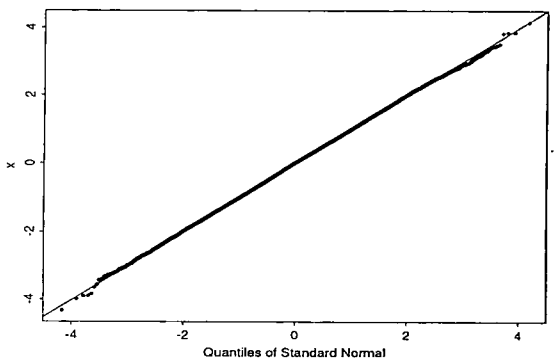
4.2.4 The RMD Algorithm

Similar to the previous two algorithms the third rows of Fig. 4.16 - 4.19 depict the graphical results of the Q-Q plot, the R/S statistic, the VT plot, and Whittle's MLE with aggregation levels, respectively. When $H = 0.99$ the Q-Q plot is not as good as those of $H = 0.5, 0.7$, and 0.9. Therefore, the marginal distribution of the sample path is not so close to a normal distribution as those of $H = 0.5, 0.7$, and 0.9. The results using the R/S statistic in Fig. 4.17 are an improvement when the Hurst parameter increases. However, when the Hurst parameter is set to 0.99, the estimation does not increase in line with the Hurst parameter. For the VT plot all four diagrams exhibit a better result, although the values are higher than the expected Hurst values. When $H = 0.90$ the result is almost the same as the target value. The graphical results using Whittle's MLE with aggregation levels reveal that for the case $H = 0.7$ the expected Hurst value is better than those of $H = 0.5, 0.9$, and 0.99. For $H = 0.5$ the result is higher than the expected value, while for $H = 0.9$ and 0.99, the results are lower than the target values.

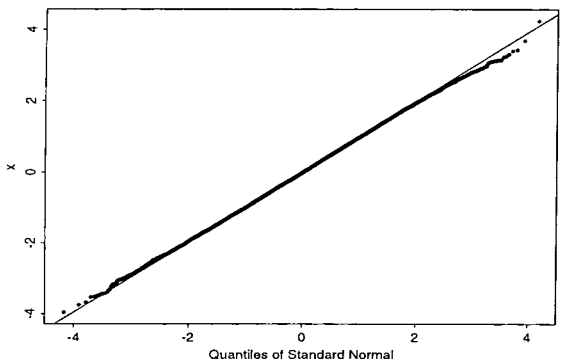
Table 4.3 shows the results of the fastest generating algorithm. When $H = 0.5$, the mean results of the three estimators are very accurate. Comparing their differences between the maximum and the minimum values, the R/S statistic exhibits a wider range, almost reaching 0.2. In the case when $H = 0.7$, the mean result using the R/S statistic shows a better result, as the other two estimators have smaller values. However, the difference between the maximum and minimum values for the R/S statistic and reaches 0.28. When examining the case of $H = 0.9$, where the difference between the maximum and minimum values is more than 0.3, the ranges of the VT plot are very large as well. The difference between the



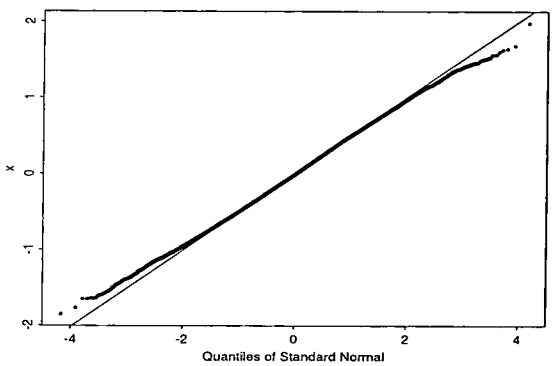
(a) H = 0.5



(b) H = 0.7

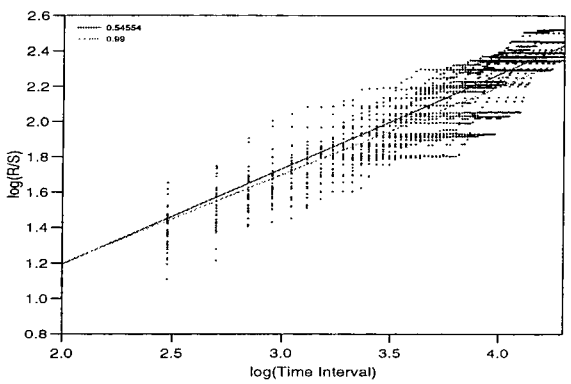


(c) H = 0.9

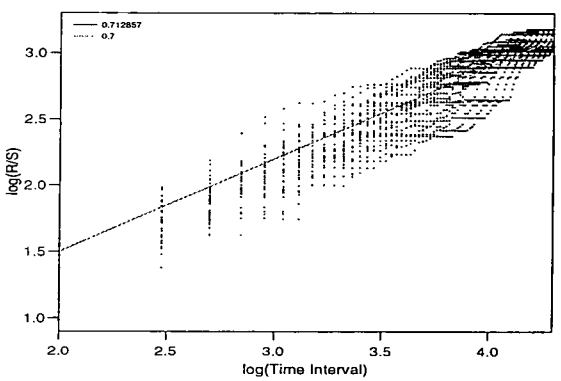


(d) H = 0.99

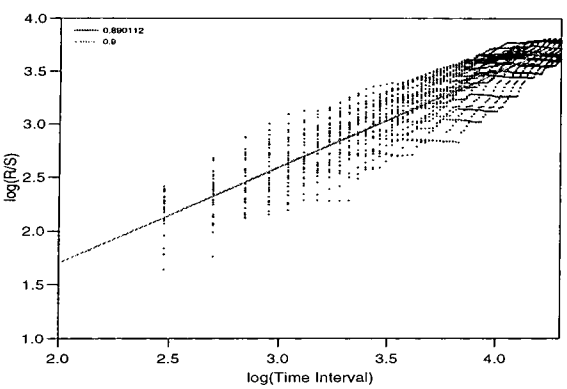
Figure 4.16: The Q-Q plots of the sample traces generated by the RMD algorithm.



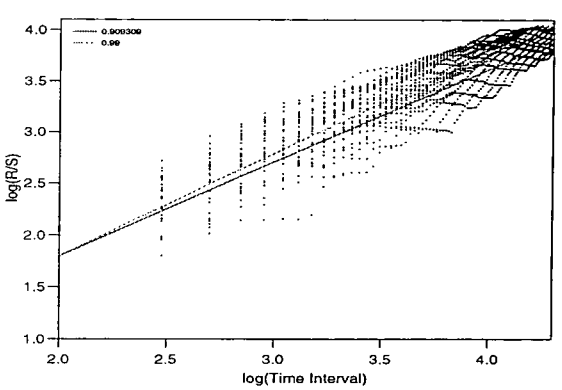
(a) H = 0.5



(b) H = 0.7

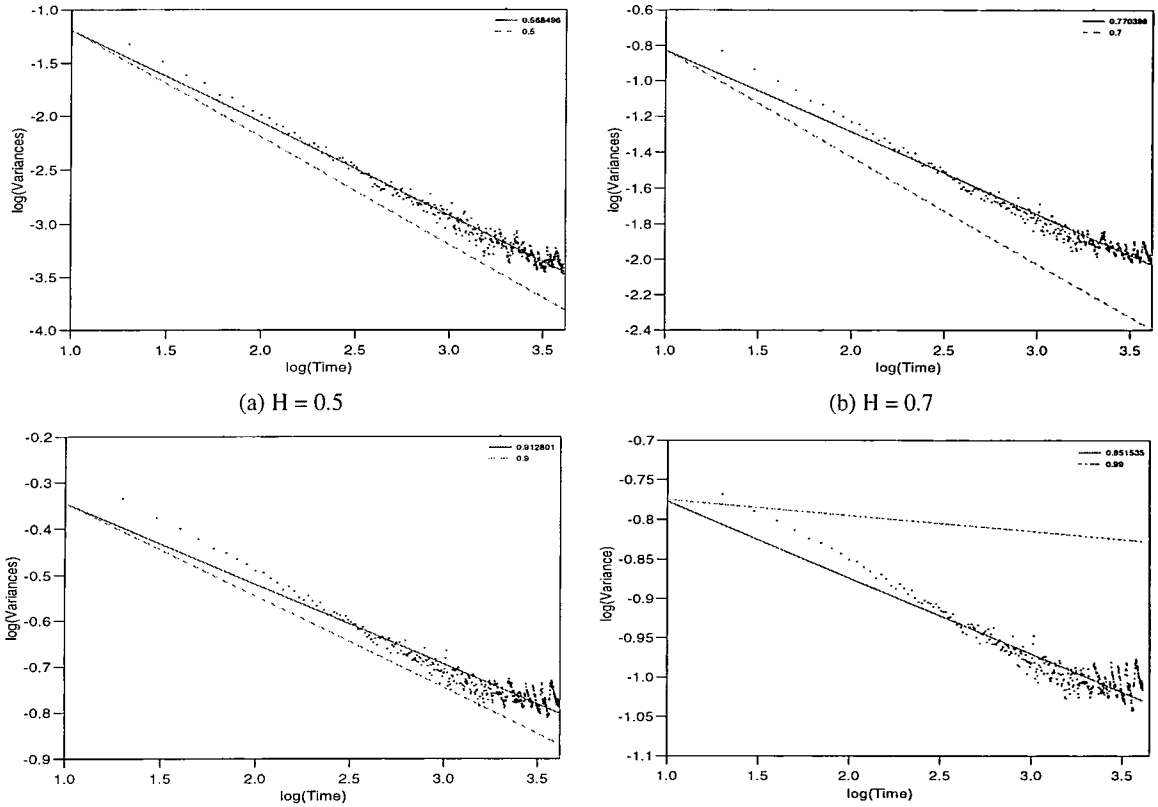


(c) H = 0.9



(d) H = 0.99

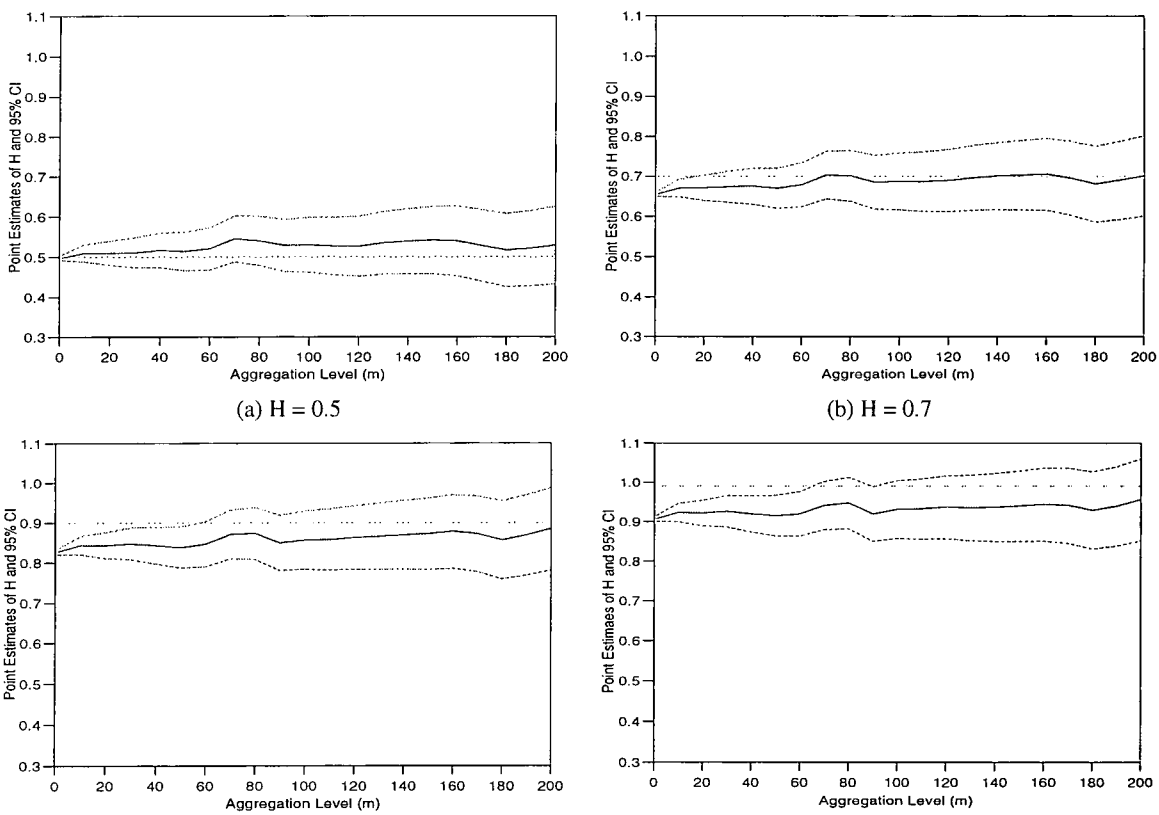
Figure 4.17: The pox plots of the R/S statistic of the sample traces generated by the RMD algorithm.



(a) H = 0.5 (b) H = 0.7

(c) H = 0.9 (d) H = 0.99

Figure 4.18: The VT plots of the sample traces generated by the RMD algorithm.



(a) H = 0.5 (b) H = 0.7

(c) H = 0.9 (d) H = 0.99

Figure 4.19: The Hurst values using Whittle's MLE of the sample traces generated by the RMD algorithm.

Hurst Parameter		Mean	Maximum	Minimum	95% CI
R/S Statistic	0.5	0.5060301	0.613618	0.411083	0.09530388
	0.7	0.6750333	0.831404	0.547873	0.1329722
	0.9	0.8081523	0.962106	0.645051	0.1568099
VT Plot	0.5	0.5000288	0.596461	0.418949	0.11478450
	0.7	0.6664359	0.770398	0.577834	0.11946620
	0.9	0.7983201	0.912801	0.716241	0.111282
Whittle's MLE	0.5	0.5003984	0.5070395	0.4943967	0.006301853
	0.7	0.6568086	0.6644614	0.6503493	0.0068355
	0.9	0.8260991	0.8354866	0.8184894	0.007784678

Table 4.3: Analytical results of the RMD algorithm.

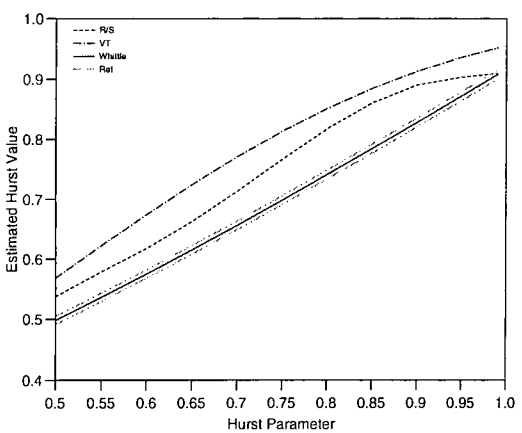


Figure 4.20: The estimated results of the RMD algorithm with different Hurst parameters.

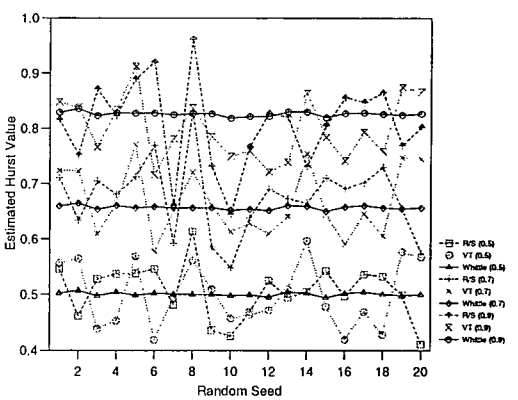
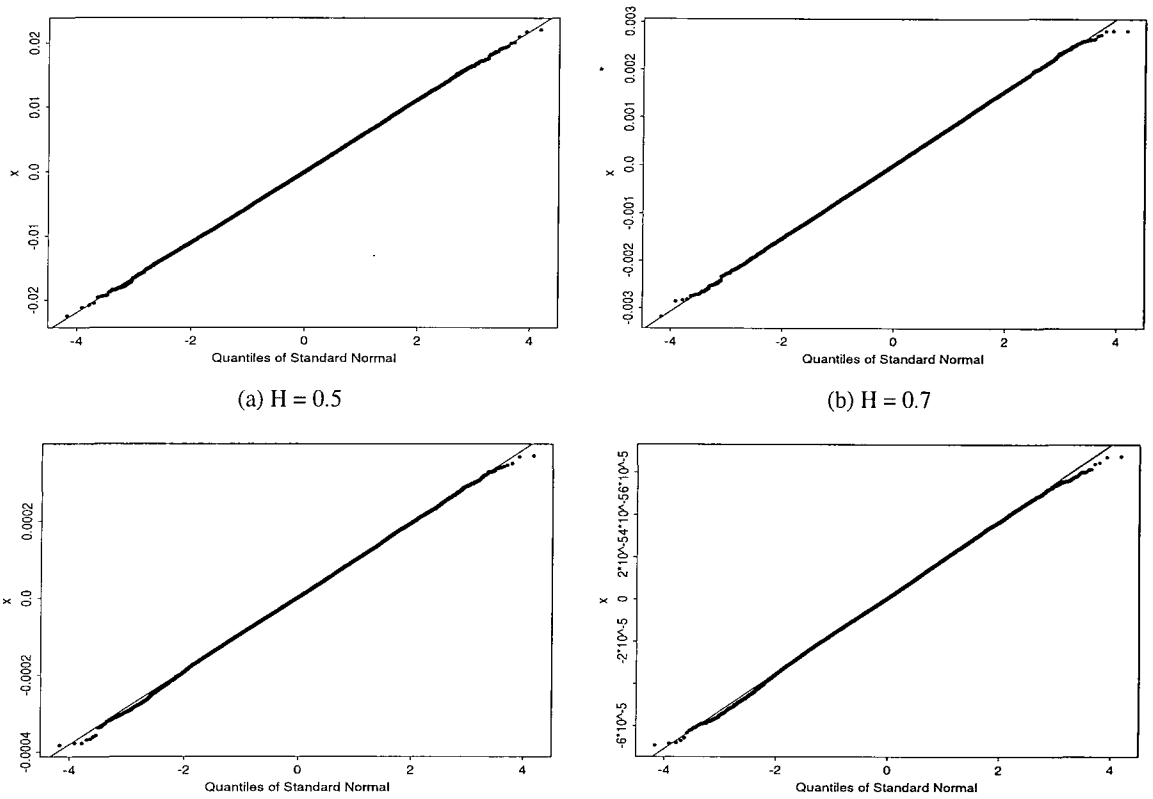


Figure 4.21: The estimated results of the RMD algorithm with different random seeds.

maximum and minimum results of $H = 0.5, 0.7$ and 0.9 are $0.18, 0.19$, and 0.19, respectively. The results span nearly $1/3$, $1/4$ and $1/5$ of the target values.

When examining the results of the chosen case, the 'R/S' line approaches the ideal line significantly better than the other two estimators in Fig. 4.20. In this case the results using the VT plot are always larger than the target values, while the results using Whittle's MLE are almost always lower than the ideal values. When the Hurst parameter is near 0.5, the results using Whittle's MLE fit the target value very well. However, the gap between them gradually increases while the Hurst parameter increases. The results using the VT plot are poor when the Hurst parameter is less than 0.9, with the result being significantly worse than the other two when the Hurst parameter is between 0.5 and 0.75. The result using Whittle's MLE is the worst case when the Hurst value is larger than 0.75. This result conflicts somewhat with the results in Table. 4.3. When comparing with Fig. 4.21 the results using the VT plot with the selected random seed are the biggest values among the results with $H = 0.7$ and 0.9. Thus, the results in Fig. 4.18 (b) and (c) mislead the visual conclusion. When $H = 0.9$ the result using the R/S statistic in Fig. 4.17 (c) exhibits a similar situation because the result is the third highest Hurst value among the twenty random seed. Thus, the selected random seed has misled the result in Fig. 4.20.



(a) H = 0.5

(b) H = 0.7

(c) H = 0.9

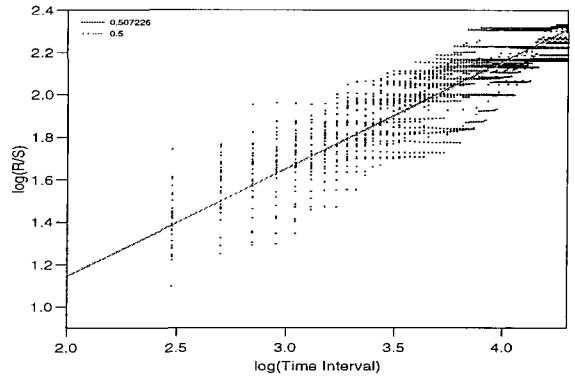
(d) H = 0.99

Figure 4.22: The Q-Q plots of the sample traces generated by the SRA algorithm.

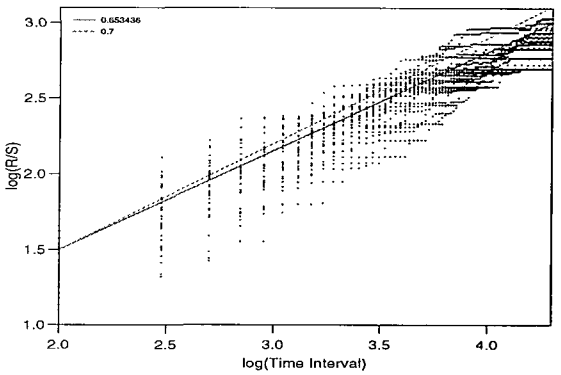
4.2.5 The SRA Algorithm

In Fig. 4.22 the diagrams are the graphical results of the goodness-of-fit test. The Q-Q plots of the SRA algorithm exhibit a good match between the scattering dots and the straight lines. There are no significant difference between these four diagrams. The R/S statistic in Fig. 4.23 (a) exhibits a very good result so that the result and reference almost form a line. When the Hurst parameter increases, the values using the R/S statistic are become worse. For the short-range dependent case, $H = 0.5$, shown in Fig. 4.24, the result using the VT plot is very close to the target value. However, the results gradually part from the target values when the Hurst parameter increases. The graphical results using Whittle's MLE with aggregation levels are shown in Fig. 4.25. As with the results using the R/S statistic and the VT plot, the result with $H = 0.5$ is the best. In particular, the results of the aggregation levels are larger than 100 and they almost overlap the reference line. However, when the Hurst parameter increases, the curve exhibits a greater fluctuation.

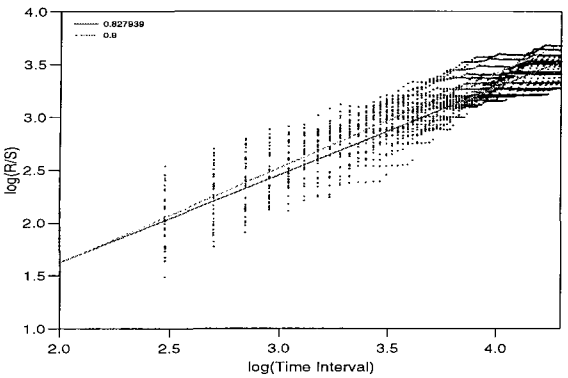
The mean results of the improved version of the RMD algorithm with twenty different random seeds are listed in Table 4.4. For $H = 0.5$ the mean result using Whittle's MLE fits the ideal value very well while the result using the R/S statistic is larger and the VT plot is smaller than 0.5. Both of the mean results using the VT plot and Whittle's MLE are significantly less than the ideal Hurst value as $H = 0.7$, but the mean result using the R/S statistic is almost the same as the target value. When the Hurst parameter is set to 0.9, the mean result using the VT plot and the Whittle's MLE have the largest and smallest gaps from the ideal value, respectively. However, on examining the difference between the



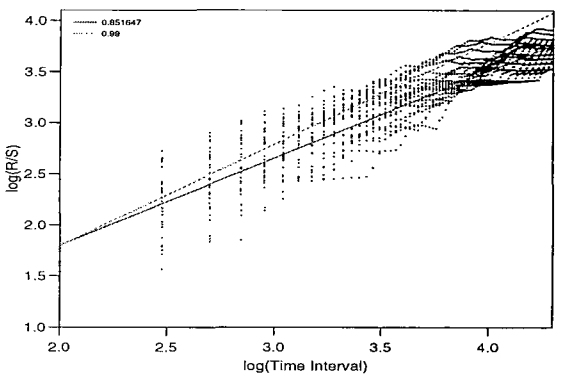
(a) H = 0.5



(b) H = 0.7

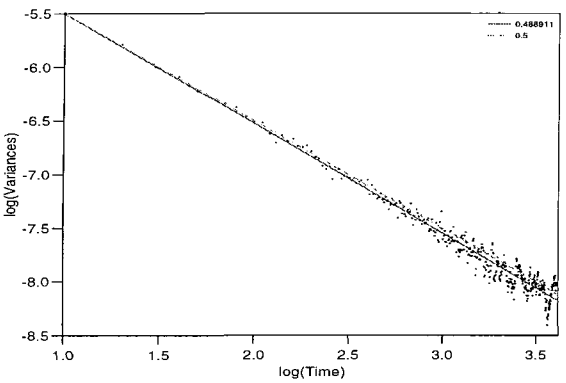


(c) H = 0.9

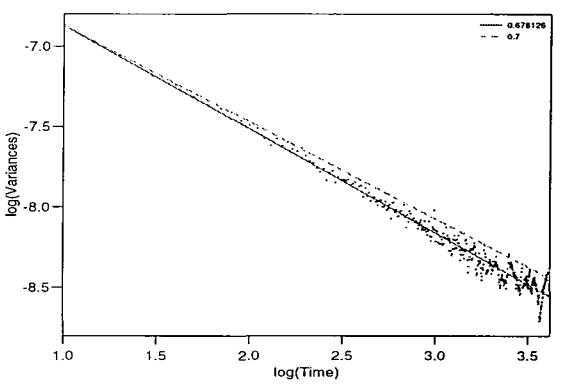


(d) H = 0.99

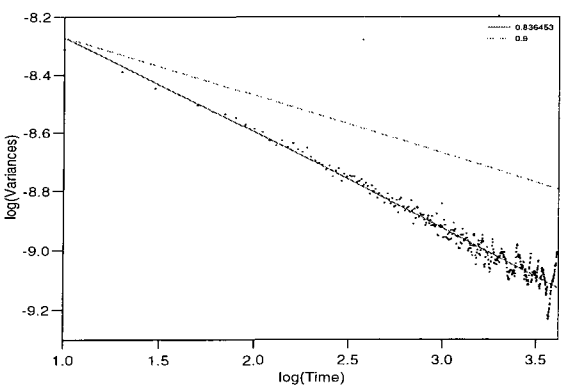
Figure 4.23: The pox plots of the R/S statistic of the sample traces generated by the SRA algorithm.



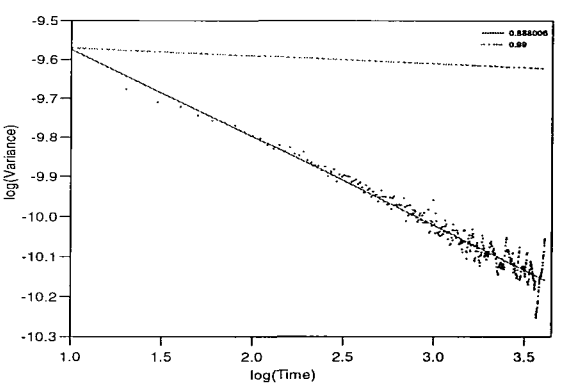
(a) H = 0.5



(b) H = 0.7

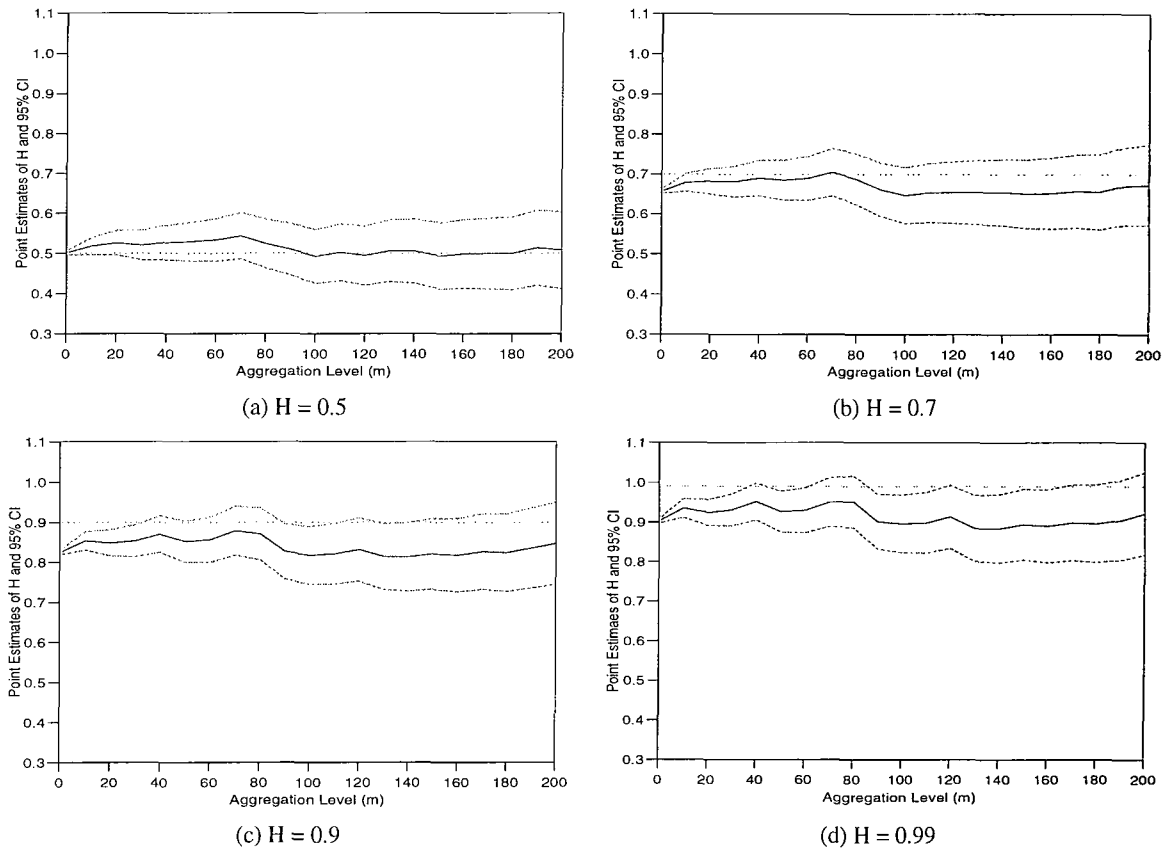


(c) H = 0.9



(d) H = 0.99

Figure 4.24: The VT plots of the sample traces generated by the SRA algorithm.



(a) H = 0.5 (b) H = 0.7

(c) H = 0.9 (d) H = 0.99

Figure 4.25: The Hurst values using Whittle's MLE of the sample traces generated by the SRA algorithm.

Hurst Parameter		Mean	Maximum	Minimum	95% CI
R/S Statistic	0.5	0.5248448	0.6055	0.40269	0.09540525
	0.7	0.6906526	0.800206	0.571149	0.1103252
	0.9	0.8177981	0.924826	0.709179	0.1079923
VT Plot	0.5	0.4877391	0.6131	0.385542	0.122876
	0.7	0.6525558	0.784003	0.529708	0.1305931
	0.9	0.7826029	0.901727	0.658293	0.1256757
Whittle's MLE	0.5	0.5004327	0.5063703	0.4942849	0.005954368
	0.7	0.656582	0.6637814	0.6505239	0.007289917
	0.9	0.8255724	0.8335734	0.8181622	0.009336806

Table 4.4: Analytical results of the SRA algorithm.

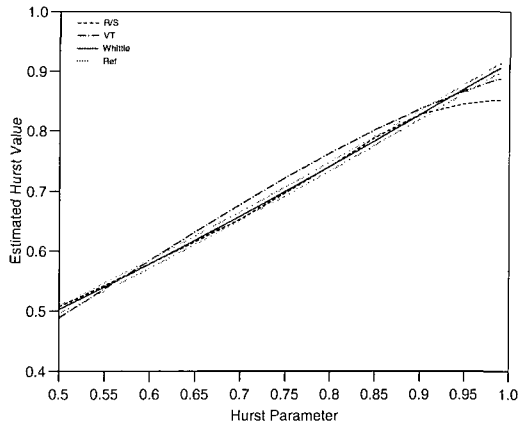


Figure 4.26: The estimated results of the SRA algorithm with different Hurst parameters.

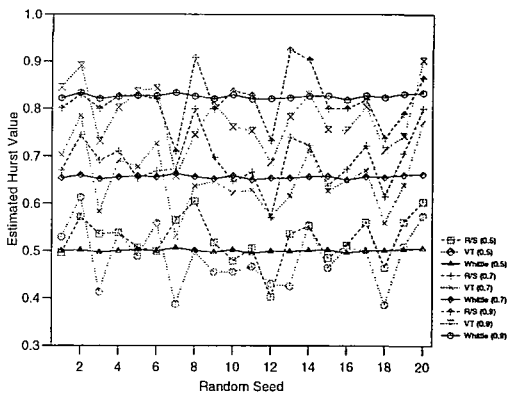


Figure 4.27: The estimated results of the SRA algorithm with different random seeds.

maximum and minimum values for all cases, Whittle's MLE shows the best result, no matter whether 0.5, 0.7, or 0.9 is chosen.

Similar to the previous algorithms, the results are depicted in Fig. 4.26. The results are obtained from the sample traces which are generated with the selected random seed and the changing Hurst parameters. In this case all curves are very close to being ideal when the Hurst parameter is less than 0.9. When the Hurst parameter is near 0.5, the results using Whittle's MLE fit the target values quite well, but the gap between the ideal value and the result gradually increases. The results using the R/S statistic almost match the 'Whittle' line until $H = 0.9$. For the VT plot, the results gradually approach the ideal values, but turn away when the Hurst value is larger than 0.75. The results with different random seeds are depicted in Fig. 4.27. For this algorithm the results using the R/S statistic and the VT plot fluctuate, whereas those for Whittle's MLE are very stable.

4.2.6 The Weierstrass-Mandelbrot Function

The WM function works very differently from the previous algorithms because it operates in the frequency domain whereas the previous algorithms operate in the time domain. The analytical results of sample traces are listed in Table 4.5. As usual, the mean value using the R/S statistic is initially higher than $H = 0.5$. Thereafter, it fits $H = 0.7$ quite closely. Finally, the mean value is lower than $H = 0.9$. For the VT plot all three different average values are significantly lower than the ideal Hurst parameters. When the Hurst parameter is set to 0.9, the mean results using the VT plot are lower than 0.7. Their 95% confidence interval is also very wide. The results using Whittle's MLE show that the mean values approach the target values when the Hurst parameter increases, while the mean value is higher than the ideal value of $H = 0.5$.

In Fig. 4.28 the results of the Q-Q plot are illustrated. All the scattering dots fit the straight lines very well. The worst case is where the Hurst parameter is 0.99. However, there is no significant difference between the line and dots. Fig. 4.29 - 4.31 depict the graphical results of the three estimators. The results using the R/S statistic and the reference line almost overlap for the first two diagrams, but when the Hurst parameter is set to 0.9 and 0.99, the values are very different from the target values. Similar to

Hurst Parameter		Mean	Maximum	Minimum	95% CI
R/S Statistic	0.5	0.5313647	0.588743	0.436694	0.08337487
	0.7	0.6787349	0.772595	0.543583	0.1146415
	0.9	0.8022376	0.986783	0.653322	0.1422364
VT Plot	0.5	0.4066917	0.657202	0.274087	0.1866828
	0.7	0.547818	0.782897	0.416097	0.1881712
	0.9	0.6696327	0.867137	0.553776	0.1753955
Whittle's MLE	0.5	0.5270633	0.5334061	0.5214302	0.005938396
	0.7	0.7136446	0.7199629	0.7079022	0.006028589
	0.9	0.906207	0.9125305	0.9005111	0.00606219

Table 4.5: Analytical results of the WM function.

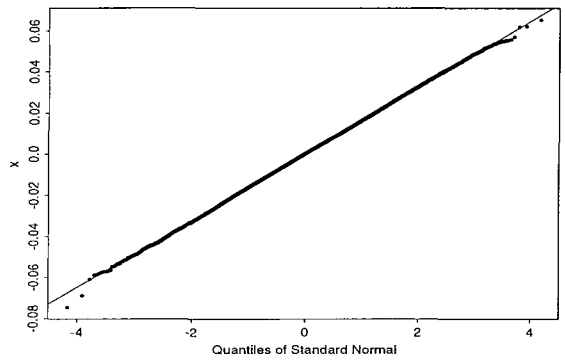
previous algorithms the results using the VT plot, shown in Fig. 4.30 (c) and (d), fit poorly with the target values when $H = 0.9$ and 0.99. In Fig. 4.31 the graphical results using Whittle's MLE with aggregation levels illustrate that the higher the Hurst parameter is, the better the curves fit. The first three diagrams show that the values fluctuate with different aggregation levels. Where there is the highest degree of self-similarity results show a good degree of stability and fits the reference line relatively well.

When investigating the chosen random seed over the Hurst parameter range, the 'Whittle' line, shown in Fig. 4.32, approaches the ideal line very closely, especially when the Hurst parameter is larger than 0.9. At $H = 0.5$ the results are the worst case, but then gradually approach the ideal value. In fact, this interval is the main concern of researchers because the observed real-world traffic with high load exhibits high Hurst values. With this selected random seed, the results using the VT plot are the worse case. The 'VT' line is always lower than the ideal values, especially, at high Hurst values. At first the curve of the R/S statistic gradually parts from the ideal values, but near $H = 0.8$ it suddenly takes a sharp turn and becomes considerably worse. When the Hurst parameter is close to 0.5, the analytical result is larger than the ideal value. However, it gradually decreases and crosses the ideal line. After that, the gap between them increases more quickly. Due to the above attributes the results using Whittle's MLE are the best ones. As with the previous algorithms, the results of the twenty different traces are illustrated in Fig. 4.33.

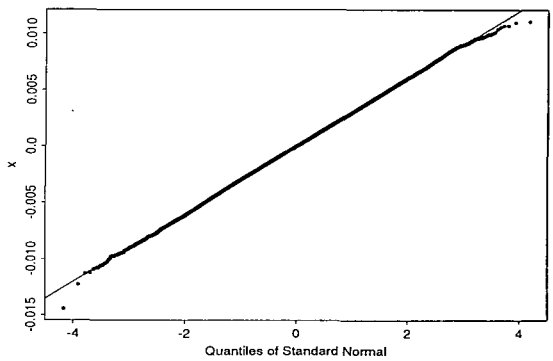
4.2.7 Whittle's Approximate Approach

As with the WM function this algorithm also generates samples in the frequency domain which are then transfered into the time domain using the inverse Fourier transform. This algorithm is derived from Whittle's MLE. At first an approximation is developed to simplify the computational effort of the estimator, then it is easy to follow the concept to generate samples by using the approximation estimator.

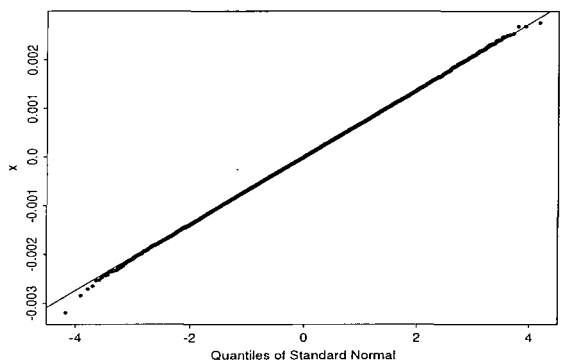
Fig. 4.34 - 4.37 are the graphical results of the chosen traces with $H = 0.5, 0.7, 0.9$, and 0.99. The Q-Q plots of Whittle's approximate approach in Fig. 4.34 exhibit very different traces. The quantiles of the traces still fit the normal distribution very well, but at both ends there are quantiles that are extremely distant from the normal distribution. This divergence is caused by the generating algorithm. Whittle's



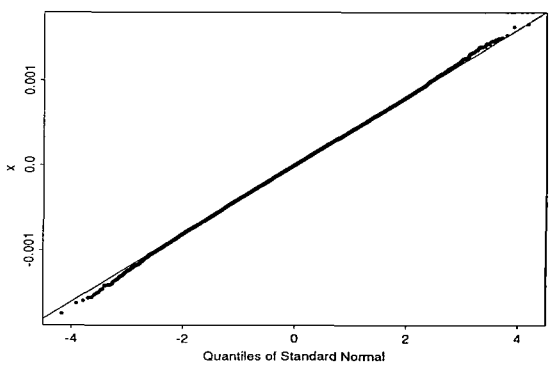
(a) H = 0.5



(b) H = 0.7

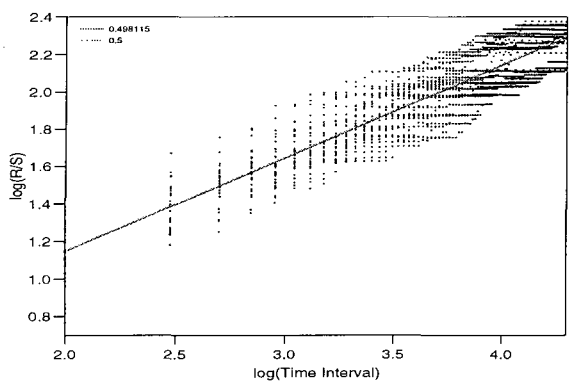


(c) H = 0.9

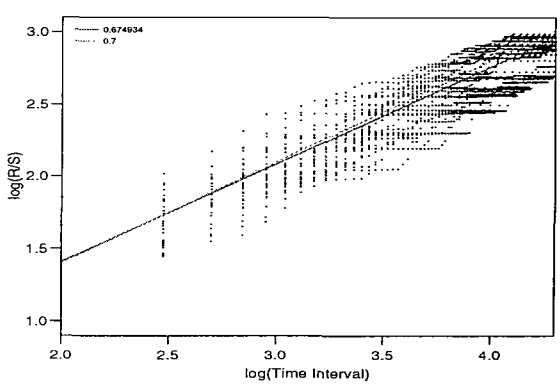


(d) H = 0.99

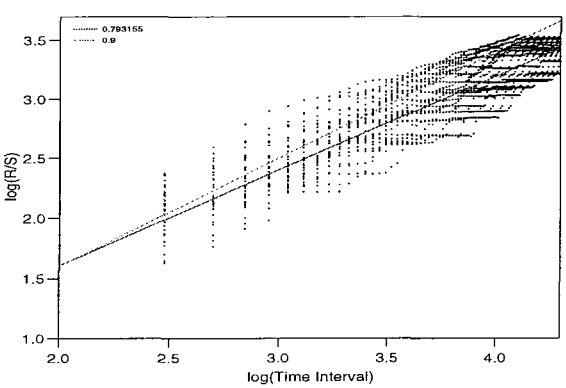
Figure 4.28: The Q-Q plots of the sample traces generated by the WM function.



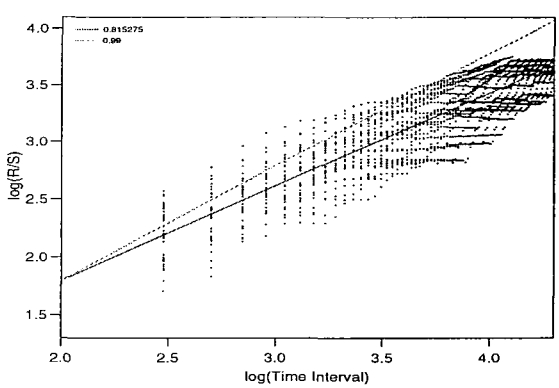
(a) H = 0.5



(b) H = 0.7

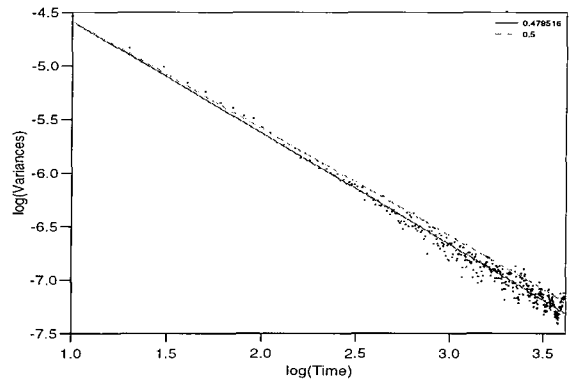


(c) H = 0.9

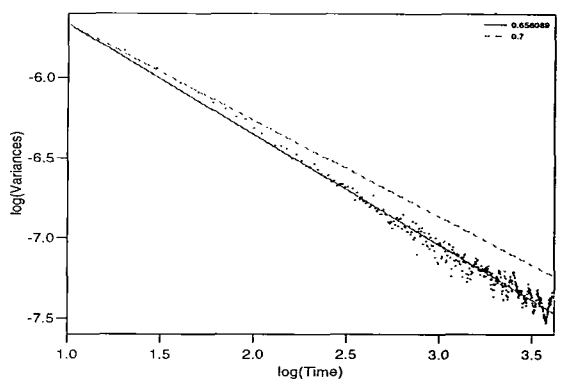


(d) H = 0.99

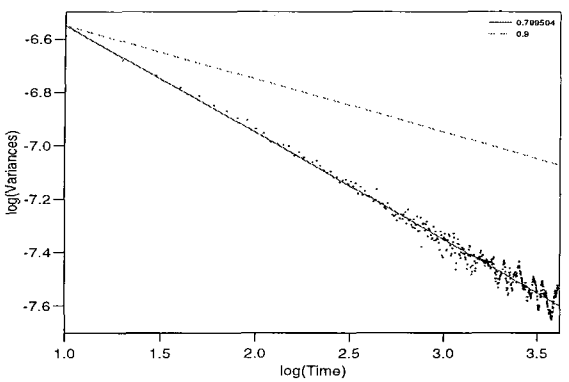
Figure 4.29: The pox plots of the R/S statistic of the sample traces generated by WM function.



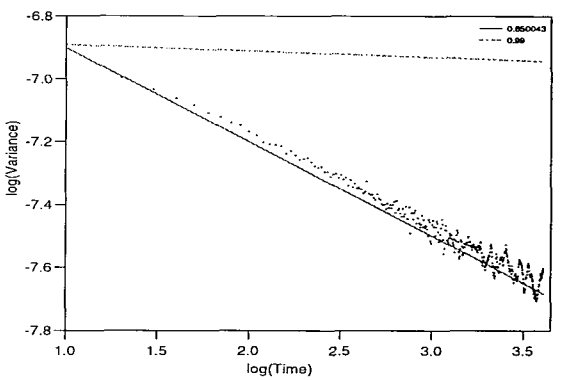
(a) H = 0.5



(b) H = 0.7

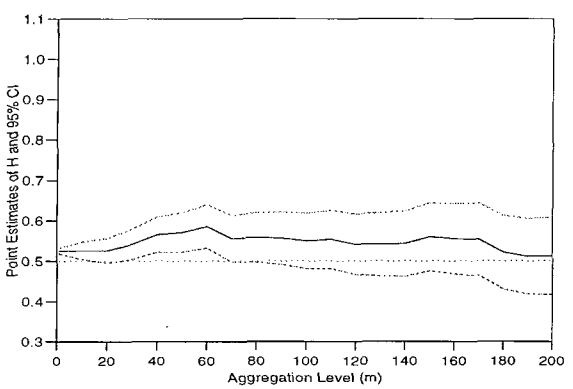


(c) H = 0.9

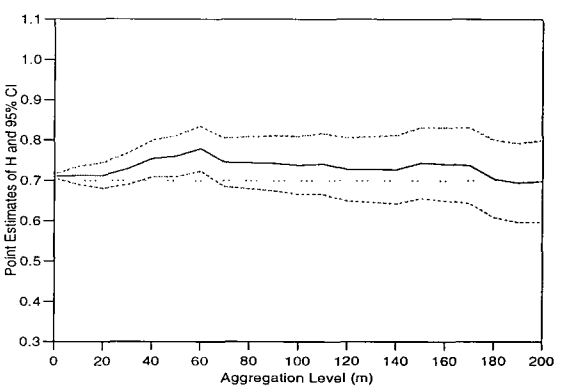


(d) H = 0.99

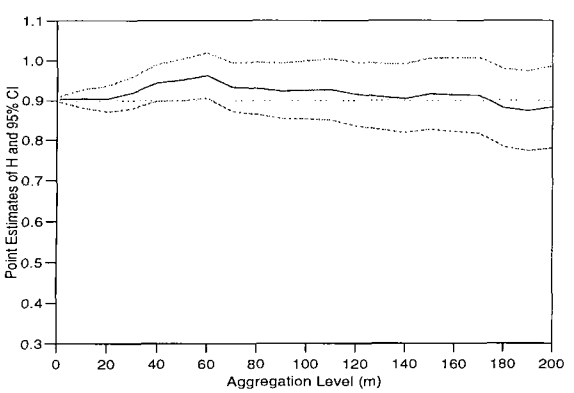
Figure 4.30: The VT plots of the sample traces generated by the WM function.



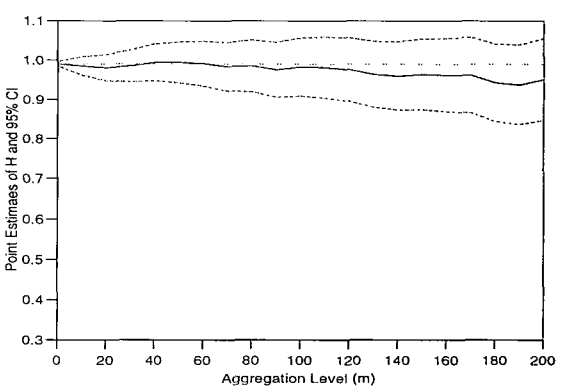
(a) H = 0.5



(b) H = 0.7



(c) H = 0.9



(d) H = 0.99

Figure 4.31: The Hurst values using Whittle's MLE of the sample traces generated by WM function.

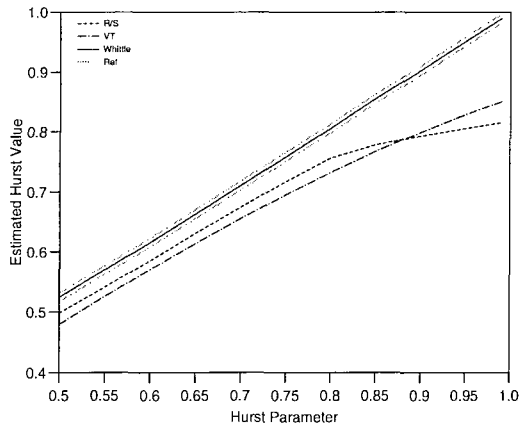


Figure 4.32: The estimated results of the WM function with different Hurst parameters.

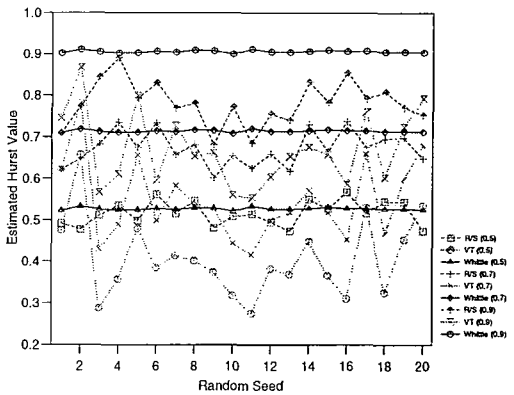


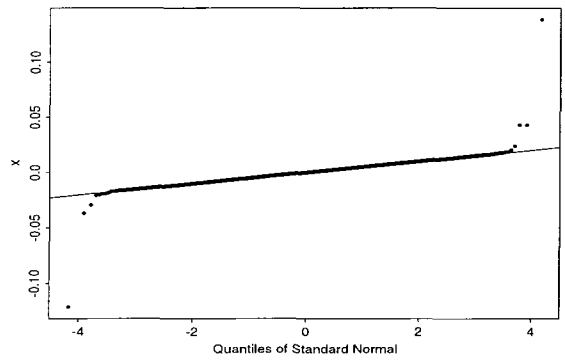
Figure 4.33: The estimated results of the WM function with different random seeds.

Hurst Parameter		Mean	Maximum	Minimum	95% CI
R/S Statistic	0.5	0.5163246	0.567309	0.472479	0.05854247
	0.7	0.6717431	0.737292	0.601846	0.07967137
	0.9	0.7804224	0.890397	0.680413	0.1060564
VT Plot	0.5	0.5065585	0.633498	0.318915	0.1740812
	0.7	0.6584451	0.776776	0.52292	0.1451696
	0.9	0.7946734	0.89281	0.69618	0.1225351
Whittle's MLE	0.5	0.5717692	0.5769492	0.5673334	0.005265109
	0.7	0.7517701	0.7569672	0.7471196	0.005442897
	0.9	0.926514	0.931481	0.9218944	0.005398148

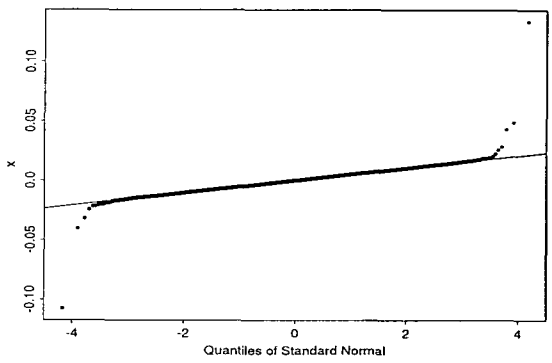
Table 4.6: Analytical results of Whittle's approximate approach.

approximate approach generates the samples in the frequency domain using an exponential distribution for magnitude and uniform distribution for phase. Unlike other algorithms they use normal distributions with transformations to generate FBM or FGN sample traces. Even the WM function still uses normal distribution to generate the magnitude part of the samples. For the pox plot of the R/S statistic in Fig. 4.35 the results and the reference line of the last two diagrams almost match when $H = 0.9$. However, the slopes of the result and the reference line have diverged again when $H = 0.99$. The VT plot in Fig. 4.36 exhibit very poor results, making it one of the two worst cases in our simulations. The results of the diagrams are almost 0.2 less than the target values. Finally, examining Whittle's MLE with different aggregation levels in Fig. 4.37, the results of $H = 0.9$ exhibit a very close match while the other three diagrams do not fit the reference lines very well. When the Hurst value is set to give the highest degree of self-similarity in our study, the results are even lower than the results with $H = 0.9$.

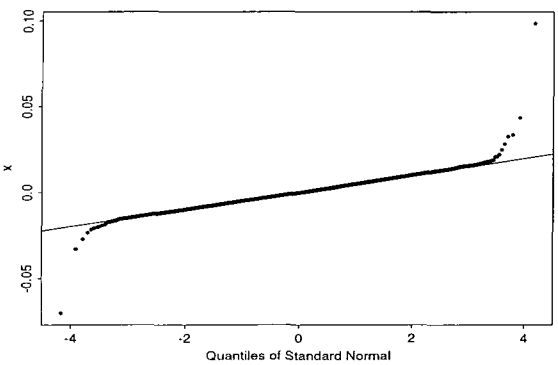
The selected case over the range $H = 0.5$ to 0.9 the analytical result is illustrated in Fig. 4.38. In this diagram the 'VT' line is much lower than the ideal line. Unfortunately, the selected case is one of the two worst cases of the twenty traces with different random seeds. When $H = 0.5$ the analytical result is the worst one, the results of $H = 0.7$ and $H = 0.9$ are the second worst case in the twenty



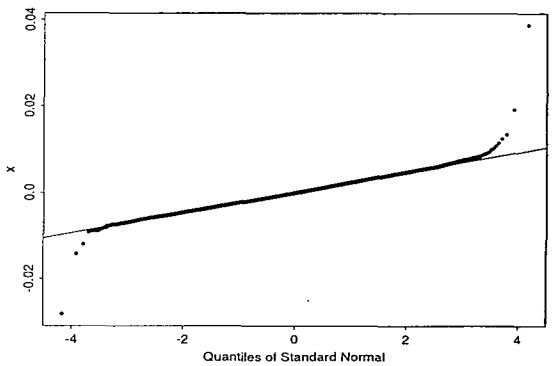
(a) H = 0.5



(b) H = 0.7

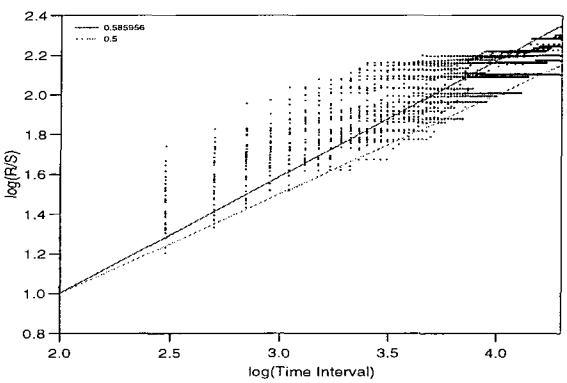


(c) H = 0.9

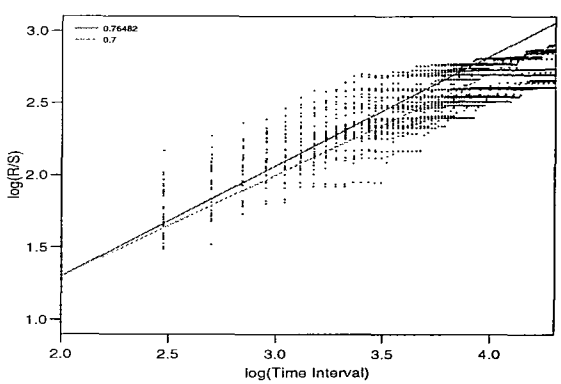


(d) H = 0.99

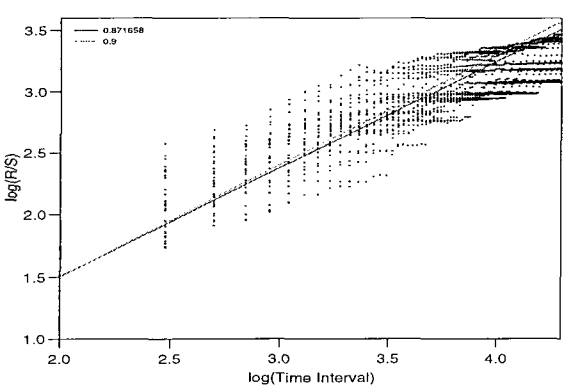
Figure 4.34: The Q-Q plots of the sample traces generated by Whittle's approximate approach.



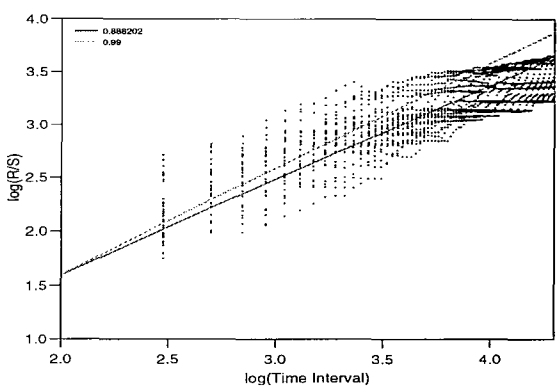
(a) H = 0.5



(b) H = 0.7

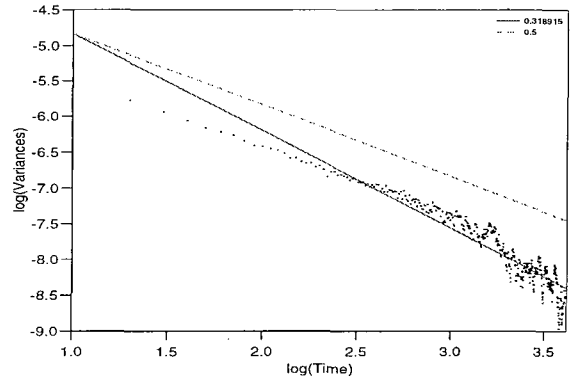


(c) H = 0.9

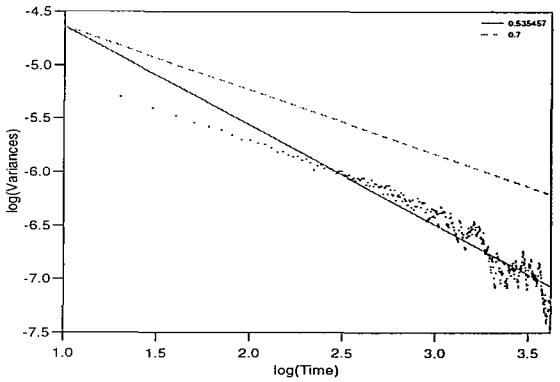


(d) H = 0.99

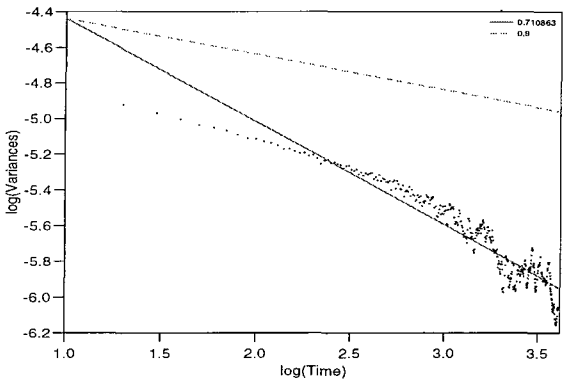
Figure 4.35: The pox plots of the R/S statistic of the sample traces generated by Whittle's approximate approach.



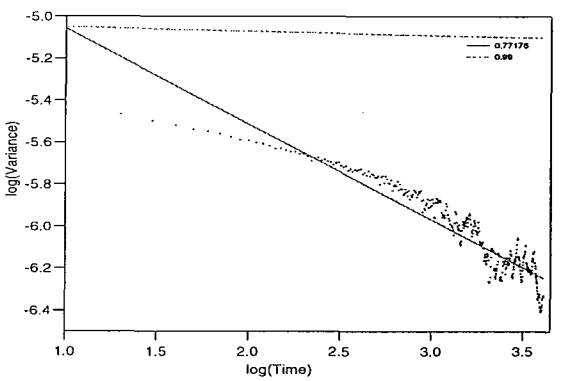
(a) H = 0.5



(b) H = 0.7

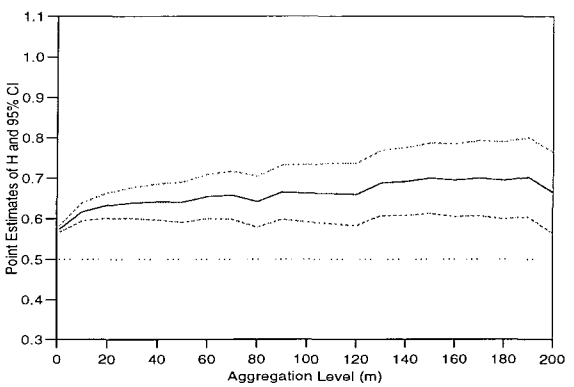


(c) H = 0.9

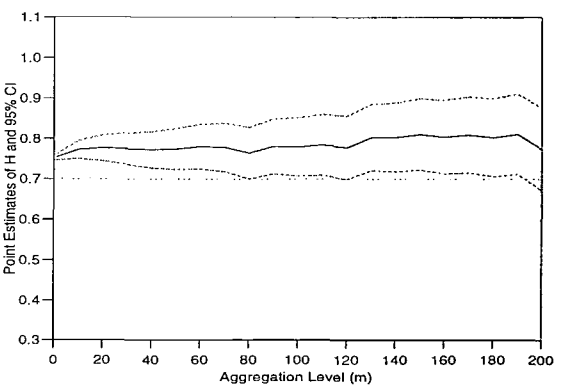


(d) H = 0.99

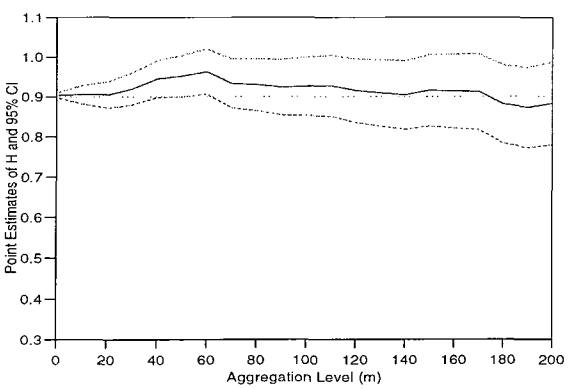
Figure 4.36: The VT plots of the sample traces generated by Whittle's approximate approach.



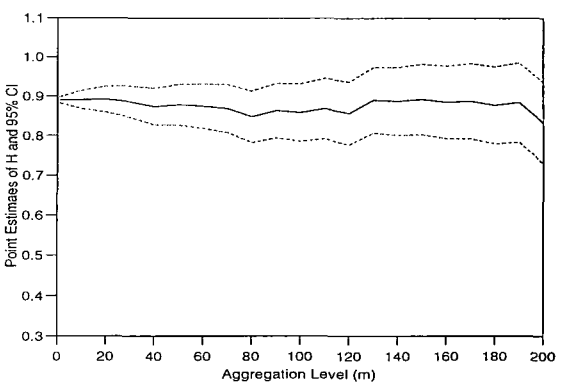
(a) H = 0.5



(b) H = 0.7



(c) H = 0.9



(d) H = 0.99

Figure 4.37: The Hurst values using Whittle's MLE of the sample traces generated by Whittle's approximate approach.

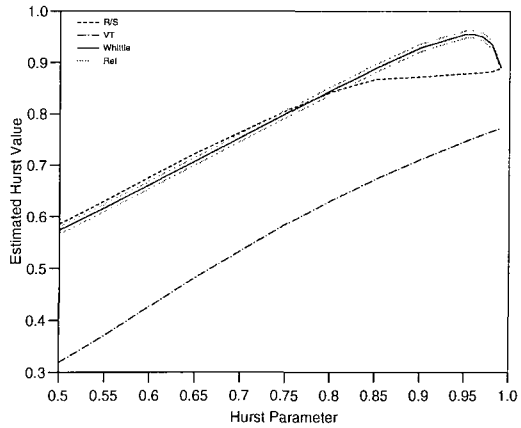


Figure 4.38: The estimated results of Whittle's approximate approach with different Hurst parameters.

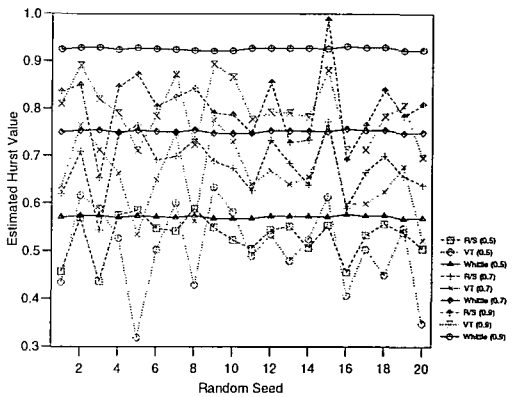


Figure 4.39: The estimated results of Whittle's approximate approach with different random seeds.

different simulations. However, the other traces maintain similar slopes without significant difference except at their point of origin. As usual, the results using the R/S statistic is larger than the ideal value near $H = 0.5$ and then smaller than the ideal value near $H = 0.9$. There is a sharp turn in the curve 'R/S' near 0.85. In this case the results using Whittle's MLE are the best. The 'Whittle' line exhibits a very weird phenomenon. At the beginning the analytical value is near 0.6 rather than the target value $H = 0.5$. Then the line gradually closes to the ideal values. As the Hurst parameter is reaching 0.9 the 'Whittle' line approaches the target values very quickly. When the Hurst parameter is around 0.95 the 'Whittle' line exhibits a dramatical change. It decreases very quickly as the Hurst parameter increases.

The mean analytical results of the twenty traces with different random seeds are listed in Table 4.6. When $H = 0.5$ the mean results using the R/S statistic and Whittle's MLE are higher than the expected value. For this case the mean result using the VT plot approaches the target value more accurately. However, its difference between the maximum and minimum values remains quite large. While the mean result using Whittle's MLE is still much higher than the target value, $H = 0.7$, that using the R/S statistic is lower than 0.7. In the case of $H = 0.9$ the mean result using Whittle's MLE remains larger than the target value, and the mean result using the R/S statistic is slightly lower, while the mean result using the VT plot shows the worst result. The values with different random seeds are illustrated in Fig. 4.39. The results using the R/S statistic and the VT plot are still highly dependent on the random seeds.

4.3 Comparisons of the Estimation Results

In Sec. 4.2 the estimated Hurst values of the algorithms have been examined individually using three different Hurst estimators. In this section comparisons between them will be investigated. The comparisons of the results are based on the means and 95% confidence intervals in Tables 4.1 - 4.6 and goodness-of-fit test. Therefore, the three diagrams, Fig. 4.40 - 4.42, are illustrated by the values listed in above tables. The asterisk, circle, and star marks represent the Hurst parameters, $H = 0.5, 0.7$, and 0.9, respectively. All the marks are the mean results of sample paths with the twenty different random seeds which were applied to the six algorithms. The barlines across the marks are the 95% confidence intervals. In addi-

tion, the Q-Q plots of the selected traces, with all algorithms illustrated, are used to evaluate the quality of fit of the normal distribution.

4.3.1 Comparisons of the Q-Q Plot

The Q-Q plots of the sample paths with different algorithms and different Hurst parameters are discussed in Sec. 4.2. For the dFGN algorithm, different Hurst parameters do not significantly affect the results which still match the normal distribution. The Q-Q plots of the fFGN algorithm still match the straight line relatively well. However, when the Hurst parameter increases, the tails of the scattering dots gradually part from the reference line. For the RMD algorithm, the scattering dots match the reference line well when $H = 0.5$, but becomes increasingly worse when the Hurst parameter increases. For the SRA algorithm the scattering dots of the Q-Q plots match the reference line very closely. The Q-Q plots of the WM function exhibit very good quality. For Whittle's approximate approach most of the scattering dots match the reference line relatively well. However, due to its different generating methodology the tails of the Q-Q plots are further away from the reference line.

When examining all the Q-Q plots, the majority of the results exhibit a relatively close fit. Of them all four Q-Q plots of the dFGN algorithm match the reference line very well. The results of the WM function are marginally worse than those of the dFGN algorithm. The quality of the fFGN algorithm is slightly worse than that of the SRA algorithm. The quality of the RMD algorithm is significantly worse than the improved version. The quality of Whittle's approximate approach is very hard to determine because of its parted tail.

4.3.2 Comparisons of the R/S Statistic

The mean values and 95% confidence intervals of the twenty Hurst values by the R/S statistic is depicted in Fig. 4.40. When examining the bottom row of marks whose original Hurst parameter is set to 0.5, all the six marks are higher than the expected value. The mean result of the RMD algorithm has the best result, if the effect of the confidence interval taken into consideration. The dFGN algorithm has the shortest bar line, while the WM function has a slightly better mean value. In point of fact, for $H = 0.5$ the results of the dFGN, fFGN, RMD and SRA should be very similar because the algorithms merely retrieve the samples from one or many Normal-distributed random samples. The aggregation of random variables makes them different. For instance, the dFGN algorithm retrieves $n = 10$ samples from a Gaussian random number generator with old $(M-1)n$ substates to generate a new sample. The SRA algorithm successively adds weighted samples retried from a random number generator. The WM function generates samples with normal and uniform distributions. Thus, the result using the R/S statistic still can maintain the accuracy and stability well.

With $H = 0.7$, almost all of the results fit the reference line very well in comparison to the two Hurst parameters, $H = 0.5$ and 0.9, especially the dFGN and fFGN algorithms whose mean results are on the reference line. In comparison the dFGN algorithm is significantly better than the fFGN algorithm because its confidence interval is much shorter than that of the fFGN algorithm. The mean result of the SRA algorithms just slightly part from the reference line. However, its confidence interval is much wider

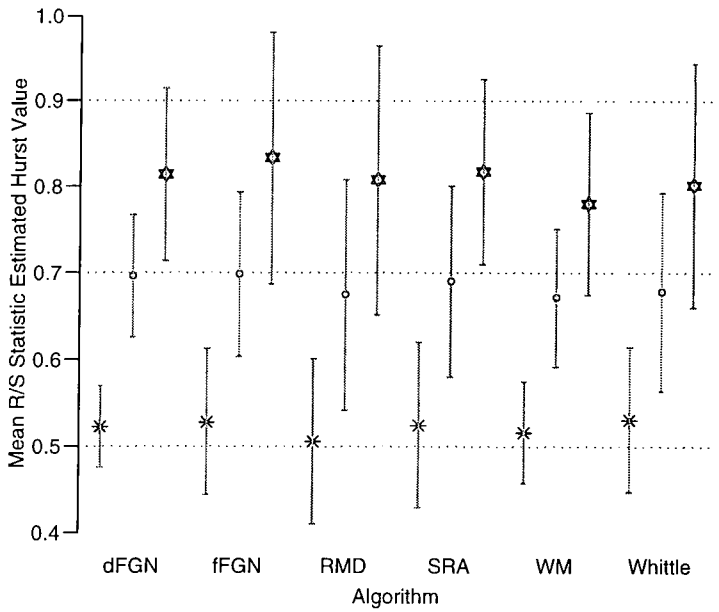


Figure 4.40: The analytical results using the R/S statistic for all algorithms.

than the WM function whose mean value is the worst case in the six algorithms. The mean results of the RMD algorithm and Whittle's approximate approach are the worst two, although their mean values are slightly better than that of the WM function, their confidence intervals are much wider than the WM function.

On the top row of Fig. 4.40 the mean results with $H = 0.9$ are illustrated. The dFGN and fFGN algorithms exhibit the best two fits. Although the fFGN algorithm has the best mean result, its confidence interval is significantly worse. The mean result of the SRA algorithm is slightly better than the dFGN algorithm, but its confidence interval is wider. The confidence interval of the WM function is very short, but its mean value is the worst of the algorithms studied. The RMD algorithm is probably the worst one because it has the widest confidence interval and its mean result does not have significant difference from the WM function.

The overall examination of the algorithms with the R/S statistic exhibits that the dFGN algorithm has the best result. Although its mean values of the three different Hurst parameters are not as good as some other algorithms, its confidence intervals are always shorter than the others. The mean values of the fFGN algorithm are best when the Hurst value is set to 0.7 and 0.9. However, its confidence intervals are much wider than those of the dFGN algorithm. The result of Whittle's approximate approach is the worst case when compared with other algorithms because its mean values are the worst or near worst case. In addition, its confidence intervals are very wide, especially when $H = 0.9$. The results of the SRA algorithms highlight that its mean values with higher degree of self-similarity are better than the RMD algorithm. It is also better than the WM function due to its more accurate mean values when the Hurst parameter is 0.7 and 0.9.

In addition, almost all of the six algorithms exhibit that the mean values with $H = 0.7$ are more accurate than the cases of the $H = 0.5$ and 0.9. The mean values with $H = 0.5$ are always larger than the target Hurst value. For $H = 0.7$ the dFGN , fFGN, and SRA algorithms are reasonably accurate.

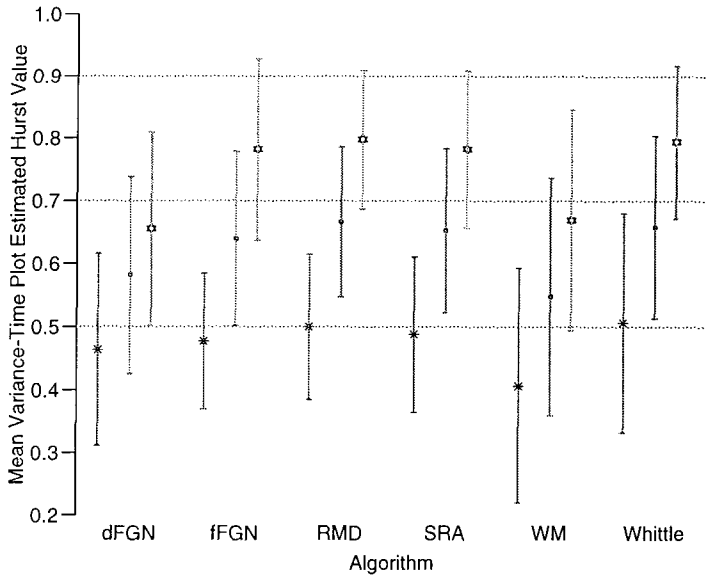


Figure 4.41: The analytical results using the VT plot for all algorithms.

However, all the mean values with $H = 0.9$ are much smaller than the target values. Their confidence intervals become wider as the Hurst parameter increases.

4.3.3 Comparisons of the Variance-Time Plot

In Fig. 4.41 the analytical results of the six algorithms using the VT plot are illustrated. The RMD algorithm exhibits the best result when $H = 0.5$. Its mean value is very close to the reference line $H = 0.5$ and its confidence interval is also the shortest. The mean values of the dFGN, fFGN algorithms, and Whittle's approximate approach are also very near the reference line, but when examining their confidence intervals the dFGN algorithm is better than the other two, while the fFGN algorithm is better than Whittle's approach. For the confidence intervals, those of the SRA algorithm are much shorter than the fFGN algorithm, but their mean values do not approach the reference line as close as those of the fFGN algorithm. Thus, it is very difficult to distinguish which of them is better. Certainly, the worst case is the WM function. Its mean value is as low as $H = 0.4$ and its confidence interval spans almost from $H = 0.2$ to $H = 0.6$.

As with the case of $H = 0.5$ the result of the RMD algorithm exhibits the best performance not only at its mean value but also its confidence interval when $H = 0.7$. The mean value of Whittle's approximate approach is slightly better than that of the SRA algorithm but the reverse is the case for the the confidence intervals. The mean result of the fFGN algorithm is significantly better than that of the dFGN algorithm and the WM function. For the WM function both its mean and confidence interval are the worst case. Its mean value is near $H = 0.55$ which is much lower than the Hurst value expected.

When examining the top row of Fig. 4.41 there is no mean value larger than $H = 0.8$. As with the previous two Hurst parameters the best is the RMD algorithm because its mean value is the closest to $H = 0.9$ and its confidence interval is also the shortest one. The result of Whittle's approximate approach is ranked second. Its mean value is slightly better than those of the fFGN and SRA algorithms and its confidence interval is also a little bit shorter. As with the previous two cases the mean value of the WM

function is very poor because it is lower than $H = 0.7$. In addition, its confidence interval covers a wide range from $H = 0.5$ to $H = 0.75$. However, the mean value of the dFGN algorithm is even worse than that of the WM function.

From above comparisons the RMD algorithm has the best performance no matter which Hurst parameter is used. The mean values of the RMD algorithm are the closest to the expected Hurst values in comparison to those of the other algorithms, although the mean value is still far from the expected Hurst value where $H = 0.9$. Whittle's approximate approach would be the second choice. All its mean values are only marginally worse than those of the RMD algorithm. It is also the only algorithm whose confidence intervals become shorter as the Hurst parameter increases. In Fig. 4.41 it is very difficult to distinguish between the performance of the fFGN and SRA algorithms. For $H = 0.9$ the mean value of the SRA algorithm is better than that of the fFGN algorithm, but its confidence interval is wider than that of the fFGN algorithm. For $H = 0.7$ the performance of the fFGN algorithm certainly is better than that of the SRA algorithm both of the mean value and the confidence interval. For the case of $H = 0.5$ the mean value of the fFGN algorithm is better but it has a wider confidence interval. Finally, it is apparent that the performance of the WM function is the worst; no matter which Hurst parameter is used its mean values are the worst ones and the confidence intervals are the widest. The dFGN algorithm is just a little better than the WM function.

Although the results of the algorithms with the VT plot are evaluated in above paragraph, it is difficult to be totally sure that the RMD algorithm is the best one and the WM function and the dFGN algorithm are the worst ones. This is because their confidence intervals are so wide, which may mislead people. Also, where $H = 0.9$, all the mean values are significantly lower than the reference line. For the WM function and the dFGN algorithm their mean values are even lower than $H = 0.7$. Because the estimator makes some instinctive approximations to simplify the calculation, it might be oversimplified when the Hurst parameter is high.

4.3.4 Comparisons of Whittle's Approximate MLE

Compared to Figs. 4.40 and 4.41 the diagram in Fig. 4.42 has very small confidence intervals. All the confidence intervals are very small, which makes the comparisons very simple. For $H = 0.5$ the dFGN, fFGN, RMD, and SRA algorithms lie on the reference line. The WM function is worse than the previous four algorithms but is better than Whittle's approximate approach. This result is to be expected because the WM function and Whittle's approximate approach generate samples in the frequency domain and then transform to the time domain rather than generate in the time domain directly. These algorithm use a uniform distribution to obtain the arguments and either a Gaussian distribution or an exponential distribution to obtain the modulus. The process of inverse Fourier transform also causes distortions. The other four algorithms directly retrieve samples from Gaussian random number generators and apply some weighting and additive operations.

When the Hurst parameter increases to 0.7, the confidence interval slightly increases. However, it is small enough not to affect the work of comparisons. From Fig. 4.42 it is apparent that the dFGN algorithm is the best one, with the WM function better than the other four algorithms. The fFGN and

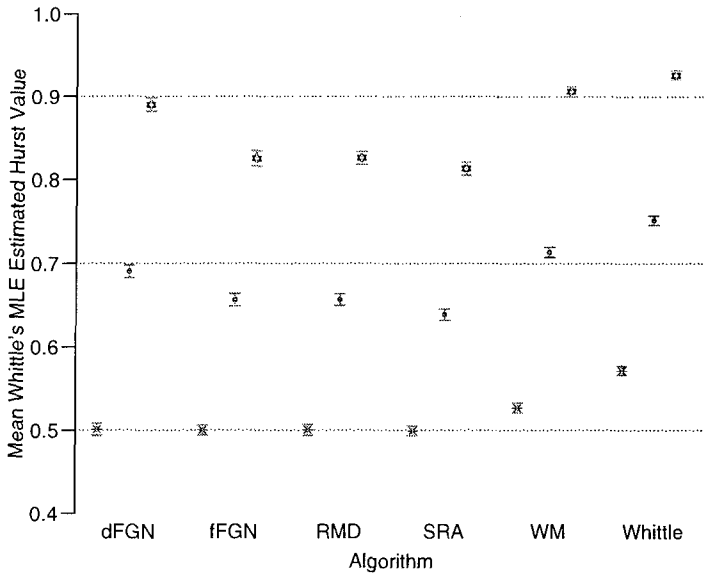


Figure 4.42: The analytical results using Whittle's MLE for all algorithms.

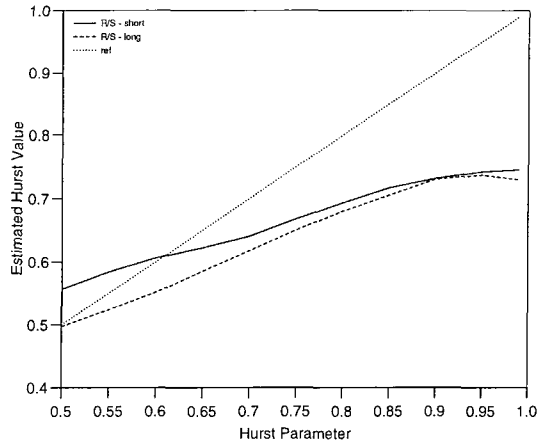
the RMD algorithms have almost the same mean values, and these are closer to the reference line than Whittle's approximate approach and the SRA algorithm.

On the top row of Fig. 4.42 the mean value of the WM function is the closest one to the reference line $H = 0.9$ while the mean value of the dFGN algorithm is slightly worse. The mean value of Whittle's approximation approach can be ranked third. Again, the results of the fFGN and SRA algorithms have almost the same mean values, and the worst one is the SRA algorithm.

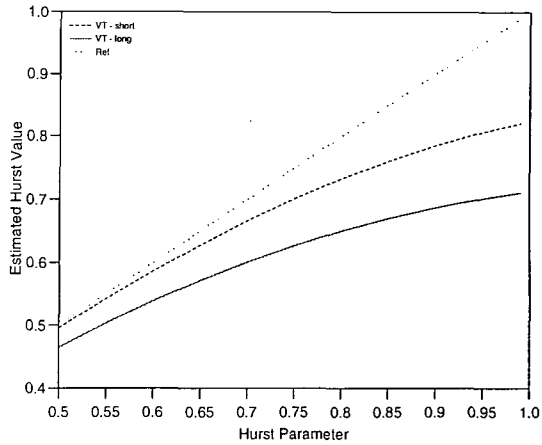
Manifestly, the dFGN algorithm has the best performance under Whittle's MLE, while the WM function is only slightly worse. Whittle's approximate approach is worse than the previous two, although the algorithm is directly derived from Whittle's MLE. There is no significant difference between the fFGN and RMD algorithms whose mean values are better than Whittle's approach when $H = 0.5$ but worse when $H = 0.7$ and 0.9. Here, Whittle's approach ranks third because the higher Hurst parameter plays a more important role in network applications. Finally, the SRA algorithm is the worst, however, the difference between the SRA and RMD algorithm is insignificant. The results do exhibit a very interesting phenomenon; the four algorithms which generate samples in the time domain have a lower Hurst values than the reference lines when the Hurst parameter is beyond 0.5. For the algorithms generating their samples in the frequency domain the mean values are always higher than the reference lines and gradually tend to the reference lines when the Hurst parameter increases.

4.4 Changing the dFGN Algorithm Simulation Parameters

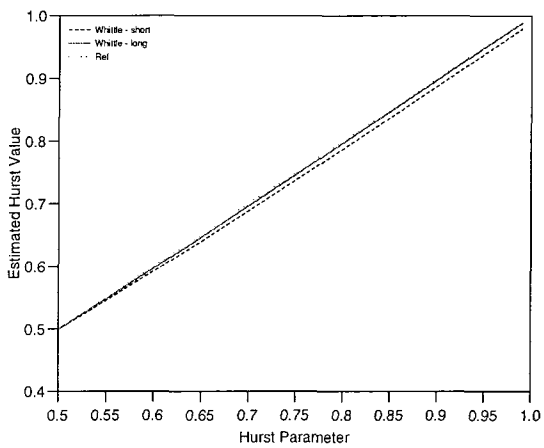
In the dFGN algorithm two parameters control the quality of simulation. From the definition of the FBM (Definition 3) the difference between two time epochs is obtained from the integration of ordinary Gaussian noise weighted by the kernel function. The range which the integral covers is from $-\infty$ to the current time, t. It means that current position is affected by any event occurring before. To simulate FBM or FGN one needs to approximate the integral representation. This creates two types of approximation



(a) The R/S Statistic



(b) The VT plot



(c) Whittle's MLE

Figure 4.43: Comparisons of the statistical analytical results of the dFGN algorithm with different subintervals and covered ranges.

errors, a 'low frequency' one due to the truncation of the limits of integration, and a 'high frequency' error caused by replacing the integral to a summation. The number of covered range M determines the 'low frequency' error and the number of sub-states, n, affects the 'high frequency' error.

In simulations previously carried out, the number of covered states is set to $M = 700$ and the number of sub-states in a state is set to $n = 10$. These values are adopted shown in the original paper [MW69]. If the two values increase, the quality of simulation will be improved. Thus, if the number of covered states is $M = 1000$ and the number of sub-states is renewed as $n = 30$, the number of samples retrieved from ordinary the Gaussian random number generator will increase from $7,000$ to $30,000$ to generate a single sample. This causes the average simulation time of 32768 samples to increase from 753.6 to 1506.9 seconds. Fig. 4.43 illustrates the results of the original and improved simulations under the selected random seed. In all diagrams the dotted lines show the target Hurst values, and all the diagrams show the results using the previously chosen random seed.

The first diagram shows the results using the R/S statistic. The result of the short-range dependent case shows that the longer the simulation the better the results. When the Hurst parameter increases, these two results gradually converge. However, the curves gradually drift away from the reference line. Previous analysis using the R/S statistic is highly dependent on the chosen random seed. Thus, it is

Hurst Parameter		Mean	Maximum	Minimum	95% CI
R/S Statistic	0.5	0.5362565	0.600374	0.450645	0.07791316
	0.7	0.7064376	0.793117	0.644358	0.07476778
	0.9	0.8234968	0.948899	0.729946	0.1082109
VT Plot	0.5	0.4597866	0.609787	0.33634	0.1341218
	0.7	0.5990193	0.747038	0.481609	0.1309809
	0.9	0.6900486	0.830467	0.587867	0.12259
Whittle's MLE	0.5	0.4996168	0.5042672	0.4942388	0.005619327
	0.7	0.6963732	0.7014564	0.6902849	0.006162258
	0.9	0.8979003	0.9038126	0.8905684	0.00625812

Table 4.7: Analytical results of the dFGN algorithm with $M = 1000$ and $n = 30$.

difficult to gauge what effect the simulation time has upon the results.

Fig. 4.43 (b) shows the results using the VT plot. Where the longer simulation time significantly improves the results. As the Hurst parameter increases, the more the longer simulation improves. However, a large gap between the reference line and the 'VT-long' line still exists when the Hurst parameter is large. Due to the variation of the results in the VT plot it is difficult to conclude that the longer simulation will improve the results. The only thing guaranteed is that the result is always less than the expected value as the Hurst parameter approaches to 1.

The results using Whittle's MLE is shown in Fig. 4.43 (c). In this diagram the reference line is not shown because the lines are very close to the ideal values. The 'Whittle-long' line shows that the longer simulation time using the dFGN algorithm improve the results. The results improve more as the Hurst parameter increases, however, the improvement is not so significant.

In addition, the set of twenty random seeds were also tested using simulation to examine their affect. Table 4.7 lists the results of the three estimators with different Hurst parameters. Comparing Table 4.1 and 4.7, the results using the R/S statistic do not improve when $H = 0.5$ and 0.7 and the differences between the maximum and minimum values become worse. For $H = 0.9$ the difference between the maximum and minimum values does not improve, although the mean value is more accurate. The mean results using the VT plot move toward the expected Hurst values, except at $H = 0.5$. However, all the mean values are still significantly lower than the expected values. Meanwhile, among the twenty sample traces there are 8, 6 and 7 results which are worse than the shorter simulations with $H = 0.5, 0.7$ and 0.9, respectively. There were 3 random seeds which generated worse results no matter what Hurst parameter was. For Whittle's MLE all the mean results improve and the confidence intervals are smaller.

4.5 Empirical Evaluation of the Estimators

From previous analysis in Sec. 4.3 the results using the VT plot and Whittle's MLE lead to conflicting conclusions. In addition, the results using the VT plot exhibit very low mean values, especially the mean values of the dFGN algorithm and the WM function which are much lower than those for the other

algorithms. However, these results are the best two with Whittle's MLE. In addition, the results using the VT for all algorithms have very wide confidence interval, but those using Whittle's MLE are very small. The R/S statistic somewhat similar to that of the Whittle's MLE. The mean values from the dFGN algorithm gives the best result.

As far as it is known, the VT plot uses some simple approximation to reduce the calculation effort, but Whittle's approximate MLE has a more rigorous mathematical derivation. Due to its mathematical complexity and difficulty an empirical study was used to examine the reliability of these two estimators. In Fig. 4.9, 4.15, 4.21, 4.27, 4.33, and 4.39 all results obtained from a set of twenty different random seeds are illustrated representing the six algorithms, individually.

From the results with different random seeds the results using the R/S statistic and the VT plot are highly dependent on the random seeds. For the R/S statistic the results tend to be higher than the ideal value when the Hurst parameter is set to $H = 0.5$ and tend to be lower than the target value when the Hurst parameter is $H = 0.9$. Of the algorithms the results using the R/S statistic, the RMD algorithm and Whittle's approximate approach all fluctuate significantly more than the others. When examining their ranges the dFGN algorithm and the WM function produce a smaller span. In addition, the R/S statistic results magnify their range when the Hurst parameter increases.

The results using the VT plot are worse than that using the R/S statistic. The VT plot results tend to a little bit lower than 0.5 when the Hurst parameter is set to $H = 0.5$. When the Hurst parameter increases, the results using the VT plot become lower than the setting of the Hurst parameter. The ranges of each of the algorithms using the VT plot do not have significant difference for the different Hurst parameters. The results of the WM function and the dFGN algorithm exhibit a larger spanning range in comparison to the other algorithms.

The results using Whittle's MLE exhibit a significantly better performance. For each algorithm all results for different Hurst parameters tend to be a straight line, which means the changing of random seeds do not affect the estimation significantly. Unlike the R/S statistic and the VT plot, the estimator has a more rigorous mathematical derivation. It uses an optimization technique with goodness-of-fit statistic instead of simple and instinct approximation. Thus, the results using Whittle's MLE should be more reliable than the other two estimators.

The dFGN algorithm is directly derived from the definition of FBM. The summation which was substituted an integral causes 'high frequency' and 'low frequency' errors. If the larger covered range and sub-states are used, the errors will reduced. From Sec. 4.4 the results using Whittle's MLE improves slightly, as was expected, when the parameters is changed from $M = 700$ and $n = 10$ to $M = 1000$ and $n = 30$. However, the results using the R/S statistic do not show that any improvement in Fig. 4.43 (a). The results using the VT plot exhibit improvement in Fig. 4.43 (b) but the results shown in Fig. 4.9 show that the results are very chaotic and there are three random seeds which always make the results of longer simulations worse than those of shorter ones. Thus, the VT plot results illustrate that not all the cases arrive at the same conclusion. In Fig. 4.43 (c) the results using the Whittle's MLE are significantly improved.

From the above discussion it has been shown that the VT plot might mislead the results of estimation of FBM or its increments, the FGN. Fig. 4.44 shows the VT plots for the dFGN algorithm, the SRA

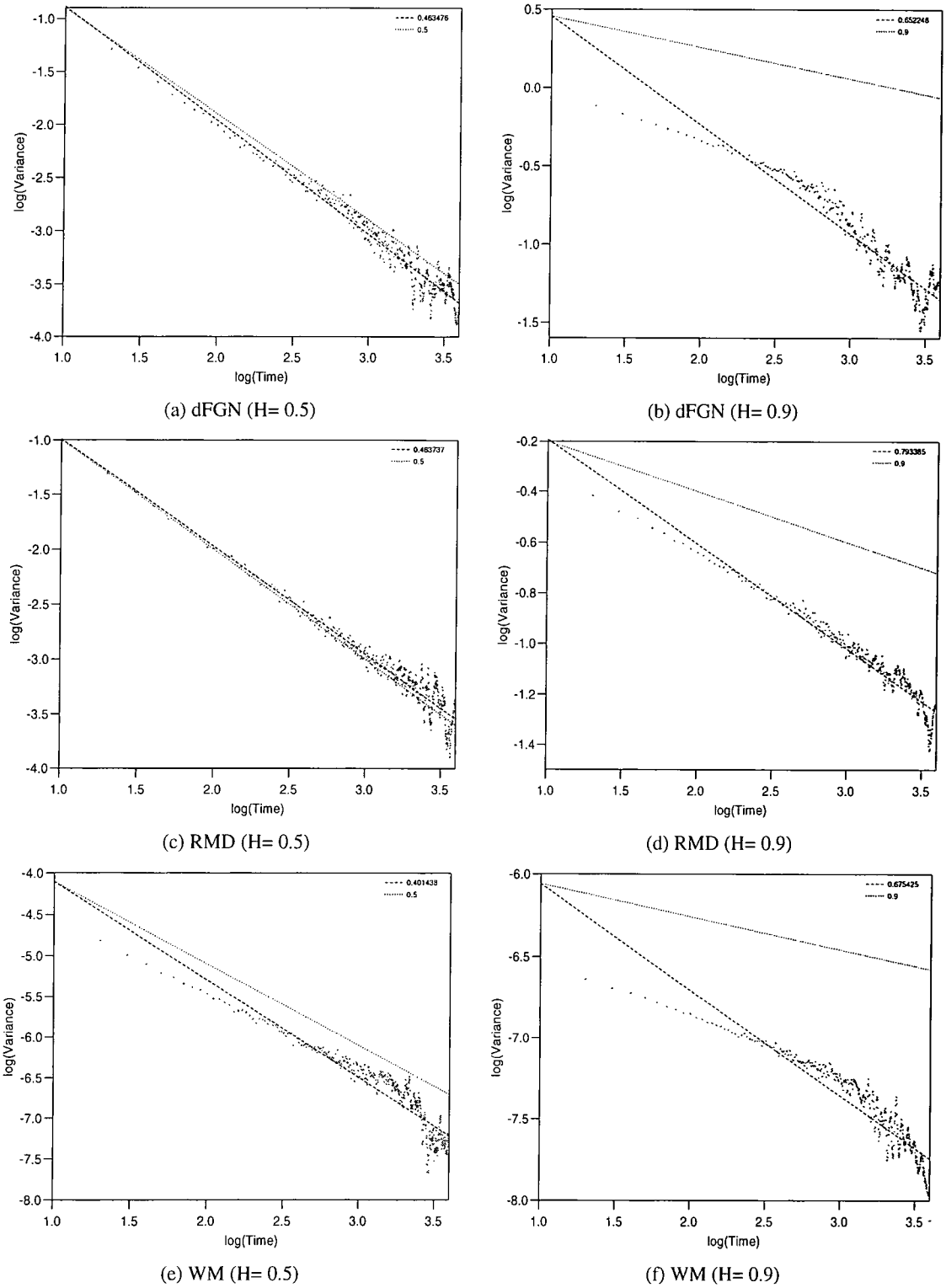
algorithm, and the WM function, using the Hurst parameters of 0.5 and 0.9. They are selected because the dFGN algorithm and the WM function present the best results using Whittle's MLE but the worst ones using the VT plot and the RMD algorithm exhibits the best performance using the VT plot but almost the worst one using Whittle's MLE. For each algorithm the graphical results with two Hurst parameters, $H = 0.5$ and 0.9, are shown.

All the diagrams are chosen from different random seeds. The criterion of choosing the random seeds is to use the random seeds which are most similar to their average values. For instance, the Fig. 4.44 (c) is the 11th random seed for $H = 0.5$ in Fig. 4.21. Fig 4.44 (d) is the 17th random seed for $H = 0.9$ in Fig. 4.21. When examining the graphical results with $H = 0.5$, the result of the RMD algorithm fits the reference line quite well while that of the dFGN algorithm is slightly separated. However, for the result of the WM function the slope of the result and reference line are quite different. From the VT scattering dots in Fig. 4.44 it can be seen that the WM function has a nonlinear tail while the others are quite straight.

When $H = 0.9$ all the three algorithms exhibited do not fit the reference line in a tolerable range. All the slopes of the results are quite different from those of the reference lines. When comparing these three diagrams on the right-hand side, to argue that the results of the RMD algorithm ares better than the other two does not convey any significant meaning. When looking at the scattering dots of the three diagrams on the right hand side, all the curves have nonlinear tails. The result for the RMD algorithm is better than either of the other two algorithms because the curve does not have such a heavy nonlinear tail. If the scattering dots chosen are not larger than 2.5, the slope of the line formed by the dots will be significantly better than the results across the whole range. In other words, the shorter the time interval picked, the better the results exhibited. However, shorter time intervals mean the correlation with previous samples will be restricted to a short range. Therefore, the characteristics of LRD will diminish. Although $\log(Time) = 2.5$ can be set as a threshold to estimate the Hurst parameter, it might change from case to case. How to find an appropriate threshold is a difficult issue.

When comparing the diagrams on both sides no matter which algorithm is used, the higher Hurst parameter will cause a more sharpened bend on the tail. The sharpened bends lead to a higher correlation with previous samples. From this point of view if the scattering dots form a straight curve, the analytical traces have less correlation with the samples themselves. Thus, the RMD algorithm produces the worst samples. This conclusion coincides with the analysis of the RMD algorithm in Sec. 4.1.4. When the sample is determined, it will be fixed and will not change throughout the simulation. This result is that the first half of the sample traces are independent of the second half of the sample traces. If the Hurst parameter is low, the correlation of the samples is still small. The drawbacks of the RMD algorithm is not significant, however, the disadvantages of the RMD algorithm will be magnified when the Hurst parameter increases.

From above discussion it can be concluded that the VT plot is not a suitable estimator for estimating the self-similar processes or the LRD processes. The estimator cannot reflect the strong correlation of the sample paths when the Hurst parameter is high. Although the estimator uses the exponent $2H - 2$ to capture the characteristics of the strong dependence, it does not consider variances of the aggregated length of the samples that will significantly change as the Hurst parameter increases.



(a) dFGN (H= 0.5)

(b) dFGN (H= 0.9)

(c) RMD (H= 0.5)

(d) RMD (H= 0.9)

(e) WM (H= 0.5)

(f) WM (H= 0.9)

Figure 4.44: Comparisons of the graphical results of the dFGN algorithm, the RMD algorithm, and the WM function with the VT plot.

The results for the R/S statistic always have large confidence intervals, although their mean values do somewhat reflect the actual Hurst values. Therefore, it is only suitable for use as a reference.

Although the results of the statistical analysis indicate that Whittle's MLE is a good estimator, it is still hard to use in practice. Aggregation levels have been proposed to solve one of the problems in statistical robustness. However, the larger the aggregation level is, the wider the confidence interval becomes. It is very difficult to use this to estimate the Hurst parameter in an on-line operation. Using a long memory to store a large amount of samples can extract more accurate Hurst values. However, it is almost impossible to have such a large memory for a single parameter. Therefore, Whittle's MLE is a good indicator of the quality of the FBM and FGN samples.

4.6 Summary

In this chapter six algorithms for generating FBM or FGN have been introduced. The dFGN algorithm is derived from the discrete version of the definition of FBM. This algorithm generates FGN by using massive ordinary Gaussian noise samples. The fFGN algorithm produces samples of FGN through Markov-Gaussian processes. The RMD algorithm simply generates the trace by recursively interpolating the mid points with a displacement. Due to the drawbacks of the RMD algorithm, the SRA algorithm was proposed to eliminate the independence between the first and last half of sample traces. Both algorithms generate traces of FBM. The WM function is an algorithm which generates traces in the frequency domain before transforming the samples to the time domain. The final result is a trace of FBM. Finally, Whittle's approximate approach was derived from the development of Whittle's MLE. It also generates the samples in the frequency domain with exponential and uniform distributions and then transforms them into the time domain. The trace in the time domain is a time series of FGN.

When considering computational effort it has been shown that the RMD algorithm is the fastest algorithm, while the dFGN algorithm is the most time-consuming. Although the SRA algorithm has the same computational complexity as the RMD algorithm, it cannot recursively generate the samples. For the WM function and Whittle's approximate approach the major computational effort is due to the inverse fast Fourier transform. The complexity of the fFGN algorithm is $O(N \log N)$, the same as the inverse fast Fourier transform. However, to generate a sample requires more operations with the Markov-Gaussian process.

After generating sample paths, three estimators have been used to estimate their Hurst parameters. In addition, a visual assessment tool, the Q-Q plot, has been used to visually assess the goodness-of-fit of the algorithms. The Q-Q plots show that the larger the Hurst parameter is, the worse the algorithms fit the normal distributions. However, for the dFGN algorithm and the WM function their Q-Q plots are indistinguishable to the eye from the normal distribution. The Q-Q plots of the SRA algorithm are slightly better than those of the fFGN algorithm and much better than those of the RMD algorithm. The Q-Q plots of Whittle's approximate approach are very hard to compare with the other algorithms because of their weird tails.

Using the R/S statistic, the VT plot, and Whittle's MLE to estimate self-similar time series, the comparative results have proved to be varied. Comparing the results using the R/S statistic, all estimated

ones tended to be higher than the expected value when the Hurst parameter is near 0.5. In contrast, the results tend to be lower than the Hurst parameter, as it tends towards 1. Using the R/S statistic, the dFGN algorithm exhibits the best result, while the fFGN algorithm is slightly worse because of the wider confidence intervals, Whittle's approximate approach is the worst, with the RMD algorithm only slightly better than the WM function. Using the VT plot the RMD algorithm exhibits the best result, while the WM function and the dFGN algorithm are the worst ones. However, when Whittle's MLE algorithm is applied, the dFGN algorithm and the WM function are the best two algorithms.

Assessment with the VT plot suggests that the dFGN algorithm is one of the worst while in contrast the assessment results using Whittle's MLE and the R/S statistic suggest that the dFGN algorithm is the best algorithm. Such a difference leads to an investigation into the effect of changing parameters for the dFGN algorithm. This work suggests that usage of Whittle's MLE could help to estimate the Hurst value more accurately. When examining the diagram of the VT plot, the image of the scattering dots forms a curve with a nonlinear tail as the Hurst parameter increases. With such a sharpened nonlinear tail the estimator tries to fit a straight line around the scattering dots. Thus, the estimator is misrepresenting the Hurst values. The results using the R/S statistic are always underestimated when the Hurst parameter tends to 1 and overestimated when the Hurst parameter nears 0.5. In addition, the results are highly dependent on the sample traces. Therefore, it can only be used as a reference. Whittle's MLE is a good estimator that can fully reflect the real Hurst value. However, it is not suitable for practical operations.

Although the dFGN algorithm has the best results except for the VT plot, its massive computational time will restrict its widespread use. Furthermore, it also has the advantage that it is the only algorithm that can generate samples on-the-fly. For other algorithms the number of samples must be decided before simulation. Although researchers can cascade several different traces into a whole trace, the sub-traces are independent of each other. The WM function is a compromise as its results are better than those of the other algorithms (except for the VT plot) and its simulation time is significantly shorter than that of the dFGN algorithm. However, the major disadvantage is that the algorithm must decide the number of samples in advance. The RMD and SRA algorithms are two algorithms which can be used for generating self-similar samples. Their simulation speed is the most attractive issue. Although the results and computational speed of the SRA algorithm are slightly worse than the RMD algorithm, it overcomes the independent disadvantage of the RMD algorithm. For simulation, the fFGN algorithm and Whittle's approximate approach have been shown to be of far less value.

The RMD and SRA algorithms have computational efficiency but cannot generate samples on-the-fly. In terms of quality, the sample traces generated by them cannot accurately reflect the Hurst parameter. The WM function is also a fast algorithm which generates good quality samples, especially when the Hurst parameter tends to 1, but the size of samples must be decided before simulation. The dFGN algorithm can generate high quality samples on-the-fly, but its simulation time is the greatest. In the next chapter several hybrid algorithms are proposed to retain the useful properties, to shorten the simulation time, and hopefully to improve the relative quality of samples.

Chapter 5

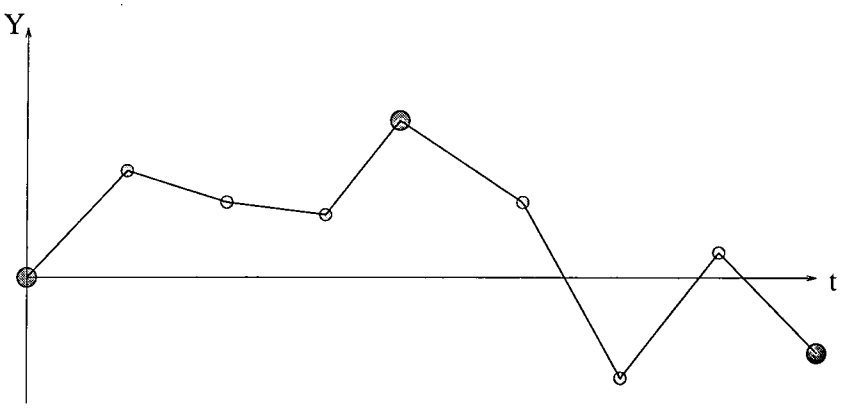
Hybrid Approaches for Generating Fractional Brownian Motion

From the previous chapter, it is known that the dFGN algorithm is the only one which can generate samples on-the-fly. The dFGN and the WM function are the best in terms of accuracy while the RMD, the SRA algorithms and the WM function are the top three in term of efficiency. Since the RMD and the SRA algorithms iteratively bisect the entire sample traces from two known endpoints, they can easily be incorporated into the dFGN algorithms. The WM function also can be incorporated with the dFGN algorithm. These characteristics lead to three hybrid approaches, which can hopefully create new algorithms with both improved accuracy and efficiency, and the ability to operate 'on-the-fly'. These new algorithms are developed in Sec. 5.1, Sec. 5.3, and Sec. 5.5, respectively, while the evaluations of their sample traces are carried out in Sec. 5.2, Sec. 5.4, and Sec. 5.6.

5.1 The dFGN Endpoints with Interpolated RMD Subtraces

The dFGN algorithm is directly derived from the definition of fractional Brownian motion. It is a moving-average procedure that aggregates $M \times n$ Gaussian random variables with a kernel function to generate one sample of fractional Gaussian noise. The kernel function is decided by the Hurst parameter. The higher the Hurst parameter, the stronger the samples correlate. Therefore, the larger the range M and the more detailed the substates n, the greater the increase in accuracy of the samples to reflect the properties of LRD. However, the computational time increases rapidly.

The idea of the RMD method is to recursively subdivide the intervals and construct the values of the process at the midpoint using the values at the endpoints. If a process Y_t is to be computed n times between 0 and 1, then the computation starts by setting $Y_0 = 0$ and sets Y_1 as a sample of a Gaussian random variable. Next, $Y_{\frac{1}{2}}$ is constructed as the average of Y_0 and Y_1, i.e., $\frac{1}{2}(Y_0 + Y_1)$ plus a displacement. Executing this procedure recursively n times, a sample trace with 2^n points is obtained. If Y_t were truly an FBM process, then the midpoint displacement $Y_{\frac{a+b}{2}} - (Y_a + Y_b)/2$ for the interval $[a,b]$ would be independent of the increment $Y_b - Y_a$ over the whole interval. The displacement would be a Gaussian random variable with mean zero and a variance which follows the scaling property.



● samples generated by dFGN

○ samples generated by RMD

Figure 5.1: A two-stage of the dFGN+RMD algorithm with 2 levels of depth.

The performance of the RMD algorithm is rather attractive, but has low accuracy in capturing the target Hurst values. In contrast, the dFGN algorithm can be close to the Hurst value, but its inefficiency prevents its widespread use. Since the RMD algorithm interpolates the trace between two endpoints, it is reasonable to generate a subtrace between two endpoints given by the dFGN algorithm.

Without loss of generality let the position of the origin at time 0 be 0. Generating increments using the dFGN algorithm, say X_1, the first two end FBM samples, Y_0 and Y_1, are obtained. Then, applying the RMD algorithm with pre-given levels of depth, n, the subtrace $\{Y_i\}$, for $i = \frac{1}{2^n}, \frac{2}{2^n}, \cdots, 1$, is generated. By using the dFGN algorithm again X_2 is produced. The third position of the FBM process is $Y_2 = X_1 + X_2$. The RMD algorithm can make use of Y_1 and Y_2 to generate another subtrace $\{Y_i\}$, for $i = 1 + \frac{1}{2^n}, 1 + \frac{2}{2^n}, \cdots, 2$. Repeating this procedure, every iteration generates 2^n samples of the FBM sample path. The procedure is depicted in Fig. 5.1.

In accordance with the hybrid approach, Fig. 5.2 illustrates FBM and corresponding FGN processes produced by the dFGN, RMD, and hybrid algorithms with $H = 0.7$. The results from the hybrid algorithm is for 5 levels of depth; that is, for every iteration a subtrace with 32 samples is generated. The dFGN and hybrid algorithms evaluate their trace with $n = 10$ and $M = 700$, i.e., the range covers 700 time units and every time unit is divided into 10 subintervals. On the left-hand column three traces of FBM are illustrated while their second-order self-similar processes, the traces of FGN, are depicted in the right-hand column.

The drawback of the massive computational effort required for the dFGN algorithm is alleviated in the combination of the dFGN and RMD algorithms because of the speed of the RMD algorithm. The sample size of this hybrid algorithm is constrained by the RMD algorithm because its number of samples must be a power of 2. The average computational time for the hybrid approach with five different levels of depth and the dFGN and RMD algorithms are depicted in Fig. 5.3. Between the results of the dFGN and RMD algorithm five different hybrid approaches are indicated by 'Hybrid-1', 'Hybrid-7', 'Hybrid-31', 'Hybrid-255', and 'Hybrid-1023'. The symbol of 'Hybrid-x' means that between two endpoints there are x samples generated by the RMD algorithm, i.e., $x + 1 = 2^n$, where n is the levels of depth.

The hybrid approach significantly reduces the computational effort. The computational time is ob-

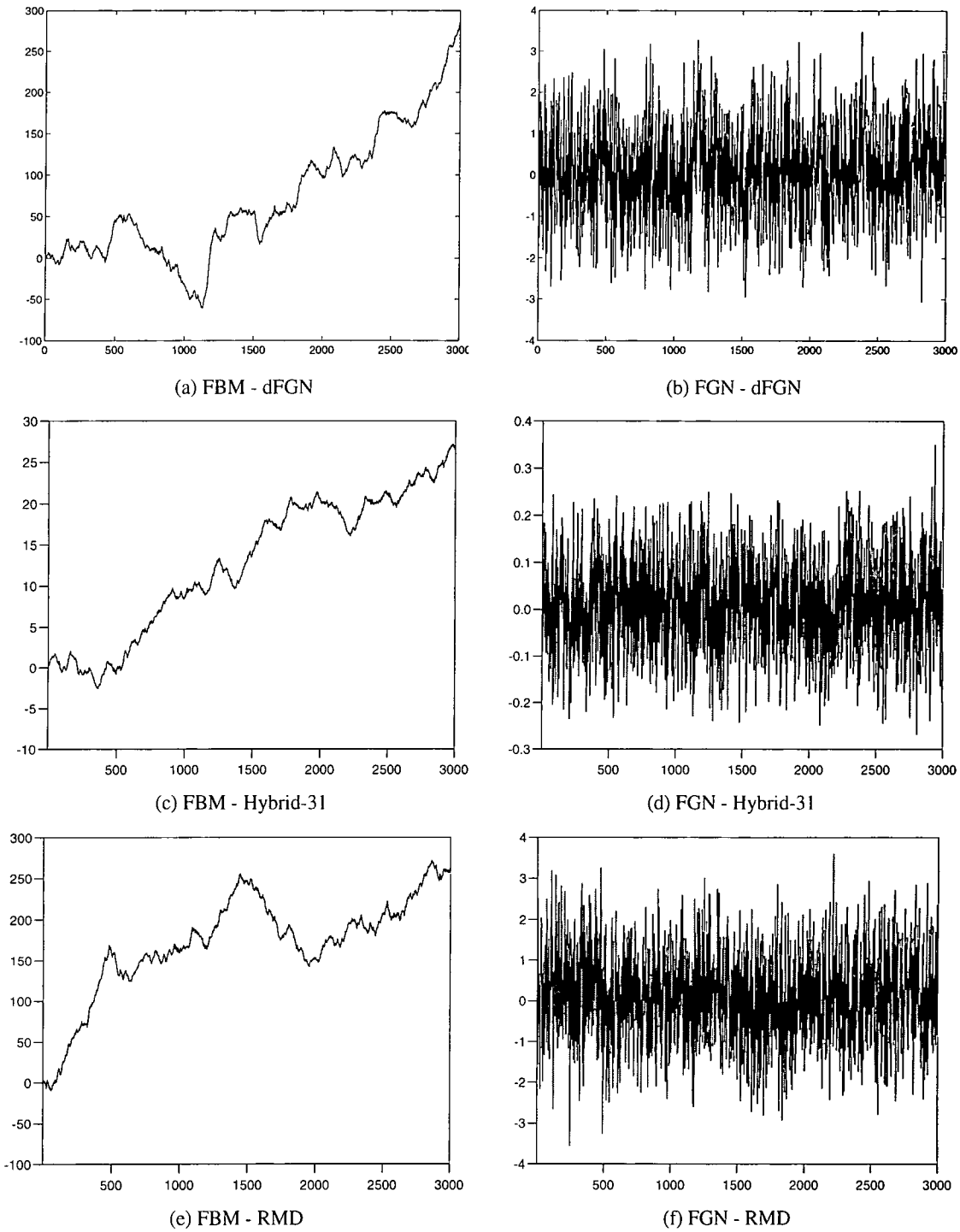


Figure 5.2: FBM and FGN traces of the dFGN, RMD and dFGN+RMD algorithms with $H = 0.7$.

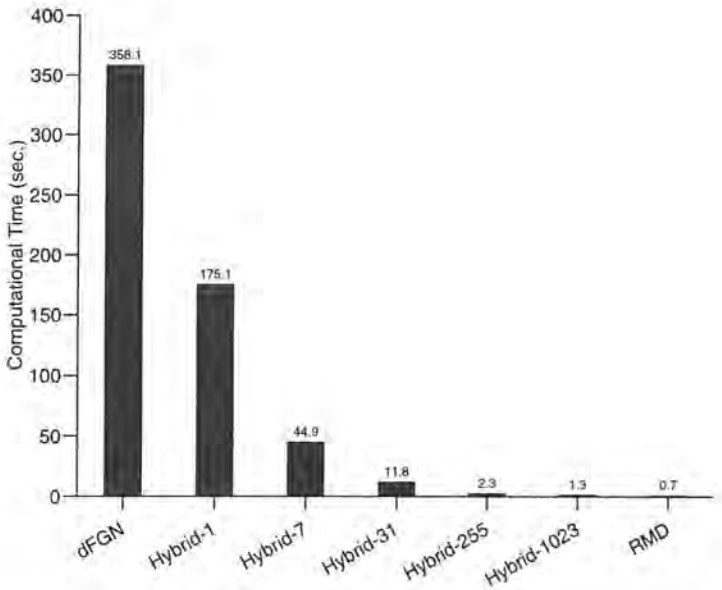


Figure 5.3: Mean computational time of the dFGN+RMD algorithm to generate 32768 sample points.

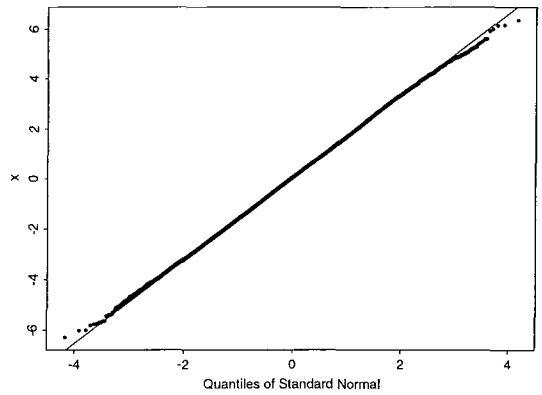
tained by more than 180 simulations for each algorithm. The dFGN algorithm takes 360 seconds more to generate 32768 samples while the RMD algorithm takes less than one second. For the dFGN+RMD algorithm with one level of depth, the samples generated by the dFGN and RMD algorithms are equally mixed. Its computational time is less one half that of the dFGN algorithm. The dFGN+RMD with 10 levels of depth is less than twice the computational effort of the RMD algorithm. The detailed average computational time of each algorithm with different Hurst parameter settings are compiled in Table D.1.

5.2 Statistical Analysis of the dFGN+RMD Algorithm

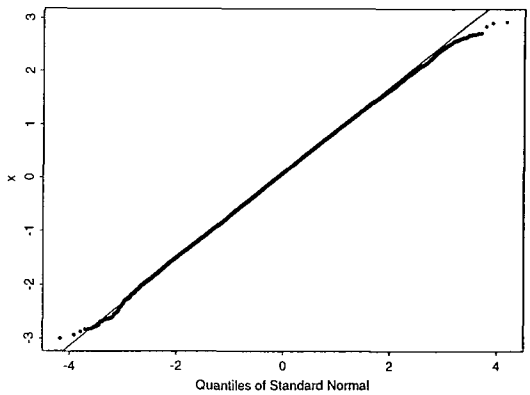
In Sec. 3.5, three estimators, the R/S statistic, the VT plot, and Whittle's approximate MLE, are introduced and used to examine the algorithms which are introduced in Sec. 4.1. Similarly, the hybrid approach is also examined by the three estimators. Although the VT plot is not an accurate estimator, it is popular and easy to apply. Therefore, it is still used as a reference to the accuracy of the hybrid approach. In addition to the estimators, the Q-Q plot is also used to examine the goodness-of-fit of the generated samples to normal distribution.

5.2.1 The Q-Q Plot

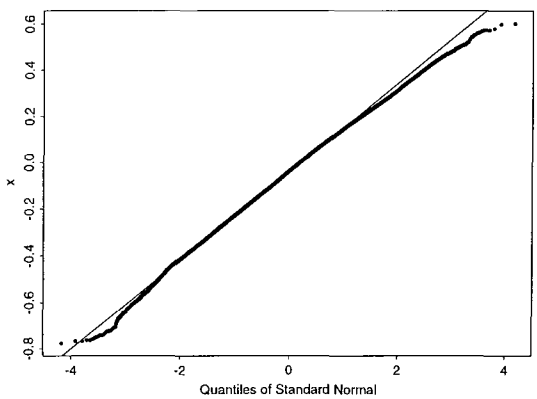
Although the hybrid approach combines two different algorithms, it still aggregates Gaussian random variables to generate samples. The higher the levels of depth are, the less the Gaussian samples are aggregated. In Fig. 5.4 the hybrid approach with six different levels of depth is depicted in order to evaluate the goodness-of-fit with normal distribution. The diagrams Fig. 5.4 (b), (c), (d), (e), (f), and (g) are depicted with 1, 3, 5, 7, 9, and 10 levels of depth. The first and last diagrams are the Q-Q plot of the dFGN and RMD algorithms. All of the diagrams show the sample traces with a selected random seed and the same Hurst value, $H = 0.99$.



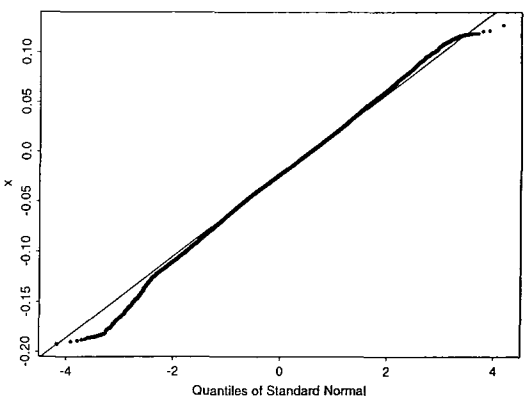
(a) dFGN



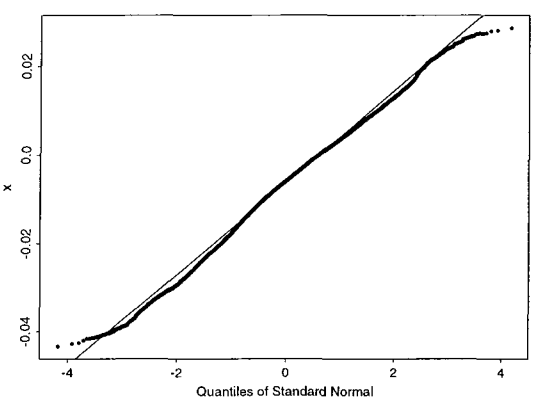
(b) Hybrid-1



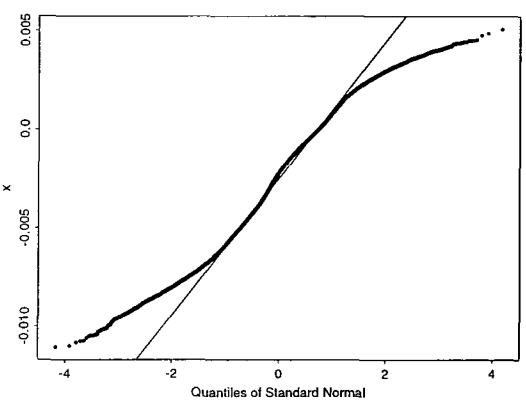
(c) Hybrid-7



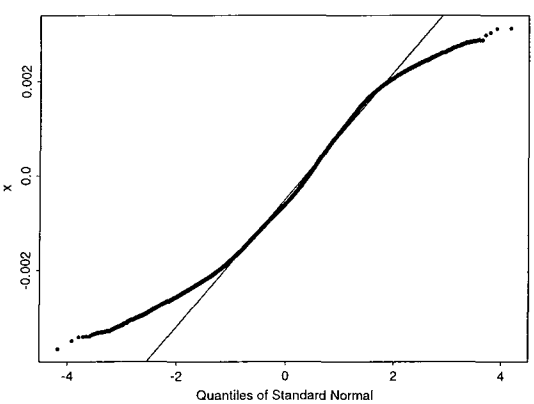
(d) Hybrid-31



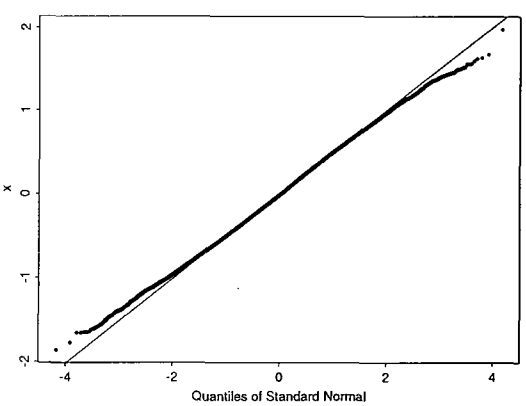
(e) Hybrid-127



(f) Hybrid-511



(g) Hybrid-1023



(h) RMD

Figure 5.4: The Q-Q plot of the sample traces generated by the dFGN, RMD and dFGN+RMD algorithms with $H = 0.99$.

Comparing the diagrams the dFGN algorithm exhibits the best result for goodness-of-fit. The Q-Q plot of the hybrid approach with 1 level of depth exhibits a better result than the RMD algorithm. The hybrid approach with 3 levels of depth and the RMD algorithm have almost the same quality while the hybrid approach with 5 levels of depth is slightly worse. The Q-Q plot of the hybrid approach with 7 levels of depth exhibits an acceptable quality. When the levels of depth in the hybrid approach increased to 9 and 10, a serious distortion of the normal distribution is exhibited. Thus, the Q-Q plots suggest that the usage of the hybrid approach is best with no more than 7 levels of depth. In turn, this means that the subtraces, generated by the RMD algorithm, between two endpoints generated by the dFGN algorithm, do not require more than 127 samples.

5.2.2 The R/S Statistic

The R/S analysis is a heuristic graphical approach that can be used to estimate the Hurst parameter of an FBM process. The detailed discussion of the R/S statistic is found in Sec. 3.2 and Sec. 3.5. The estimated Hurst value is directly extracted from the slope of the regression line of the R/S values.

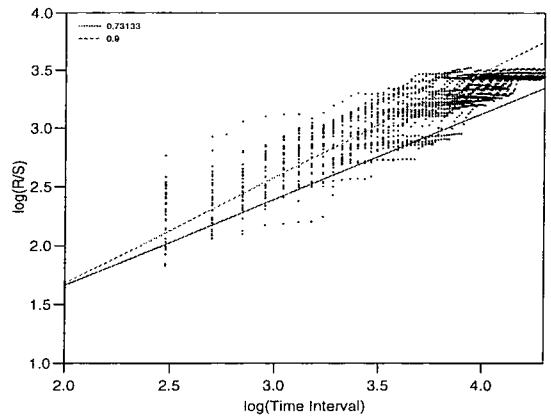
The results using the R/S statistic are demonstrated in Fig. 5.5. The solid line is the regression line and its slope is the Hurst value, the dotted line in each diagram is the reference line that represents the target Hurst parameter. All the sample traces that have been selected for these diagrams use $H = 0.9$ with the same random seed is the same as for the sample traces generated by the Q-Q plots. The results using the R/S statistic exhibit chaotic behavior. It is very difficult to extract a simple rule to demonstrate the variety of the results. The RMD algorithm is closest to the expected Hurst parameter almost the same as the simulation result. However, the hybrid approach with 10 levels of depth, which was the closest case with the RMD algorithm in Fig. 5.5 (g), does not exhibit a significantly good result. The result in Fig. 5.5 (f), even exceeds the theoretical limitation $H = 1.0$. The result for the hybrid-1 combination is more for either the dFGN or RMD algorithms alone.

The chaotic results using the R/S statistic make it difficult to decide which case is better. One possible explanation for the spread of results is the selection of the random seed values. From the study in Chapter 4 the R/S statistic is highly dependent on the selection of random seeds. Although the different combinations of the hybrid approach are generated from the same random seed, the mixing rates permute the sequences of the Gaussian random variables. Thus, it is almost impossible to predict the trend of the results using the R/S statistic.

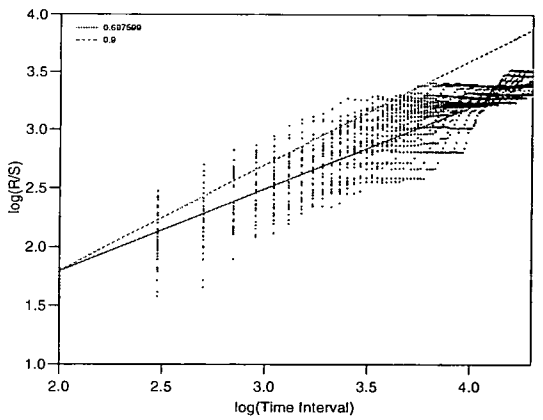
5.2.3 The Variance-Time Plot

The VT plot is another heuristic estimator in which a regression line is extracted from the plot of logarithmic variance versus logarithmic time interval. The slope of the regression line is equal to $2 - 2H$. The estimator is derived from the property of FGN by which the variance of two different samples of FGN follows the scaling property. The detailed derivation of the estimator is stated in Sec. 3.5.

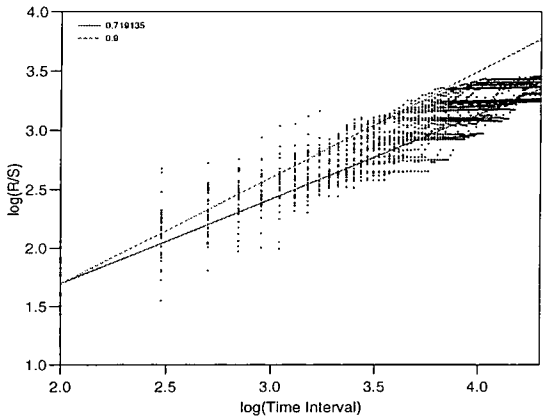
The results using the VT plot are illustrated in Fig. 5.6. The diagrams are illustrated being the same sample traces as the Q-Q plots. All the sample traces are generated with the same random seed and the highest degree of self-similarity in our simulation, $H = 0.99$. As with previous figures, the first and last



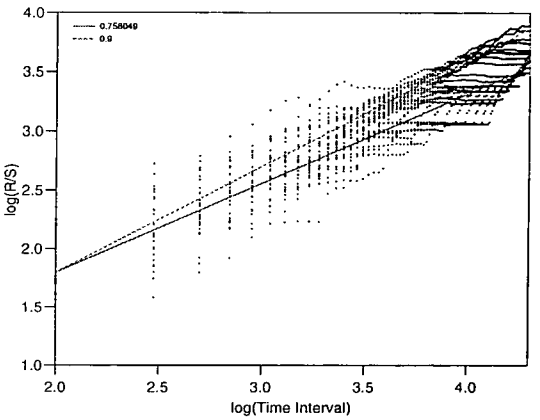
(a) dFGN



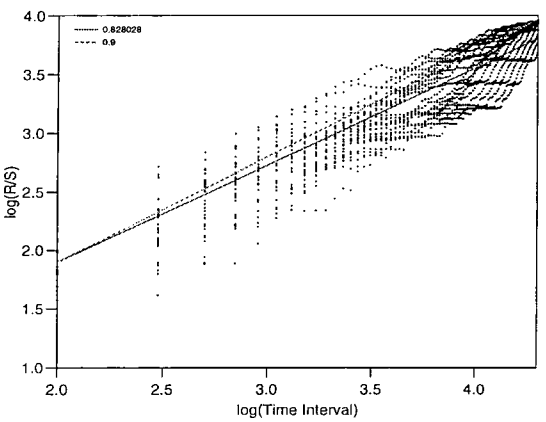
(b) Hybrid-1



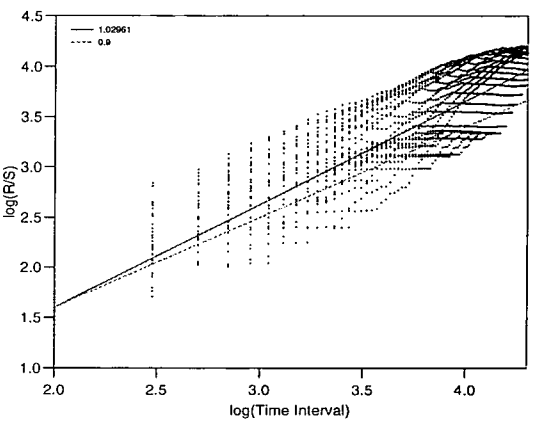
(c) Hybrid-7



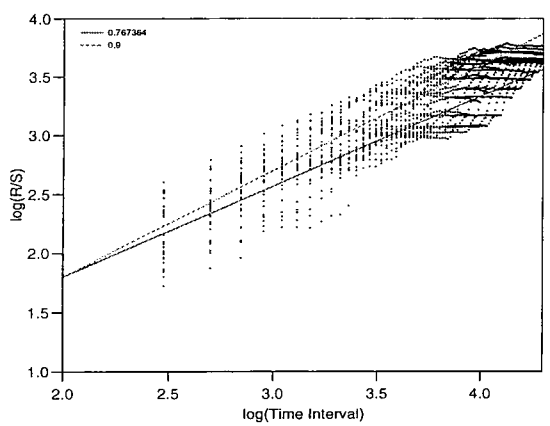
(d) Hybrid-31



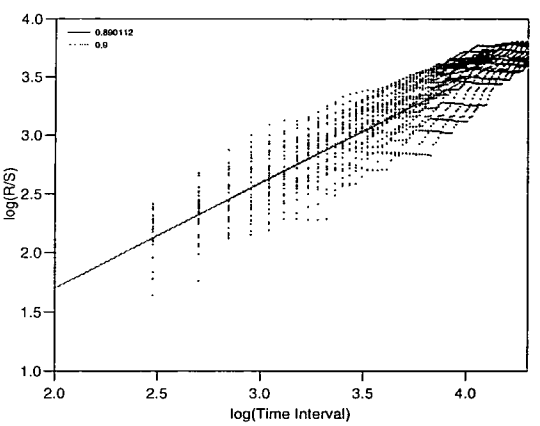
(e) Hybrid-127



(f) Hybrid-511

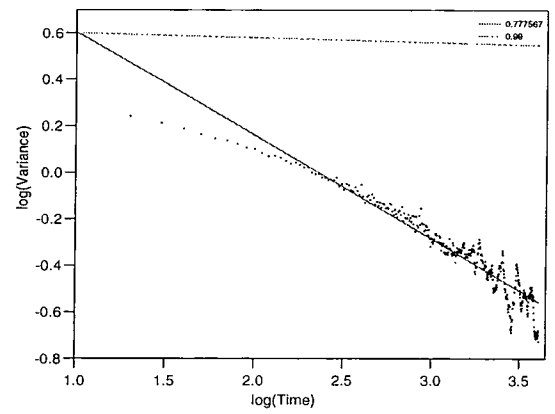


(g) Hybrid-1023

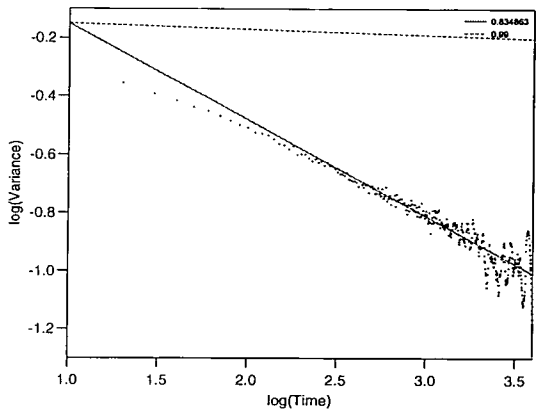


(h) RMD

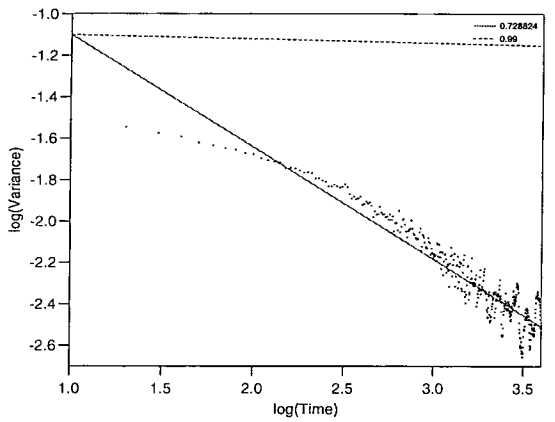
Figure 5.5: The pox plots of R/S statistic of the sample traces generated by the dFGN, RMD, and dFGN+RMD algorithms with $H = 0.9$



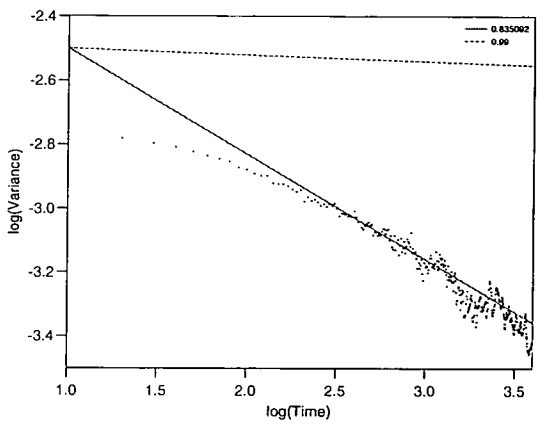
(a) dFGN



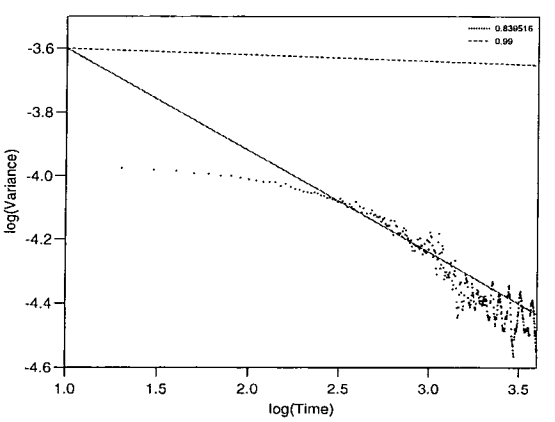
(b) Hybrid-1



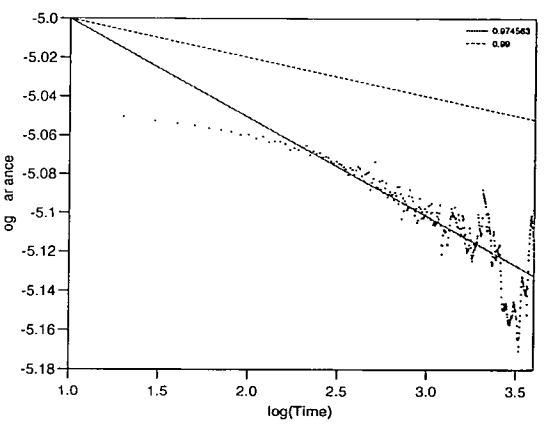
(c) Hybrid-7



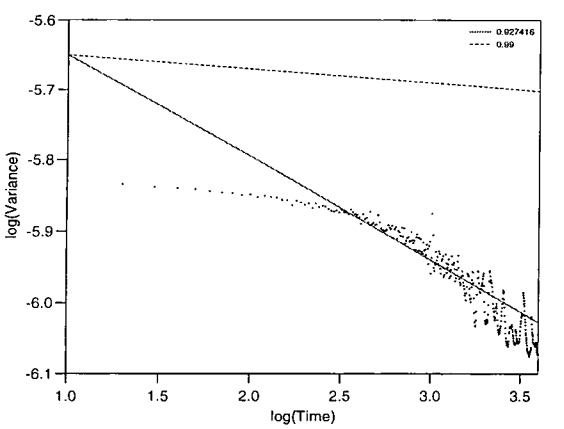
(d) Hybrid-31



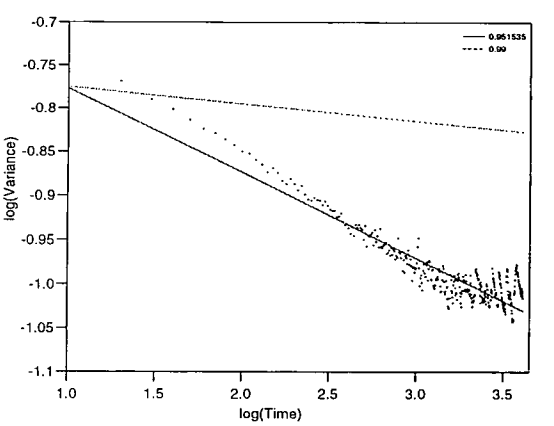
(e) Hybrid-127



(f) Hybrid-511



(g) Hybrid-1023



(h) RMD

Figure 5.6: The VT plots of the sample traces generated by the dFGN, RMD and dFGN+RMD algorithms with $H = 0.99$.

diagrams are generated by the dFGN and RMD algorithms. The diagrams with the hybrid approach are procured with $1, 3, 5, 7, 9$, and 10 levels of depth. In these diagrams, the best result is neither the case of the dFGN algorithm nor that of the RMD algorithm. The hybrid approach with 9 levels of depth exhibits the best result. However, when the level of depth increases or decreases one step, the value drops sharply away from the target value. There is only one case, the level of depth of 3, where the value is worse than the dFGN algorithm. A number of the higher levels of depth are even better than that of the RMD algorithm. However, the results still vary. Furthermore, the VT plot is also highly dependent on the selection of random seed. Therefore, it is difficult to select which case is more accurate.

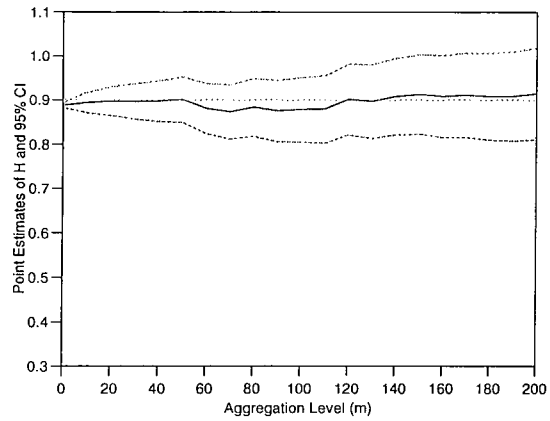
From Fig. 5.6 (e)-(h), the dots suggest a polynomial regression rather than linear regression. In property L'-4, Sec. 3.3, the approximation of the autocorrelation is proportional to the lag k with a simple power law. With theoretical approximation, the factor β is equal to $2-2H$. In practice, taking infinite samples to produce the variance-time plot is impossible. The simplification and truncation lead to the approximation error and produce the non-linear regression results. Using nonlinear curves (polynomial regression) to fit the dots would lead to great difficulty in finding the relationship between the curve and the Hurst parameter. Therefore, it is impractical to use nonlinear curves to fit the results of the variance-time plot.

5.2.4 Whittle's Approximate MLE

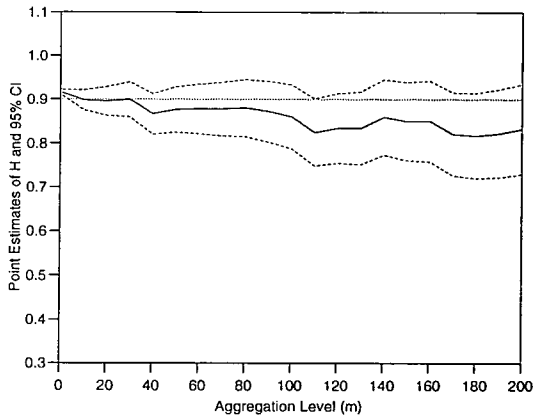
While the R/S and VT tests focus on the time domain, the Whittle's approximate estimator examines the properties in the frequency domain. Whittle's MLE extracts the Hurst value by minimizing a log-likelihood function. This estimator is derived through a rigorous mathematical derivation rather than a heuristic approach. The detailed discussion of Whittle's MLE is stated in Sec. 3.5.

The results using Whittle's MLE with aggregation levels is depicted in Fig. 5.7. As with the previous figures for the other estimators the first and last diagrams are the results of the dFGN and RMD algorithms. Between them are the sample traces generated by the different levels of depth. The sample traces used are the same as those analyzed by the VT plots, generated with a selected random seed and the Hurst parameter , $H = 0.9$. The initial result of the hybrid approach in Fig. 5.7 (b) is even higher than the target Hurst value $H = 0.9$. However, the mean value quickly drops below the target value when the aggregation levels are greater than 30. In Fig. 5.7 (c) the hybrid approach with 3 levels of depth is illustrated. The values with different aggregation levels are almost along the reference line, the target value $H = 0.9$. When the levels of depth increase, the initial values gradually drop below the reference value. When the levels of depth are greater than 5, the values are mostly lower than those of the RMD algorithm except for the initial values. Thus, these results suggest that the hybrid approach is preferred when the level of depth is less than 5.

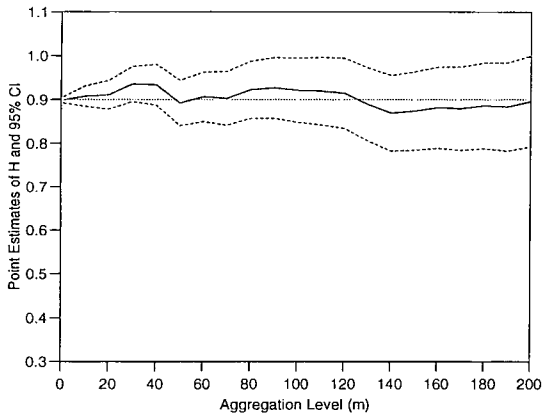
The results with a fixed Hurst parameter above are followed by a study of the relationship of the level of depth at different Hurst parameters. The first row of diagrams in Fig. 5.8 are the values of sample traces generated with $H = 0.5$. Similarly, the other rows of diagrams depict the results obtained from different Hurst parameters, $H = 0.7$, 0.9, and 0.99. The right-hand side of the diagram illustrates the results of the Hurst approach with 1 - 5 levels of depth while the other side depictes the results obtained



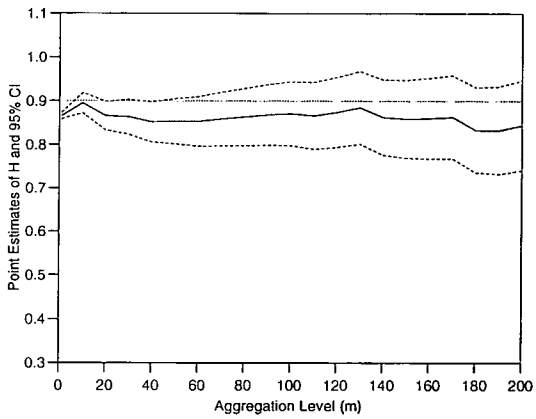
(a) dFGN



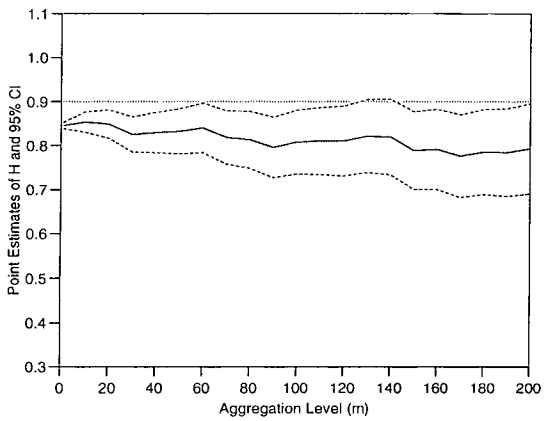
(b) Hybrid-1



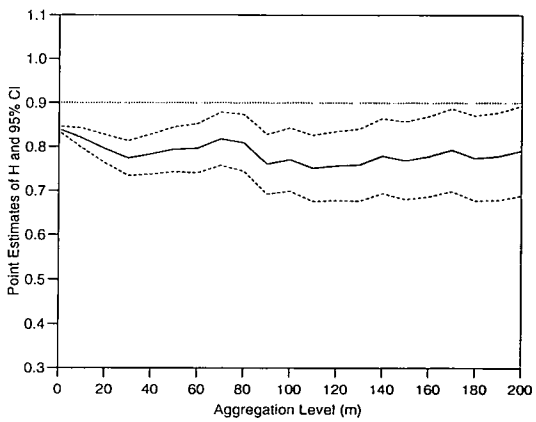
(c) Hybrid-7



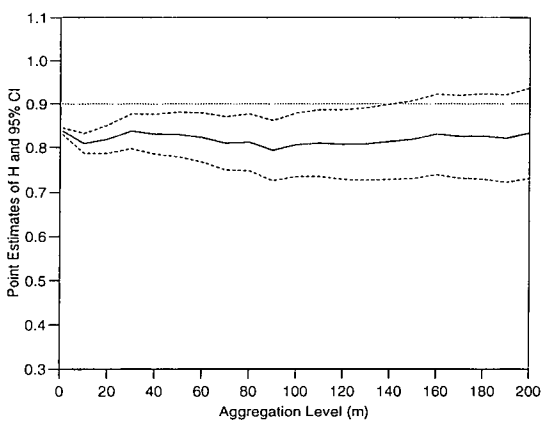
(d) Hybrid-31



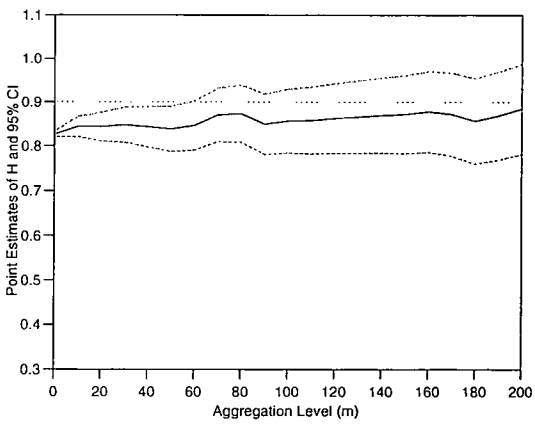
(e) Hybrid-127



(f) Hybrid-511

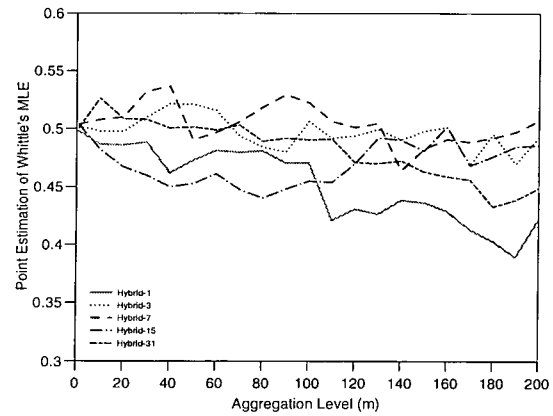


(g) Hybrid-1023

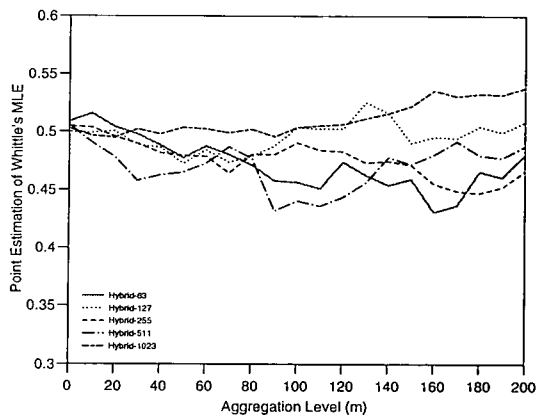


(h) RMD

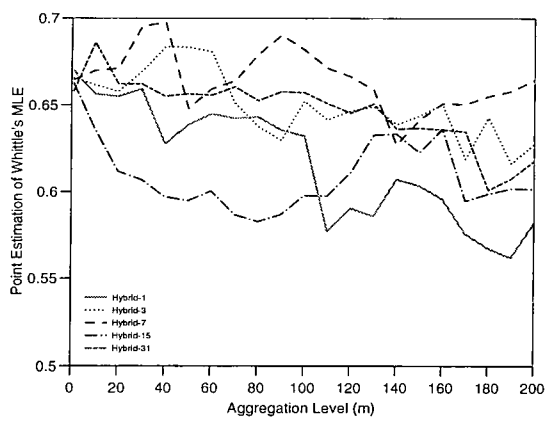
Figure 5.7: The Hurst values of the sample traces generated by the dFGN, RMD, and dFGN+RMD algorithms with Whittle's MLE.



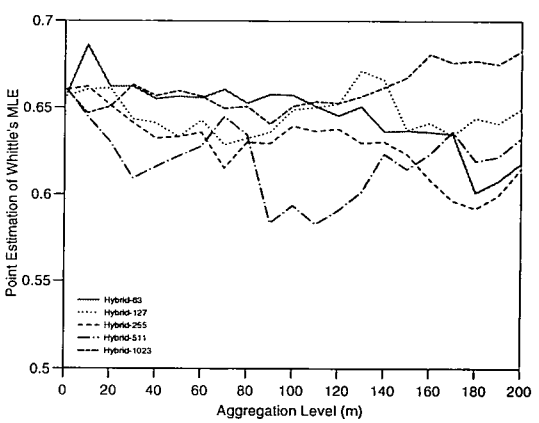
(a) H = 0.5



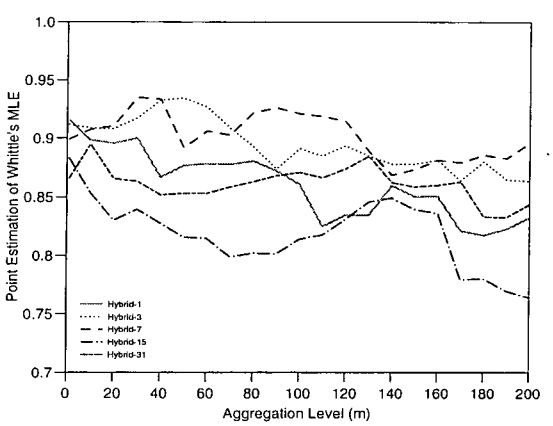
(b) H = 0.5



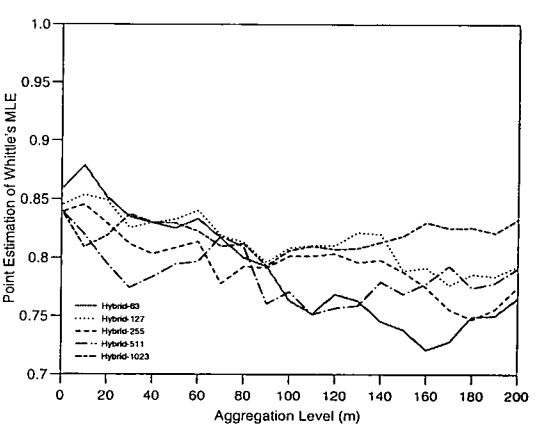
(c) H = 0.7



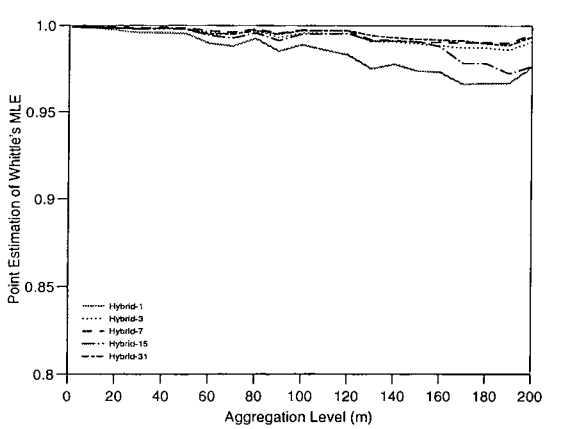
(d) H = 0.7



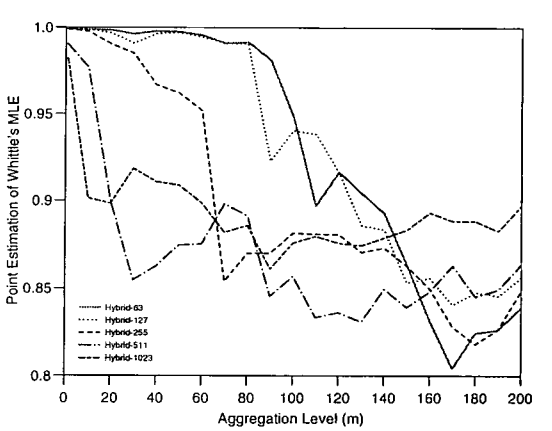
(e) H = 0.9



(f) H = 0.9



(g) H = 0.99



(h) H = 0.99

Figure 5.8: Analytical results of the dFGN+RMD algorithm by Whittle's MLE with different aggregation levels.

from the sample traces with 6 - 10 levels of depth. All the sample traces are generated with a selected random seed.

In Fig. 5.8 (a) and (b) the curves portray a chaotic deployment over the range from 0.38 to 0.54. Surprisingly, the worst case is the hybrid approach with 1 level of depth. In this case, not only is the simulation time the longest but also, the results with different aggregation levels are not as accurate as expected. When the aggregation level increases, the values drop sharply. For other cases, the results fluctuate without any significant increasing or decreasing trend. However, the majority of the results in these two diagrams are lower than the target Hurst parameter. When examining the whole traces as one aggregation level, the curves in Fig. 5.8 (a) are around $H = 0.5$ while the curves in Fig. 5.8 (b) are lower than the target value.

On the second row of Fig. 5.8 the result using Whittle's MLE with aggregation levels are illustrated for $H = 0.7$. In the left hand diagram the results of the hybrid approach with one level of depth still exhibit the worst case when the aggregation levels are greater than 100. All the curves in Fig. 5.8 (c) and (d) are all less than $H = 0.7$, the target value. For $H = 0.9$ in Fig. 5.8 (e) and (f), the curves in these two diagrams form two different groups. When the level of depth is less than 4, some of the results attain the target values. When the level of depth is larger than or equal to 4, the values are never higher than $H = 0.9$. The curves in diagram (f) illustrate that all the results are significantly inadequate.

In the case of $H = 0.99$ the curves in Fig. 5.8 (g) are much neater than the other diagrams. All curves approach the target value, $H = 0.99$, significantly well when the aggregation level is less than 50. However, comparing these curves in this diagram the 'Hybrid-1' is still the worst case of the five. When examining the last diagram the curves cover a much wider range. All the curves in Fig. 5.8 (h) exhibit a similar phenomenon in that all of them show a sharp drop. For the 'Hybrid-63' and 'Hybrid-127' the sharp drop occurs when aggregation level is set as 80. For 'Hybrid-255' it occurs at 40. For 'Hybrid-511' the drop happens when aggregation level is 10. When the level of depth is set as 10, i.e., 'Hybrid-1023', the curve drops immediately. Actually, when examining diagram (f) the curves also slightly drop but not as significantly.

This situation might be caused by the inaccuracy of the RMD algorithm. When the level of depth increases, the RMD algorithm dominates the hybrid sample traces. When the Hurst parameter is still low, the result from the RMD algorithm are still acceptable. However, when the Hurst value increases, the results from the RMD algorithm are highly inaccurate. While the aggregation level is small, the samples generated by dFGN algorithm still retain their effect. However, as the subtraces become shorter, so the samples generated by the dFGN algorithm become fewer. For instance, when the level of depth is 10 and the aggregation level is 10, every subtrace contains about 3277 samples. In each subtrace only three samples are generated by the dFGN algorithms. Although the same proportion of samples are contained in the whole sequence, the samples generated by the dFGN algorithm contain only 3 strongly correlated samples in each sequence. Therefore, it indicates that the hybrid approach with the higher level of depth must be used very carefully.

In Fig. 5.8 the 'Hybrid-1' curves in (a) and (c) are the worst cases for $H = 0.5$ and 0.7. However, when a different random seed is used, the situation changes. Therefore, when the Hurst parameter is lower, the results are more sensitive to the random seeds. In addition, when examining the same level of

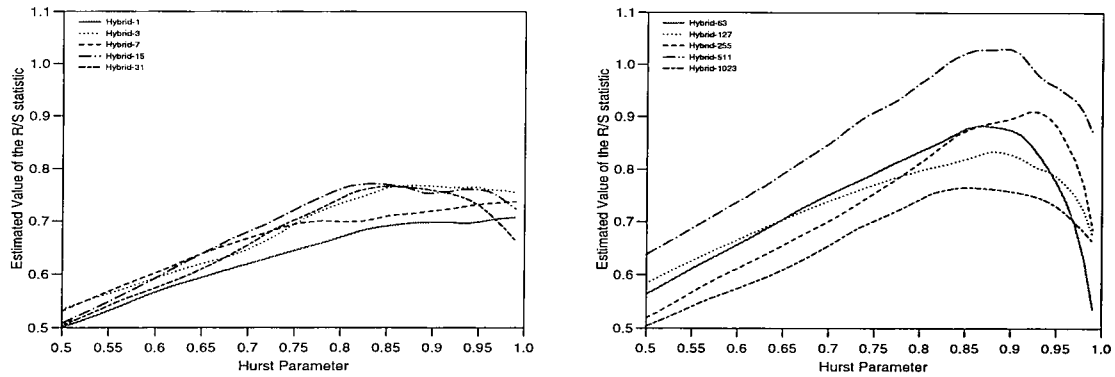


Figure 5.9: The estimated results using the R/S statistic with different Hurst parameters.

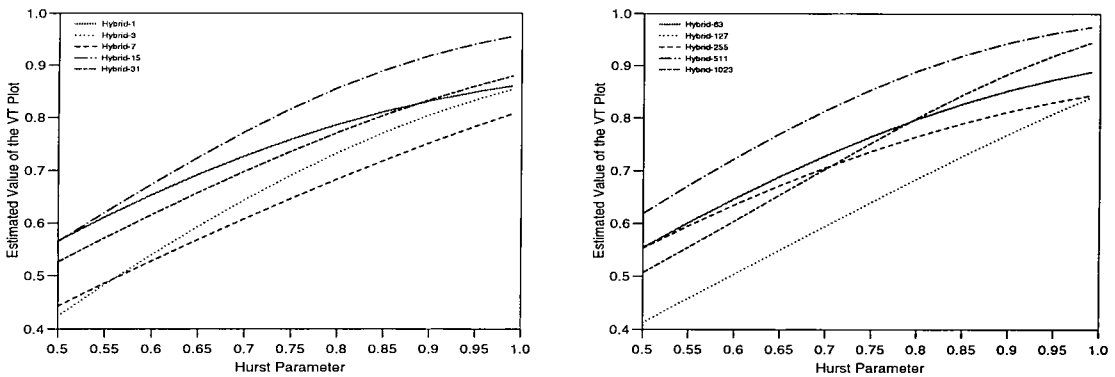


Figure 5.10: The estimated results using the VT plot with different Hurst parameters.

depth for all Hurst parameters, all the curves looks similar. This feature is especially true in the left-hand side column where all the curves have a similar fluctuating trend. The range of the $H = 0.7$ is larger than for those of $H = 0.5$ and 0.9. For $H = 0.99$ the trend is maintained but is minimized, and is not affected by the choice of random seed.

5.2.5 Comparisons of Different Combination Schemes

In above section the Q-Q plot, the R/S statistic, the VT plot were applied to examine the sample traces generated with a selected random seed. The effect of changing the Hurst parameters for the three estimators are illustrated in Fig. 5.9 - 5.11. In Fig. 5.9 the left-hand side of the diagram depictes the result of the hybrid approach with level of depth from 1 to 5 while the right-hand side illustrates the results with levels of depth from 6 to 10. Although the R/S statistic is a random-seed sensitive estimator, it can still be a reference for the Hurst parameter. The curves in the diagrams are not smooth, especially in the case of the curves extracted from the sample traces with higher levels of depth which show sharp turns between 0.85 to 0.95. The curves on the left-hand side of the diagram exhibit significantly lower results as the Hurst parameter increases, while the curves on the right of the diagram display chaotic behavior. The 'Hybrid-511' curve is even higher than unity when the Hurst parameter is around 0.85 and 0.9. The 'Hybrid-63' curve has a deep drop when the Hurst parameter is higher than 0.9. Although the curves in Fig. 5.9 are obtained from the same selected random seed, the curves obtained by other random seeds indicate a similar trend. Therefore, the diagrams suggest that the level of depth should not exceed 5.

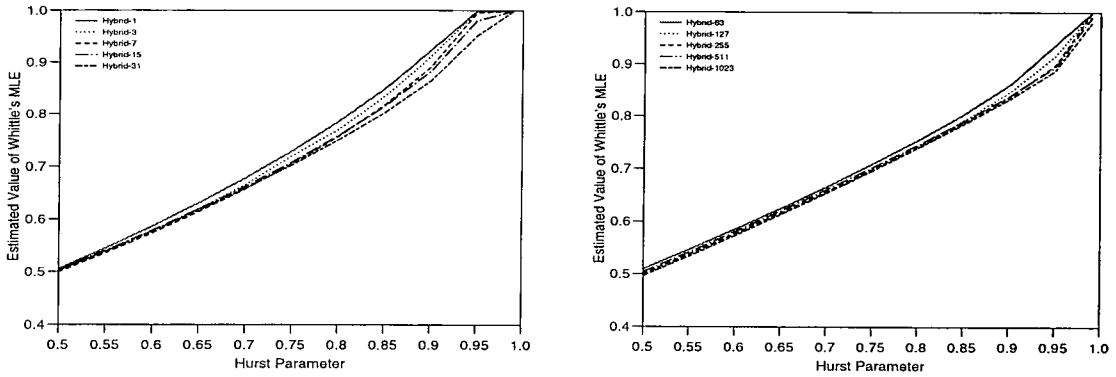


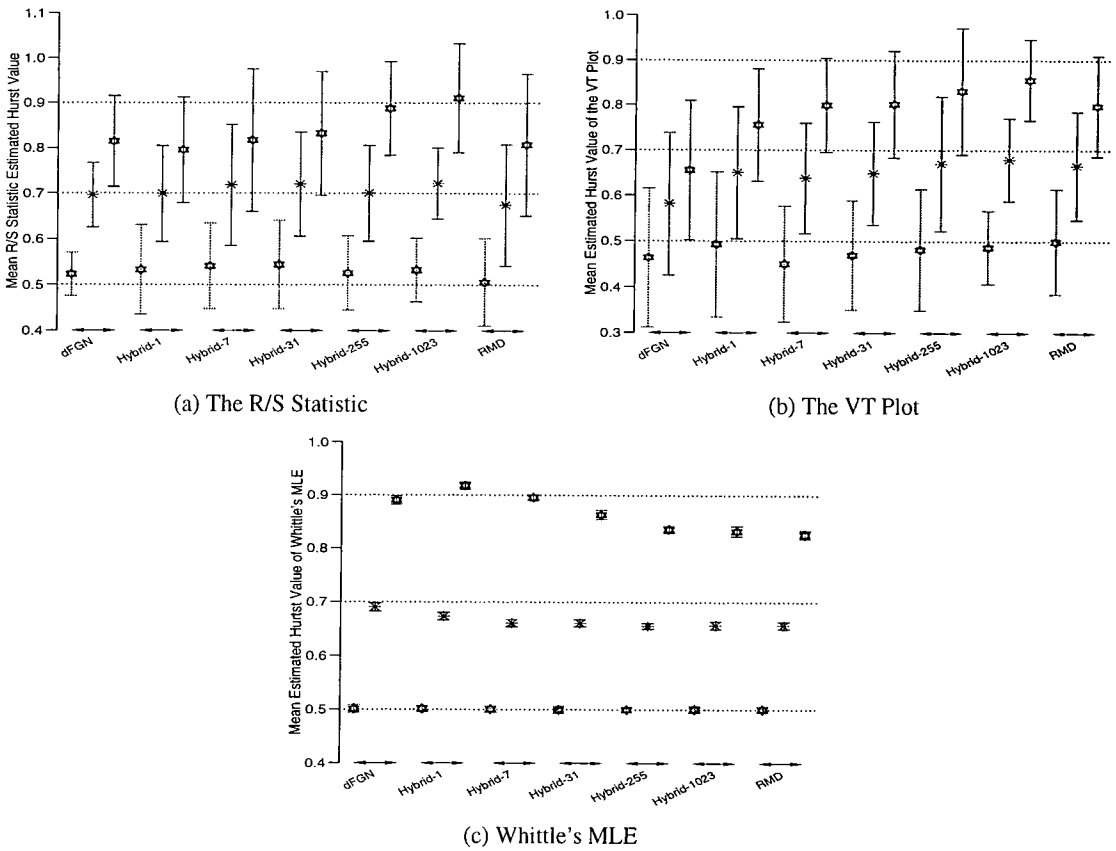
Figure 5.11: The estimated results using Whittle's approximate MLE with different Hurst parameters.

The diagrams of Fig. 5.10 illustrate the results using the VT plot. Unlike the curves of the R/S statistic, the curves produced by the VT plot are much smoother. However, the curves are still chaotic. The values for the different levels of depth are very different when the Hurst parameter is near 0.5. When the Hurst parameter approaches unity, none of the results exceed the target value.

The results using Whittle's MLE with changing Hurst parameters are depicted in Fig. 5.11. As with the previous two estimators the hybrid approach with 1 to 5 levels of depth is depicted in left-hand side of the diagram while the other levels of depth are illustrated on the right hand side. When the Hurst parameter is near 0.5, all the curves fit the target value very well. When the Hurst parameter increases, the curves gradually disperse. For the curves, 'Hybrid-1', 'Hybrid-3', and 'Hybrid-7', the values reach 0.99 when the Hurst parameter is more than 0.95. For 'Hybrid-15' the result is also greater than the target value when $H = 0.95$. The curves on the left hand side are under the target values between $H = 0.55$ to 0.85. The curves on the right hand side exhibit significantly worse results. The results are always lower than the target values between 0.6 to 0.95. All the curves in both diagrams are concave. Thus, to prevent under and over estimation, the 'Hybrid-31' and 'Hybrid-63' algorithms are the better choice with 'Hybrid-31' being slightly better than the 'Hybrid-63'.

The mean results of the hybrid algorithm with different levels of depth and the dFGN and RMD algorithms are illustrated in Fig. 5.12. The data for this diagram is compiled in Table D.7. The scattered symbols are the mean values of the estimated results. The bar lines on the scattered symbols are their 95% confidence intervals. The three dotted lines on these figures are the reference values, $H = 0.5, 0.7$, and 0.9.

The mean results using the R/S statistic are depicted in Fig. 5.12 (a). The bottom row contains the results obtained from the sample traces generated with $H = 0.5$. In this case the RMD algorithm still exhibits the best result. All the mean results of the hybrid approaches are worse than those of either the RMD or the dFGN algorithm. When $H = 0.7$ the hybrid algorithm with 1 and 8 levels of depth exhibit very accurate results. For $H = 0.9$ the hybrid approach with higher level of depth exhibits a more accurate mean value. When examining the confidence intervals the range slightly increases then decreases. Thus, the figure suggests that the higher level of depth, the better the results. However, this conclusion leads to a controversial argument because a higher level of depth will cause worse results. For example, when referring to Fig. 5.9 the hybrid approach with higher level of depth has a sharp turn near $H = 0.9$. In



(a) The R/S Statistic

(b) The VT Plot

(c) Whittle's MLE

Figure 5.12: Mean estimated Hurst values of the dFGN+RMD algorithm.

Fig. 5.12 (a), the mean results with highest Hurst parameter is 0.9. Therefore, the figures are misleading.

Fig. 5.12 (b) depicts the mean results using the VT plot, along with the confidence intervals for the dFGN, RMD, and hybrid algorithms. However, from the previous analysis, the VT plot has been found to be a poor estimator as it attempts to fit a straight line to nonlinear scattered dots. Thus, the results using the VT plot are used as a reference rather than a conclusion.

The mean results of the sample traces with Whittle's MLE are illustrated in Fig. 5.12 (c). All the mean values are on the reference line when $H = 0.5$. The mean results with $H = 0.7$ exhibit a gradually decreasing trend from the left hand side to right hand side, the mean results also exhibit a very insignificant improvement when the level of depth is higher than 5. The 'Hybrid-255' and 'Hybrid-1023' do not exhibit a significant improvement. When $H = 0.9$ the 'Hybrid-1' is the only case whose mean result is over the reference line.

5.3 The dFGN Endpoints with Interpolated SRA Subtraces

Similar to the RMD algorithm, the SRA algorithm generates interpolated samples between two endpoints but attempts to overcome the shortcoming of generating independent subtraces. Therefore, the SRA algorithm can more easily be incorporated with the dFGM algorithm that produces endpoints to generate FBM samples. However, the SRA algorithm changes existing samples at every iteration. For the generated samples, the addition of the Gaussian random variables with different variance followed by Eq. (4.22) will not affect the two cascade subtraces. The two endpoints generated by the dFGN algo-

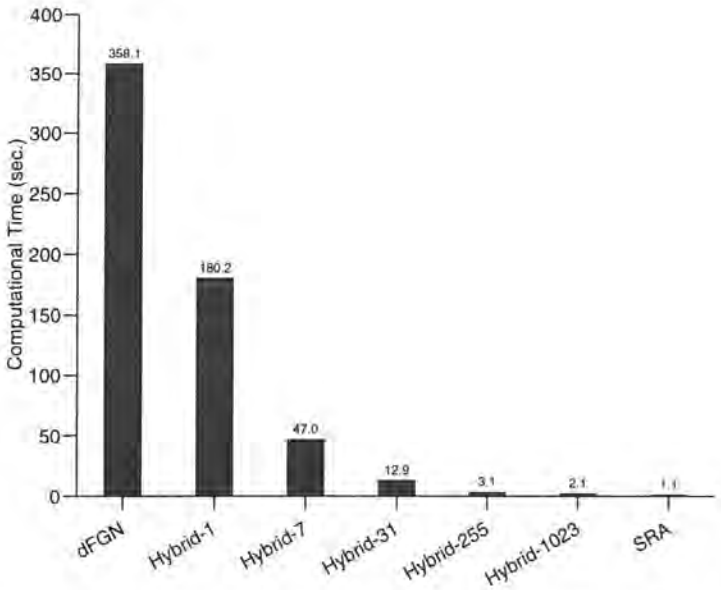


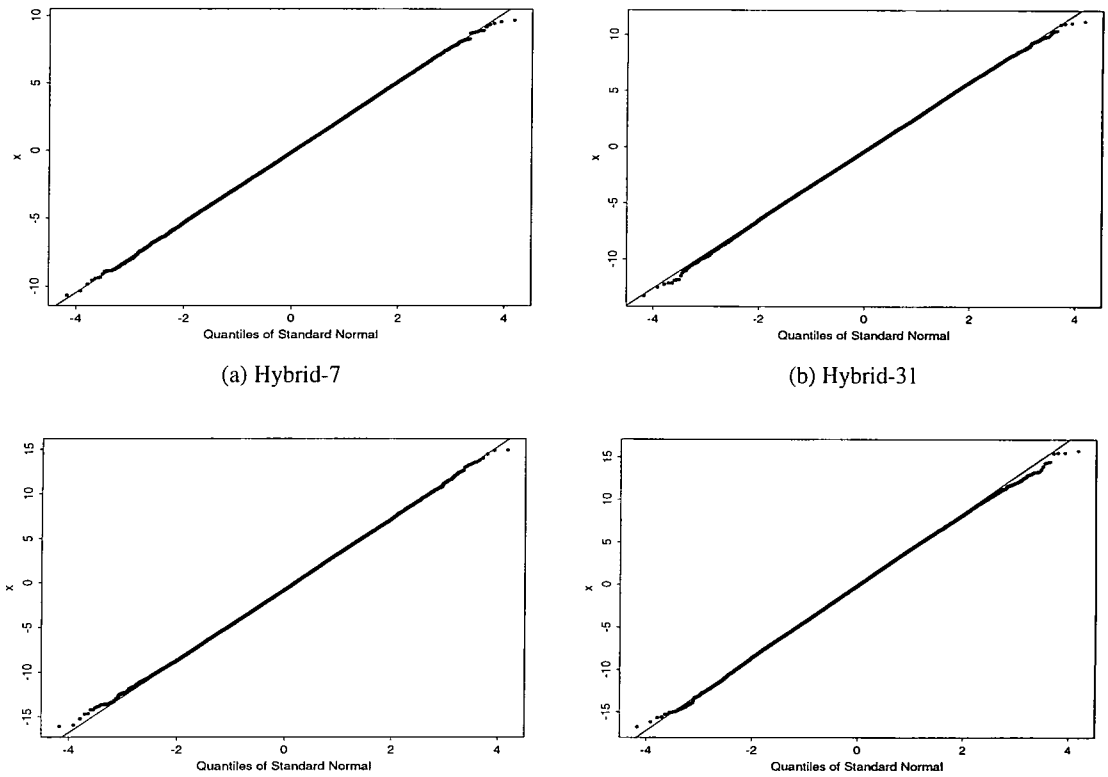
Figure 5.13: Mean computational time of the dFGN+SRA algorithm to generate 32768 sample points.

rithm need to add extra Gaussian value by each iteration as well. The endpoint between two subtraces doubly adds Gaussian random variables. Actually, the original design of the SRA algorithm was for the generation of samples from a whole flat axis. Adding extra Gaussian random variables to endpoints is necessary. To incorporate the dFGN algorithm with the SRA algorithm it is not necessary to add the extra Gaussian values to the endpoints. Therefore, the two endpoints for a subtrace are fixed. Cascading subtraces can smoothly form a FBM trace.

It is better for the sample size of the SRA algorithm to be an integer power of 2 of the level of depth. Although a generalized version has been proposed to produce samples with arbitrary size, it is too simplified to generate good sample traces. The average computational time of the dFGN, SRA, and dFGN+SRA algorithms are depicted in Fig. 5.13. All the simulations generate 32768 samples. There are at least 60 simulations for each algorithm to obtain the average computational time. The two outer bars of this figure are the average computational time of the dFGN and SRA algorithms. Between them, algorithm 'Hybrid-x' means that x samples generated by the SRA algorithm are interpolated between two end points generated by the dFGN algorithm. All the detailed computational times are shown in Table D.2. The worst case of the hybrid algorithm is about twice as fast compared to the dFGN algorithm. The computational time of the dFGM+SRA algorithm with 10 levels of depth is slightly less than twice that of the SRA algorithm. Thus, the hybrid algorithm significantly reduces the computational effort of the dFGN algorithm, but still retains the advantage of the 'on-the-fly' operation.

5.4 Statistical Analysis of the dFGN+SRA Algorithm

The generated samples of the dFGN+SRA algorithm are examined by the Q-Q plot, the R/S statistic, the VT plot, and Whittle's MLE in the following sections. The final part of this section discusses the statistical estimated results of the hybrid approach.



(a) Hybrid-7

(b) Hybrid-31

(c) Hybrid-255

(d) Hybrid-1023

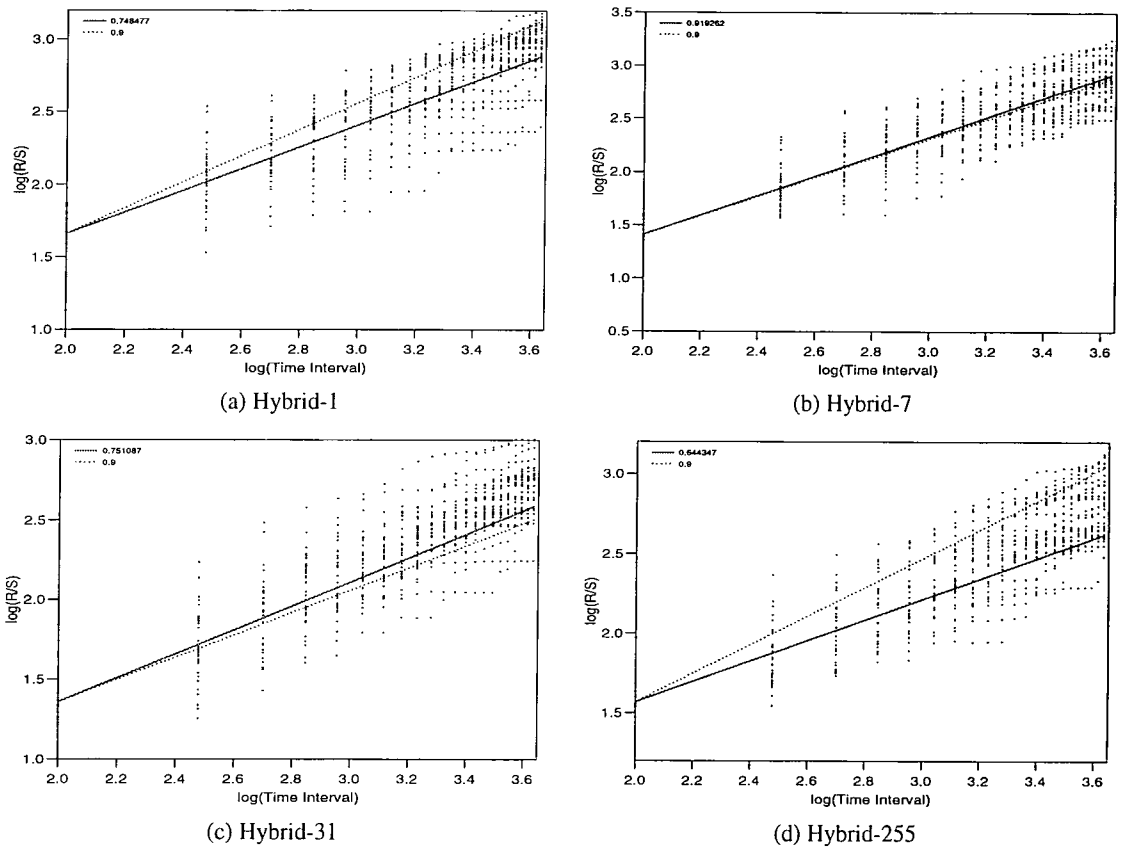
Figure 5.14: The Q-Q plot of the sample traces generated by the dFGN+SRA algorithm with $H = 0.99$.

5.4.1 The Q-Q Plot

The dFGN+SRA algorithm cascades subtraces generated by the SRA algorithm with the endpoints produced by the dFGN algorithm. Both of the fundamental algorithms are based on ordinary Gaussian random variable with weighted variances. The distribution of the hybrid approach must fit normal distribution. Four different combinations of the dFGN and SRA algorithms have been used to visualize the goodness-of-fit with the Q-Q plot as shown in Fig. 5.14. The dots are the quantile of the sample traces. The diagonal line is the ideal quantile of the normal distribution. If the dots fit the line closely, the samples are following the normal distribution. The sample traces are generated with the same random seed but with different combination schemes. It can be seen that the values of quantile in all diagrams fit the normal distribution well. Comparing with the Q-Q plots in Fig. 5.4 for the dFGN+RMD algorithms, the dFGN+SRA algorithm exhibit much better results.

5.4.2 The R/S Statistic

The pox plot of the R/S statistic of the hybrid algorithm is illustrated in Fig. 5.15. All the dots shown in the diagrams are obtained by applying Eq. (3.64). The solid line is the result of the regression and its slope is the Hurst value. The dotted line in each diagram is the reference line that represents the target Hurst parameter. In Fig. 5.15 (a) the dots are generated from a sample trace generated by the dFGN+SRA algorithm with 1 level of depth, i.e., between two endpoints produced by the dFGN algorithm only one sample is generated by the SRA algorithm. In this case, the result is significantly different from the



(a) Hybrid-1 (b) Hybrid-7

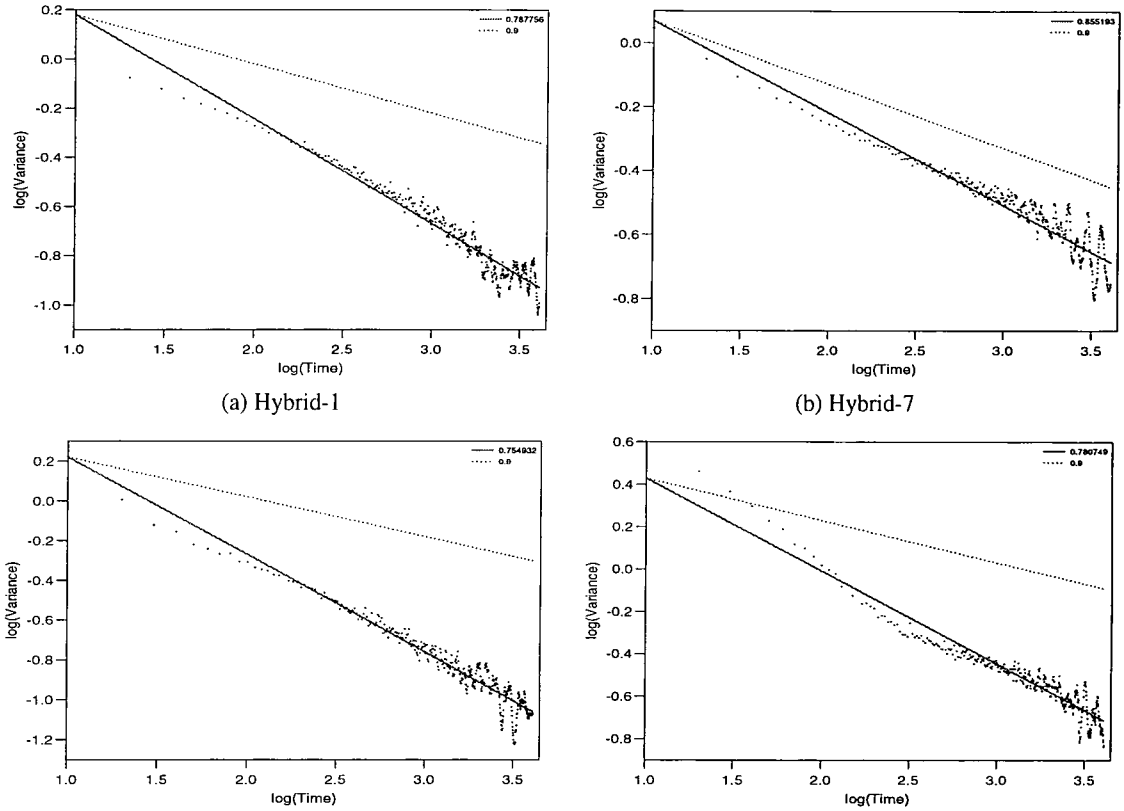
(c) Hybrid-31 (d) Hybrid-255

Figure 5.15: The pox plots of R/S statistic of the sample traces generated by the dFGN+SRA algorithm with $H = 0.9$.

target Hurst value, $H = 0.9$. The second diagram Fig. 5.15 (b), in which the samples are generated by the dFGN+SRA algorithm with 3 levels of depth fits the reference line well. However, as the level of depth increases further, the results gradually become worse.

5.4.3 The Variance-Time Plot

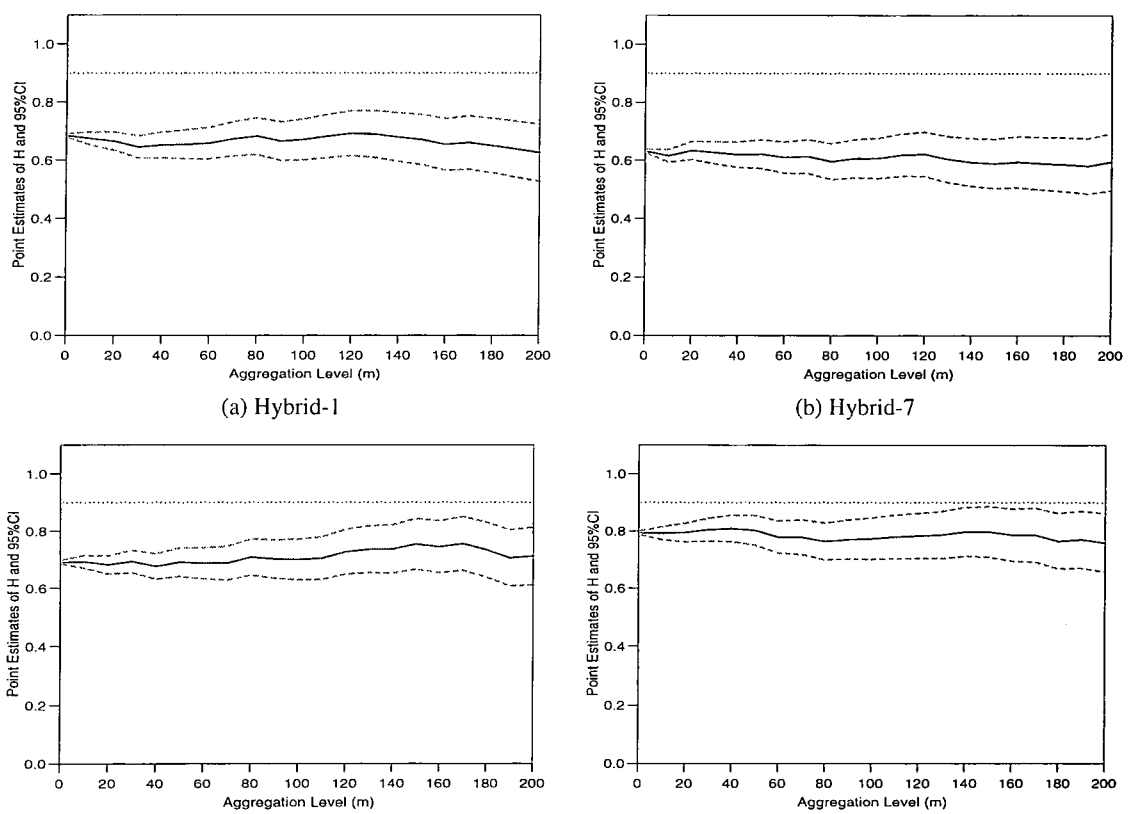
The same sample traces which are estimated by the R/S statistic are now used with the VT plot, and are depicted pictorially in Fig. 5.16. The dots in every diagram are obtained by applying Eq. (3.67). The slope of the dotted line is $2-2H$ by which the target Hurst parameter is represented. The solid line is the regressive result. The Hurst value can be obtained from the slope of the solid line. The dots calculated by the VT plot exhibit very different behavior in each of the four graphs. The dots in Fig. 5.16 (a) and (c) form convex curves while the dots in (d) form a concave curve. In Fig. 5.16 (b) the dots form more of a straight line. Thus, this value has better results than the other cases. However, the VT plot cannot accurately represent the Hurst parameters. From the previous analysis, the diagrams of the VT plot suggest that a threshold value may help to obtain a more accurate result. Fig. 5.16 (d) is a counter example, which illustrates that the slope of the dots, for values of log(Time) less than 2.5, is even worse than the result for the whole range.



(a) Hybrid-1 (b) Hybrid-7

(c) Hybrid-31 (d) Hybrid-255

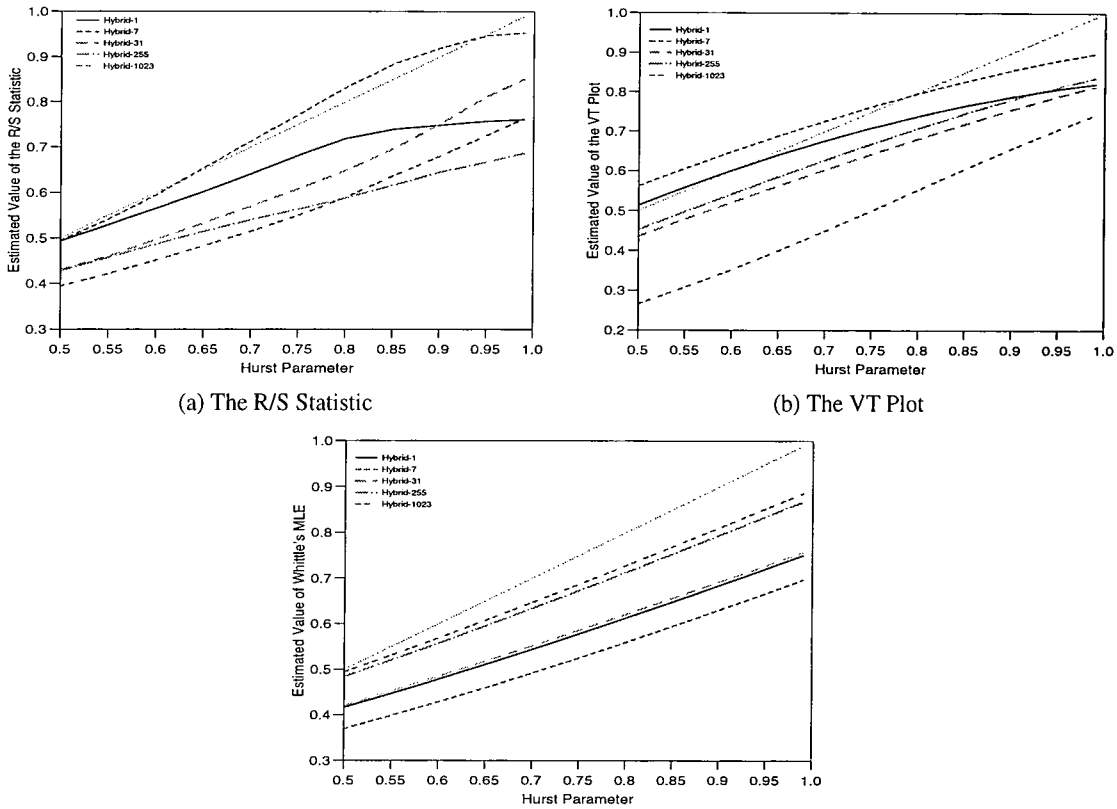
Figure 5.16: The VT plots of the sample traces generated by the dFGN+SRA algorithm with $H = 0.9$.



(a) Hybrid-1 (b) Hybrid-7

(c) Hybrid-31 (d) Hybrid-255

Figure 5.17: Analytical results of the dFGN+SRA algorithm by Whittle's MLE with different aggregation levels.



(a) The R/S Statistic

(b) The VT Plot

(c) Whittle's MLE

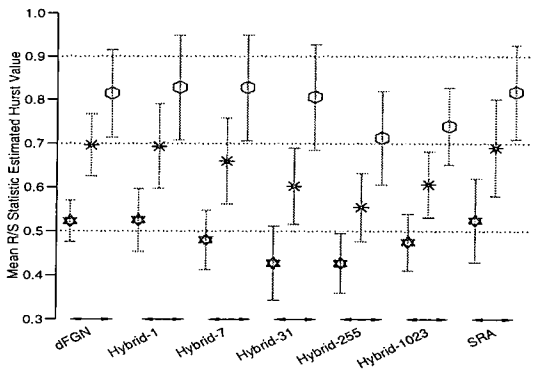
Figure 5.18: Estimated Hurst values of the dFGN+SRA algorithm with different Hurst parameters.

5.4.4 Whittle's MLE

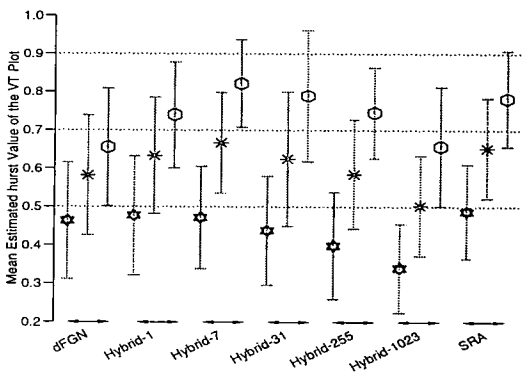
The same sample traces as used in the previous two subsections are examined by Whittle's MLE with the aggregation levels shown in Fig. 5.17. As with the previous estimators, the dotted lines in these diagrams are the reference Hurst value, $H = 0.9$. The solid lines are the results with different aggregation levels. The two dashed lines in each diagram are the upper and lower bound of 95% confidence intervals. All the diagrams in Fig. 5.17 show that the values remain the same as the aggregation level changes. However, all the results are significantly different from the target Hurst values. Even the upper bounds of the 95% confidence intervals never reach the reference lines. On comparing these four graphs the results gradually approach the reference lines with greater levels of depth. It means that the more levels of depth used in this hybrid algorithm, the more accurate the results.

5.4.5 Comparisons of Different Combination Schemes

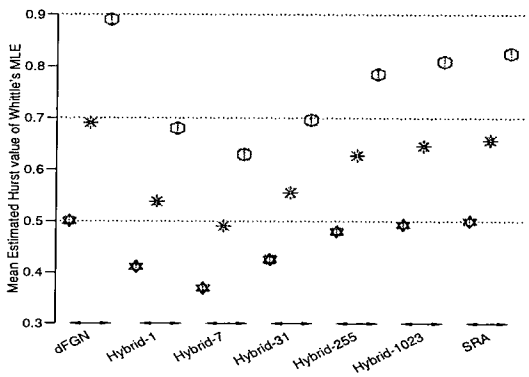
Fig. 5.18 depicts the values using selected random seed which is chosen to investigate the effect of the changing Hurst parameter. In this figure, three different estimators are illustrated with different diagrams. All the symbols of the curves are represented in the legend. A dotted straight line, which does not appear in the legend, is the reference line which is used to represent the desired Hurst values. The result using the R/S statistic with different combination schemes is shown in Fig. 5.18 (a). In this diagram, only two combination schemes, the 'Hybrid-1' and 'Hybrid-7' curves, exhibit accurate results when the Hurst parameter is $H = 0.5$. The other three schemes cannot generate accurate sample traces. When the Hurst



(a) The R/S Statistic



(b) The VT Plot



(c) Whittle's MLE

Figure 5.19: Mean estimated Hurst values of the dFGN+SRA algorithm.

parameter increases, the result of the 'Hybrid-1' scheme gradually departs from the reference line. Only the result of the 'Hybrid-7' scheme fits the reference line well.

Fig. 5.18 (b) illustrates the results using the VT plot. As with the results using the R/S statistic, the 'Hybrid-1' and 'Hybrid-7' curves are the two best combination schemes. When the Hurst parameter is lower than 0.7, the 'Hybrid-1' curve fits the reference line reasonably well. However, it gradually parts from the reference line. Although the 'Hybrid-7' curve gradually departs from the desired values if the Hurst parameter is larger than 0.8, it is the closest curve to the reference line. The 'Hybrid-1023' scheme is the worst case in this diagram. The curve starts from 0.27 and finishes at 0.74 as the Hurst parameter is set from 0.5 to 0.99.

Unlike the previous two diagrams, the 'Hybrid-7' scheme is the worst case when using Whittle's MLE in Fig. 5.18 (c). All the curves in this diagram are below the reference line, with the best results for the sample traces which are generated as short-range dependence, then they gradually depart from the desired Hurst values. The 'Hybrid-1023' scheme exhibits the best result by Whittle's MLE while the 'Hybrid-255' curve is only a little worse. The 'Hybrid-1' and 'Hybrid-31' curves form another group but are significantly worse than the best two schemes.

From previous study the R/S statistic and the VT plot have very wide confidence intervals. Only Whittle's MLE offers a concise result. Fig. 5.19 depicts the mean values of the dFGN, SRA, and dFGN+SRA algorithms and their confidence intervals. Table D.8 records the data shown in Fig. 5.19. Three separate diagrams are presented in order to compare the results of the three estimators in this figure. All these diagrams contain three dotted lines which indicates the desired Hurst values, $H = 0.5, 0.7,$

and 0.9. The symbols of the star, asterisk, and hexagon represent the mean results of the sample traces with $H = 0.5, 0.7$, and 0.9, respectively. The barline on the symbols are their 95% confidence intervals. The values displayed in this figure are obtained from twenty different sample traces. The text on the x-axis indicates the testing algorithm. The text of 'Hybrid-x' means that the sample traces are generated by the dFGN algorithm with $\log_2(x+1)$ levels of depth.

The results using the R/S statistic for this hybrid algorithm are illustrated in Fig. 5.19 (a). When examining the bottom row of the diagram ($H = 0.5$), the mean results of the 'Hybrid-31' and 'Hybrid-255' schemes are the worst two. The others have almost the same results in terms of quality, although two of them are under the reference line, while the other three are over it. When considering their confidence intervals, the dFGN algorithm still has the shortest barline while the 'Hybrid-1', 'Hybrid-7', and 'Hybrid-1023' scheme are significantly better than the SRA algorithm. For the cases with the asterisk symbol the dFGN, SRA, and 'Hybrid-1' cases lie on the desired reference line. The other four combination schemes are considerably worse than them. At the top of this diagram, all of the mean values are inaccurate. The 'Hybrid-1' and 'Hybrid-7' schemes are the best two, but their confidence intervals are slightly wider than the dFGN and SRA whose mean results are slightly worse than the best two hybrid schemes. From this diagram, the dFGN+SRA algorithm with the level of depth less than 3 can generate good quality sample traces.

The bottom row of Fig. 5.19 (b) displays the mean results using the VT plot and their confidence intervals with $H = 0.5$. The 'Hybrid-1' and 'Hybrid-7' schemes are the best two except for the SRA algorithm when the sample traces are generated with short-range dependence. When the Hurst parameter increases to 0.7, the mean result of the 'Hybrid-7' scheme is better than that of the SRA algorithm. The top row of the diagram shows that the 'Hybrid-7' scheme is the best result. Therefore, this diagram suggests that the dFGN+SRA algorithm with 3 levels of depth should generate the best quality of fractal samples. However, this suggestion is not reliable because the VT plot is a poor estimator as has been shown in our previous study.

It has been shown that Whittle's MLE is the best estimator among the three chosen estimators. The results using Whittle's MLE is illustrated in Fig. 5.19 (c). In this diagram, all the confidence intervals are so small that they can be ignored. For the 'star' symbols, the dFGN, SRA, and 'Hybrid-1023' exhibit the best three results while the others are lower than the reference line. For $H = 0.7$ and 0.9 only the dFGN case exhibits accurate results. The hybrid algorithm with different combination schemes are worse than the SRA algorithm.

According to the above discussion, the analysis of the results using the R/S statistic indicate that lower levels of depth can generate better fractal sample traces. In contrast, the dFGN+SRA algorithm with higher levels of depth produces a better quality of sample traces when examining the samples with Whittle's MLE. Actually, the lower levels of depth in the dFGN+SRA algorithm will produce larger approximation errors. From the derivation of the SRA algorithm, the basic assumption is that the FBM trace are of infinite resolution. The successive additions of the Gaussian random variables with weighted variance will be seriously truncated by the lower levels of depth. Ideally, for each sample, there should be infinite Gaussian random variables added to it. However, there are at most 10 weighted Gaussian random variables that have been added to the samples in the dFGN+SRA algorithm with 10 levels of

depth. To generate 32768 samples of fractal traces needs at the most 15 times of addition. Therefore, using the dFGN+SRA algorithm will not generate any sample traces that are more accurate than using the SRA algorithm.

Although the approximation error of the SRA algorithm can easily be overcome, it still suffers a more fundamental problem which means that this algorithm will not be better than the RMD algorithm. Initially, the SRA algorithm was proposed to overcome the shortcomings of the RMD algorithm and there was an expectation that it would produce better quality sample traces. In the RMD algorithm, a sample will not change once it is fixed, whereas the SRA algorithm continues changing the value of a sample by adding extra Gaussian random variables. Although the value of this sample changes, the sample is still independent from other samples. For every iteration, the SRA algorithm adds an extra value to each existing sample and newly generated sample. The extra Gaussian random variables are still independent. Thus, the samples are independent of each other. According to this discussion the SRA algorithm never generates fractal samples which are better in terms of quality in comparison to the RMD algorithm.

5.5 The dFGN Endpoints with Weierstrass-Mandelbrot Subtraces

In Chapter 4, the dFGN algorithm and the WM function are the two algorithms which generate the highest quality FBM or FGN sample traces. The dFGN algorithm is derived from the definition of FBM. It is a moving-average algorithm which can generate samples on-the-fly. The only drawback of this algorithm is its massive computational time. The WM function is originally a continuous non-differentiable function that has the same properties as FBM. It represents the properties of FBM in the frequency domain. Thus, to generate FBM samples using the WM function, there is a requirement to transform the random samples in the frequency domain to the time domain. Making use of the fast Fourier transform (FFT) is the most economical method to deal with this transformation. Using a hybrid of the WM function with the dFGN algorithm may reduce the computational effort and produce a better quality of fractal samples.

To combine these two algorithms is a difficult problem because the dFGN algorithm generates samples in the time domain while the WM function produces random samples in the frequency domain. The RMD and SRA algorithms were easily incorporated with the dFGN algorithm. These two algorithms generate samples by interpolating two existing samples with a number of weighted Gaussian random variables. This same method cannot be used to combine the dFGN algorithm and the WM function. To deal with this problem the samples generated by the dFGN algorithm become the major part of a subtrace, while the samples generated by the WM functions can be viewed as the fluctuations of the subtraces. The difference of two endpoints will cause a monotonic increasing or decreasing function within them.

Let $B_H(t)$ be a FBM process whose values at two time epochs are known. Assume that $B_H(t) = a$ and $B_H(t+T) = b$. Without loss of generality, let the time epoch t be 0. So, $B_H(0) = a$ and $B_H(T) = b$.

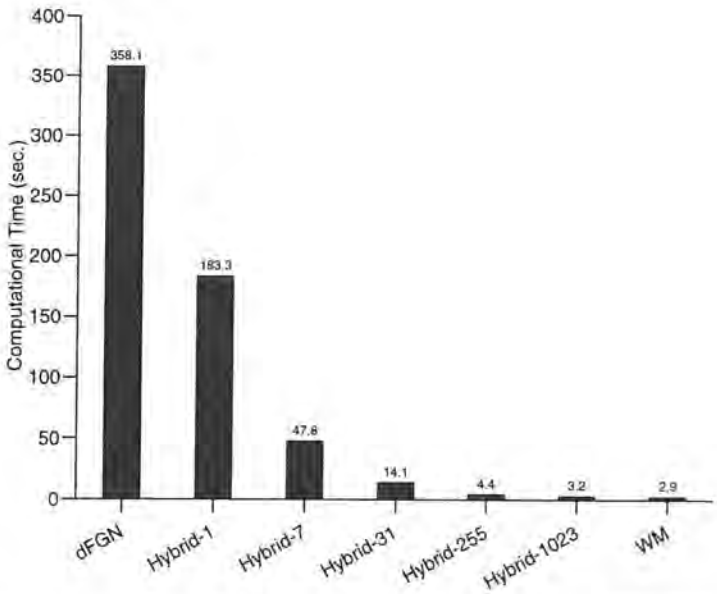


Figure 5.20: Mean computational time of the dFGN+WM algorithm to generate 32768 sample points.

The variance of the FBM process can be expressed as

$$E[|B_H(T) - B_H(0)|^2] = \sigma^2 = (b-a)^2. \tag{5.1}$$

Then, the FBM process between these two time epochs is

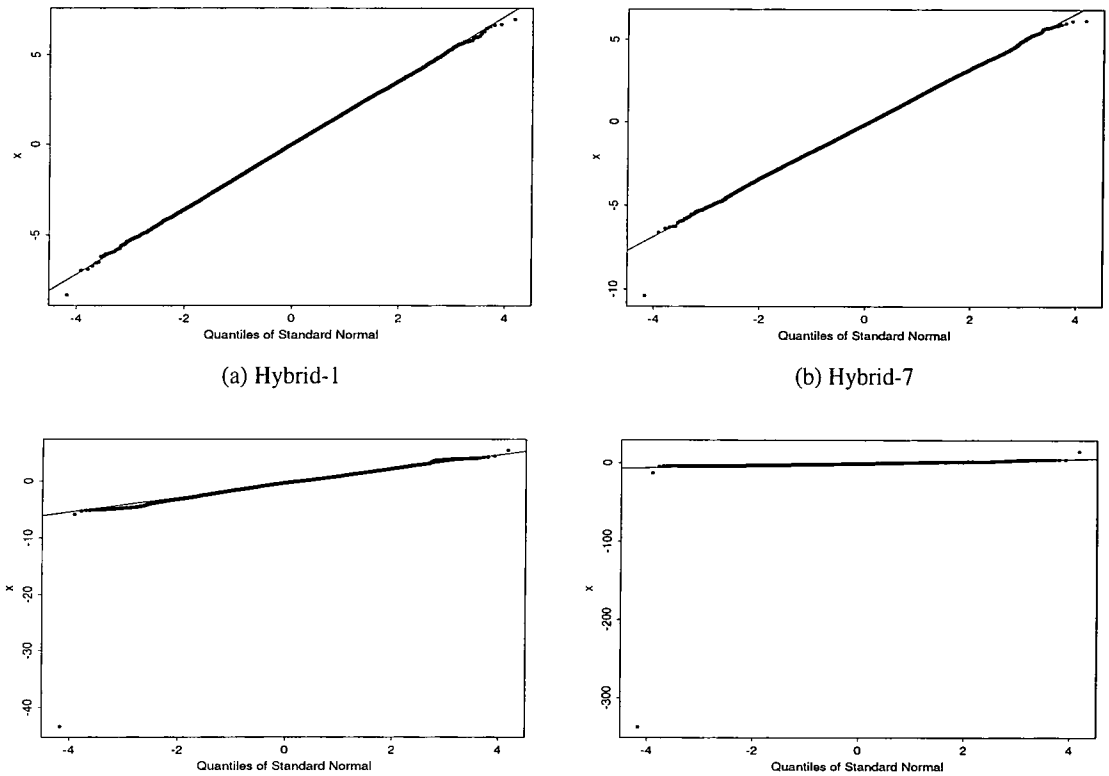
$$B_H(t) = \frac{t}{T}(b-a) + B_H(0) + B_{H,WM}(t), \text{ with } 0 \le t < T, \tag{5.2}$$

where $B_{H,WM}(t)$ is the fluctuating part of the subtrace. The magnitude part of the WM function follows Gaussian random variables while the phase part follows an exponential distribution. Therefore, the variance of the magnitude part must be revised. After derivation (see Appendix B), the variance of a Gaussian random variable is

$$\sigma_f^2(k) = [(\frac{k}{N})^{2H}\sigma^2 - (\frac{k}{N})^2(b-a)^2] / \sum_{n=1}^{N} n^{-1-2H} \tag{5.3}$$

$$= \sigma^2[(\frac{k}{N})^{2H} - (\frac{k}{N})^2] / \sum_{n=1}^{N} n^{-1-2H}. \tag{5.4}$$

The comparison of the computational times of the dFGN algorithm, the WM function, and the dFGM+WM algorithm is illustrated in Fig. 5.20. The two outer bars of this figure are the results of the dFGN algorithm and the WM function. Between them, the results of different combination schemes of the dFGN+WM algorithm are depicted. Similar to the previous hybrid approaches, algorithm 'Hybrid-x' indicates that x samples generated by the WM function are placed within two end points generated by the dFGN algorithm. The results shown in Fig. 5.20 are the average values of at least sixty simulations for each algorithm. Every simulation generates 32768 samples. The average computational time of the WM function is slightly less than 3 seconds while that of the dFGN algorithm is about 120 times more. The worst case of the hybrid algorithm is about half of the computational time of the dFGN algorithm. The 'Hybrid-1023' scheme is the best of the dFGN+WM algorithm in which the average simulation time is 23% more than the WM function. Therefore, the hybrid approach significantly reduces the disadvantage of the dFGN algorithm.



(a) Hybrid-1 (b) Hybrid-7

(c) Hybrid-31 (d) Hybrid-255

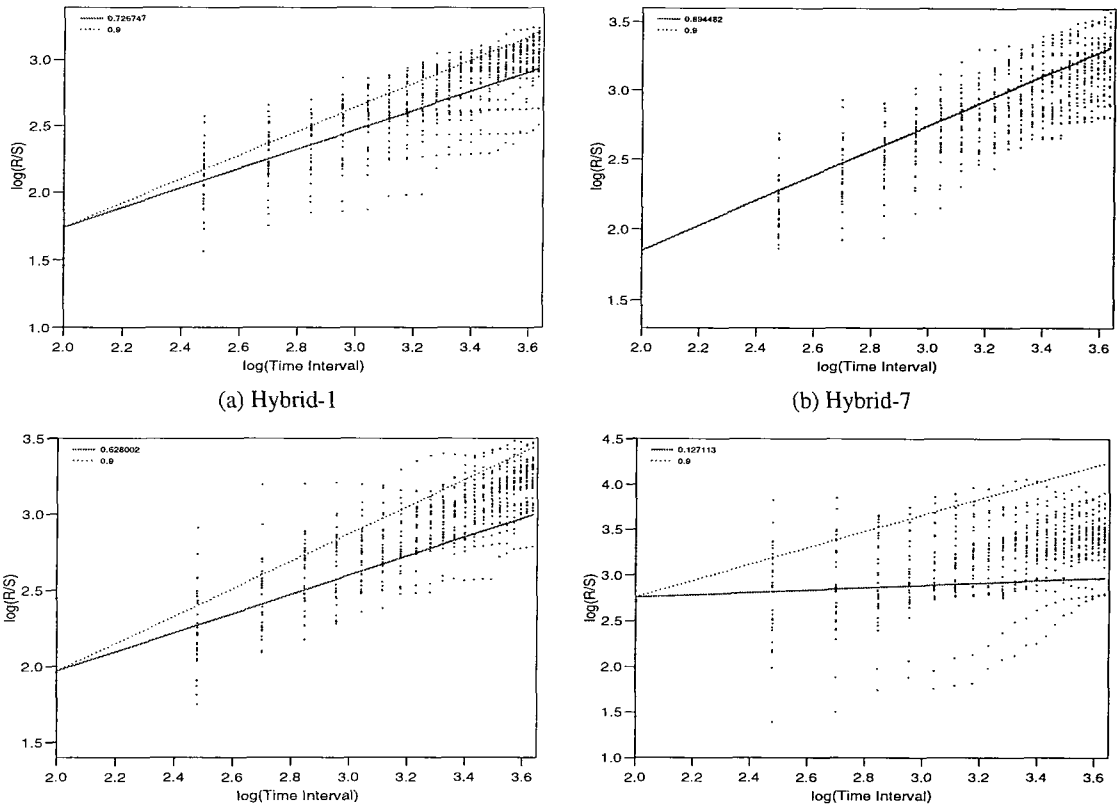
Figure 5.21: The Q-Q plot of the sample traces generated by the dFGN+WM algorithm with $H = 0.99$.

5.6 Statistical Analysis of the dFGN+WM Algorithm

The simulation results of the dFGN+WM algorithm are examined by the Q-Q plot, the R/S statistic, the VT plot, and Whittle's MLE. First, the Q-Q plot is used to examine the goodness-of-fit of the simulation samples. Then, the pox plot of the R/S statistic, the plot of variance versus time, and Whittle's MLE with aggregation levels are applied to extract the Hurst values from the simulation results. In the final part of this section, many simulations with the same combination scheme and Hurst parameter are used to test the robustness of the algorithm.

5.6.1 The Q-Q Plot

The dFGN algorithm generates samples by adding together weighted Gaussian random variables with a weighting function. Therefore, its samples should fit the normal distribution. Although the WM function generates its samples in the frequency domain, the magnitude part is based on Gaussian random variables and its phase is a uniform distribution. After transformation, the samples in the time domain become a normal distribution as well. Four different combination schemes of the dFGN+WM algorithm are examined by the Q-Q plot with normal distribution as shown in Fig. 5.21. The dots in these diagrams are their quantile values. The reference line in each diagram represents the ideal quantiles of normal distribution. All the diagrams exhibit that the samples fit a normal distribution well. For the 'Hybrid-31' and 'Hybrid-255' schemes, most of the samples fit the reference line accurately, but a sample point is distant from the reference line in the left bottom corner. However, this sample will not affect the matching



(a) Hybrid-1 (b) Hybrid-7

(c) Hybrid-31 (d) Hybrid-255

Figure 5.22: The pox plots of R/S statistic of the sample traces generated by the dFGN+WM algorithm with $H = 0.9$.

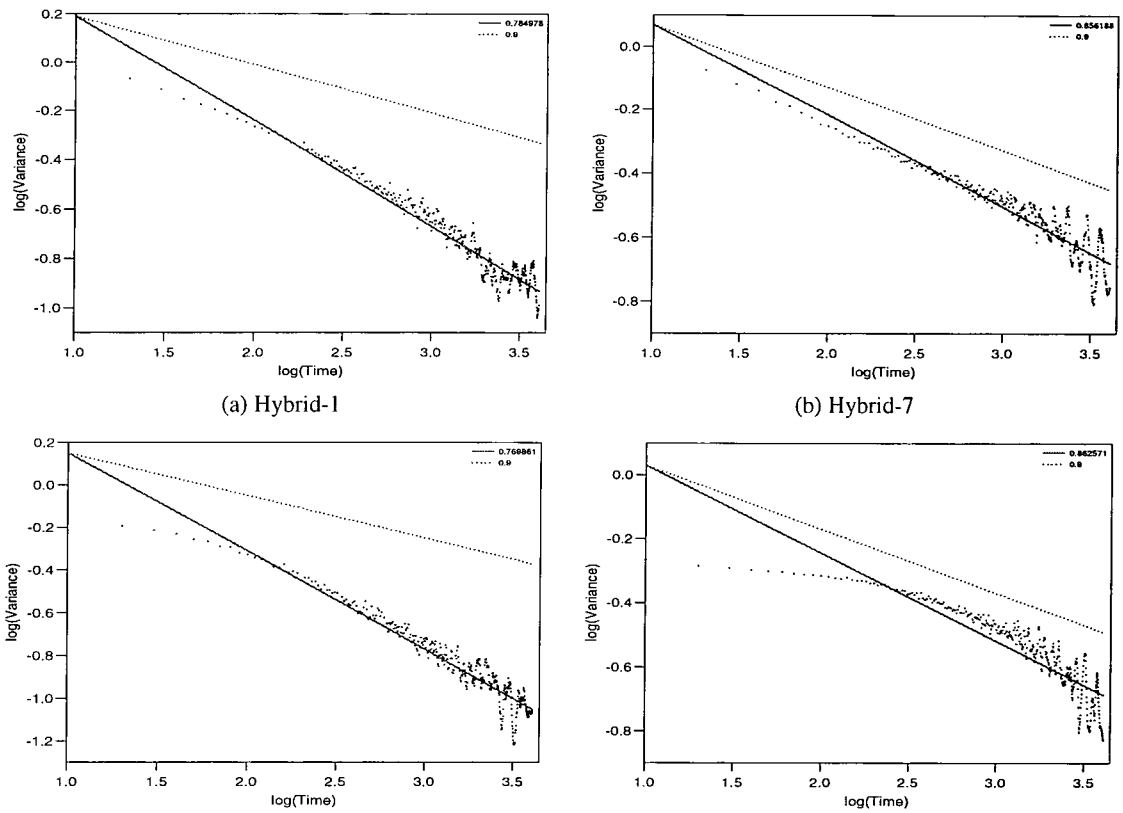
of a normal distribution.

5.6.2 The R/S Statistic

The same selected random seed as previous subsections is used to generate samples with $H = 0.9$. Their R/S statistic results are illustrated in Fig. 5.22. The dots shown in these four diagrams are obtained by the calculation of rescaled adjusted range. The solid line is the regressive result of the dots while the dotted line is the desired Hurst value. The slope of these two lines are the Hurst result and the ideal Hurst value, respectively. Fig. 5.22 (a) illustrates the result of the dFGN+WM algorithm in which the sample generated by the dFGN algorithm follows the sample generated by the WM function, and vice versa. For the 'Hybrid-7' scheme the samples generated by the dFGN+WM algorithm accurately fit the desired Hurst value, as shown in Fig. 5.21 (b). When the size of the interpolated WM sample increases, the result decreases dramatically. In Fig. 5.22 (d), the dots of the R/S results are scattered over the diagram and its regressive result is almost zero, a line parallel to the x-axis.

5.6.3 The Variance-Time Plot

The same sample traces analyzed in the R/S statistic are examined by the VT plot as well, shown in Fig. 5.23. Four different combination schemes are tested in this figure. The dots in these diagrams represent the calculated results. The Hurst values of these sample traces can be obtained from the slope



(a) Hybrid-1 (b) Hybrid-7

(c) Hybrid-31 (d) Hybrid-255

Figure 5.23: The VT plots of the sample traces generated by the dFGN+WM algorithm with $H = 0.9$.

of the regressive line, β, following the relationship, $\beta = 2 - 2H$. The dots in the 'Hybrid-31' and 'Hybrid-255' schemes form nonlinear curves while the other two form straighter curves. The 'Hybrid-7' and 'Hybrid-255' cases are two which fit the reference line reasonably accurately, especially in comparison to the other two schemes. From previous analysis the 'Hybrid-255' scheme is not so reliable, therefore this might mislead the result. The dots of the 'Hybrid-255' scheme form a nonlinear curve. The regressive line cannot very well match the dots, so the real Hurst value might change from case to case.

5.6.4 Whittle's MLE

Fig. 5.24 depicts the results using Whittle's MLE with different aggregation levels. The solid line represents the point estimated result using Whittle's MLE. The two dashed lines in each diagrams are the upper and lower bounds of 95% confidence intervals of the solid line. The dotted lines in this figure are the desired Hurst parameter, $H = 0.9$. In Fig. 5.24 (a) the result is almost parallel to the reference line, but the value is significantly lower than the desired Hurst value. The result of the 'Hybrid-7' scheme fits the reference line when the aggregation level is low, but the result decreases dramatically as the aggregation level increases. The results of the 'Hybrid-31' scheme shown in Fig. 5.24 (c) are much higher than the desired Hurst value when the aggregation level is low and then they decrease quickly as the aggregation level increases. For the 'Hybrid-255' scheme, the result is always much lower than the target Hurst value, $H = 0.9$. The curve in Fig. 5.24 (d) fluctuates and decreases as the aggregate level increases. The point result is even lower than 0.2 if the aggregation level is larger than 130.

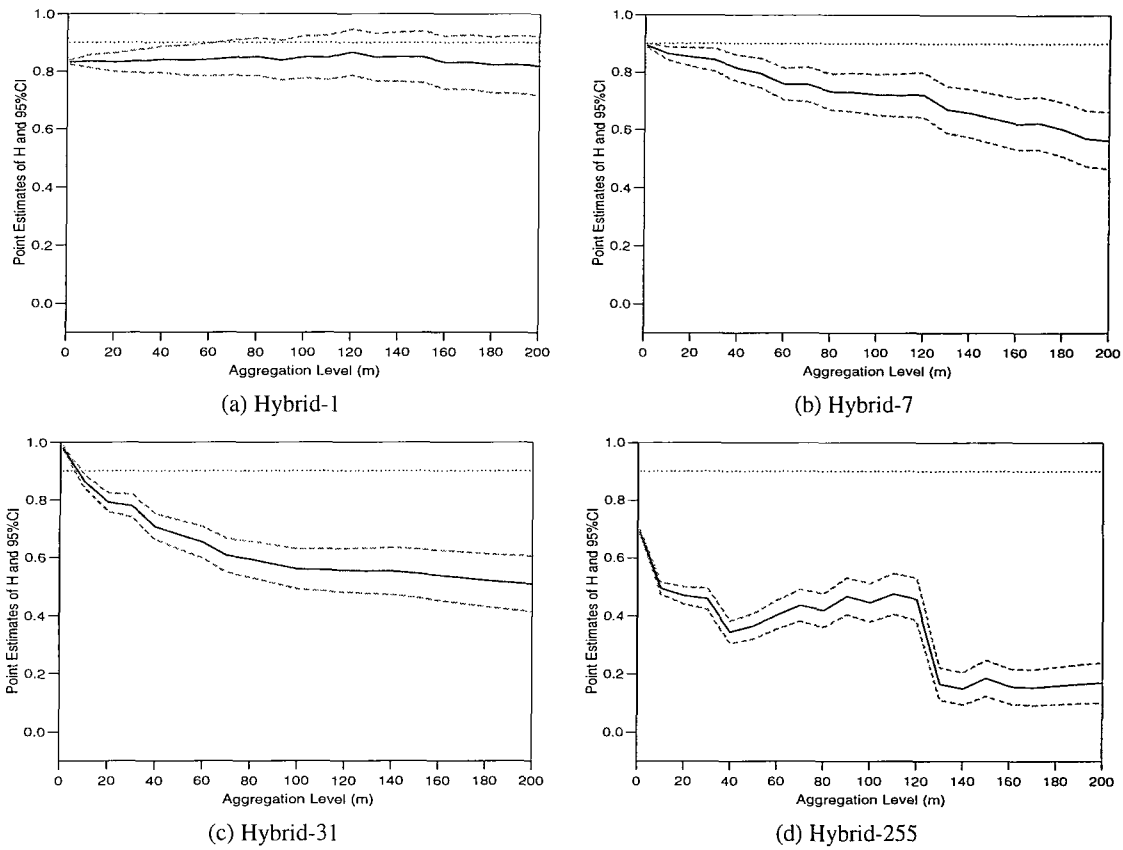


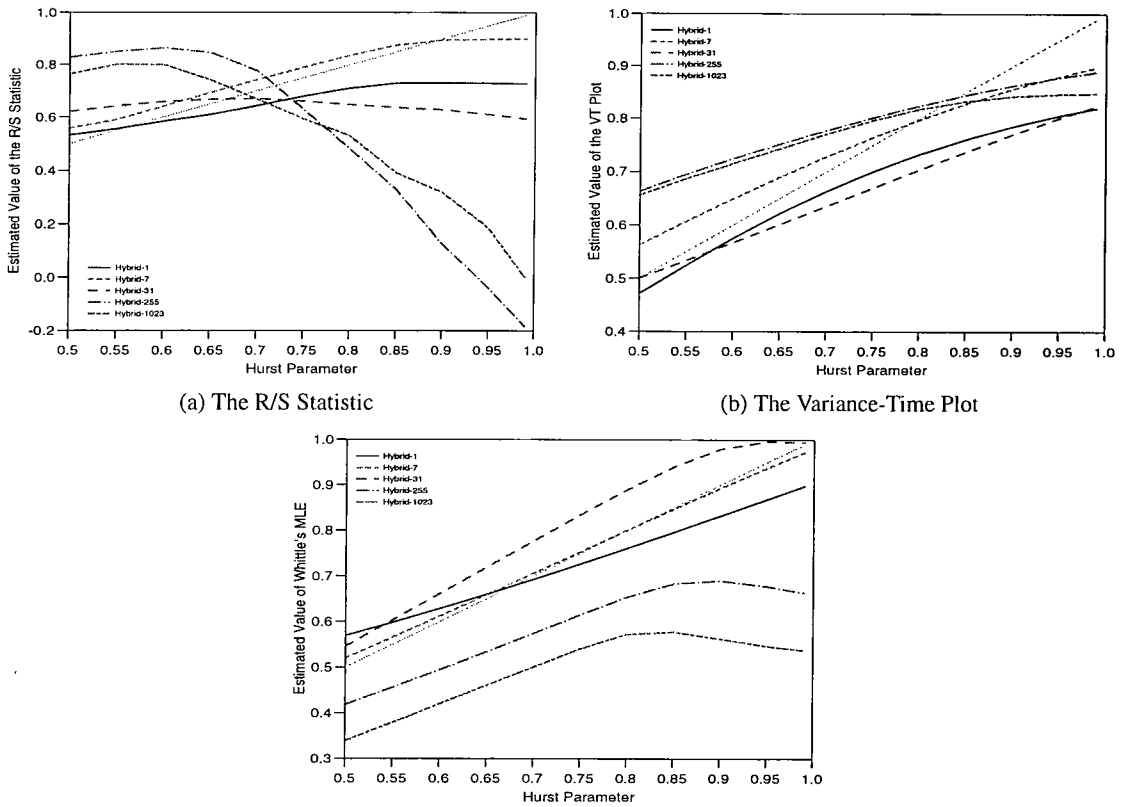
Figure 5.24: Analytical results of the dFGN+WM algorithm by Whittle's MLE with different aggregation levels.

5.6.5 Comparisons of Different Combinations

The effect of changing the Hurst parameter from short-range dependence to LRD is illustrated in Fig. 5.25. Three diagrams are used to present the results using the R/S statistic, the VT plot, and Whittle's MLE with different combination schemes. The meanings of the lines are indicated in the legend. The reference dotted line not shown in the legend is the target Hurst value.

The curves shown in Fig. 5.25 (a) are not similar. Curve 'Hybrid-255' and curve 'Hybrid-1023' slightly increase, then dramatically decrease as the Hurst parameter increases, to the extent that the values are even less than 0. This result violates the fundamental definition of FBM whose Hurst parameter must be in the interval $(0, 1)$. Of the other three curves, the 'Hybrid-7' scheme exhibits the best result, which is almost along the reference line. The value using the R/S statistic with 'Hybrid-7' scheme is slightly larger than the target value when the Hurst parameter is less than 0.9, thereafter the result is less than the target Hurst value. Although curve 'Hybrid-1' exhibits better results while the Hurst parameter is less than the 0.65, thereafter it continuously departs from the reference line.

In Fig. 5.25 (b) the results using the VT plot shows that the 'Hybrid-7' scheme is again the best result. Although the 'Hybrid-1' and 'Hybrid-31' schemes have more accurate results when the Hurst parameter is less than 0.7, the line quickly decreases as the sample traces have a high degree of self-similarity. The results using Whittle's MLE are shown in Fig. 5.25 (c). Curve 'Hybrid-155' and curve 'Hybrid-1023' are much lower than the reference line. Moreover, the results of the 'Hybrid-255' and 'Hybrid-1023'



(a) The R/S Statistic

(b) The Variance-Time Plot

(c) Whittle's MLE

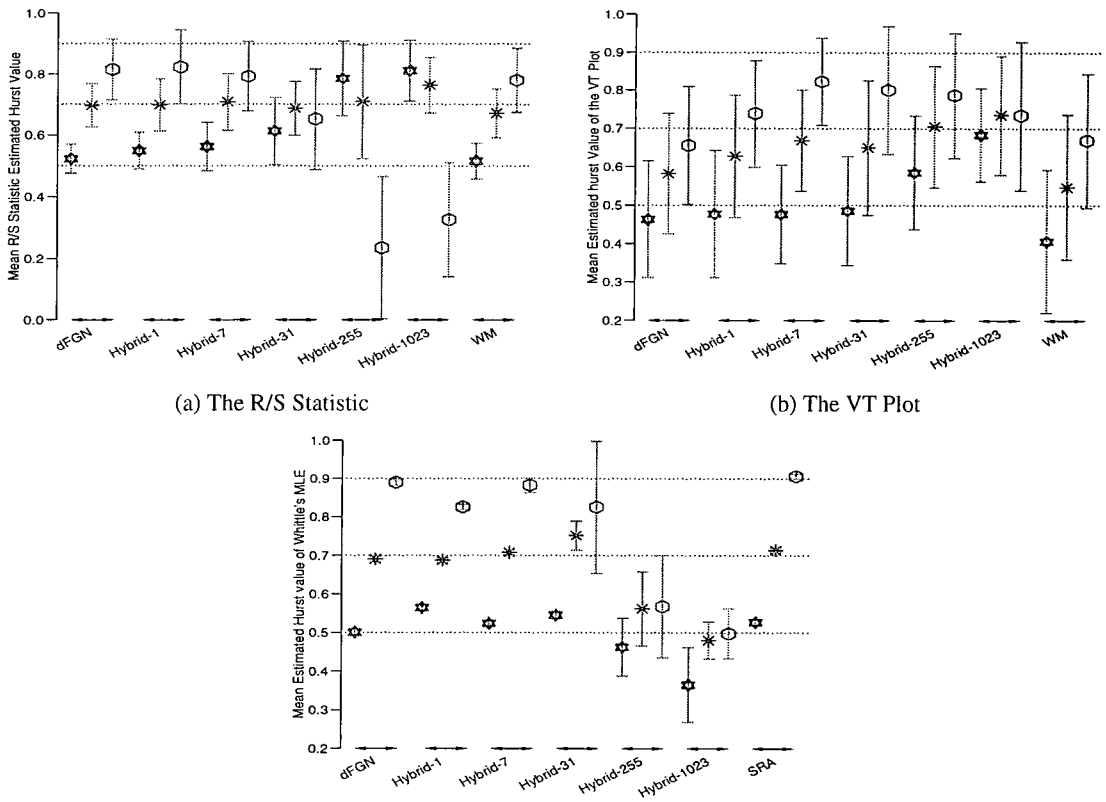
Figure 5.25: Estimated Hurst values of the dFGN+WM algorithm with different Hurst parameters.

schemes even decrease when the Hurst parameter is in excess of 0.8 and 0.9, respectively. Among the other three schemes, curve 'Hybrid-7' once more exhibits the most accurate result.

The mean results of the three estimators and their 95% confidence intervals are illustrated in Fig. 5.26. In these three diagrams, the symbols of the star, asterisk, and hexagon represent the mean results of $H = 0.5, 0.7$, and 0.9, respectively. The barlines on the symbols are their 95% confidence intervals. The three dotted lines are the target Hurst values, $H = 0.5, 0.7$, and 0.9, respectively. The two ends of these diagrams are the mean results of the dFGM algorithm and the WM function. Between them are the dFGN+WM algorithm with different combination schemes.

For the mean results using the R/S statistic in Fig. 5.26 (a) the mean results of the dFGN algorithm and the WM function with $H = 0.5$ are better than other schemes. The mean results of the dFGN+WM algorithm with $H = 0.5$ dramatically increase as the size of the interpolated WM samples increase. The mean results of the 'Hybrid-255' and 'Hybrid-1023' are even higher than 0.7. The confidence intervals of the hybrid algorithm gradually increase. For $H = 0.7$, most of the mean results are on the reference line except for the 'Hybrid-1023' scheme. For $H = 0.9$, the mean results of the 'Hybrid-1' scheme is even better than that of the dFGN algorithm. When the size of interpolated WM samples increases, the mean result of the dFGM+WM algorithm decreases dramatically. The worst cases are the mean result of the 'Hybrid-255' scheme and the 'Hybrid-1023' scheme. These last two combination schemes of the dFGN+WM algorithm are not worth considering.

Fig. 5.26 (b) illustrates the mean results using the VT plot. For $H = 0.5$, the results of the 'Hybrid-1', 'Hybrid-7', and 'Hybrid-31' schemes gradually approach the target Hurst parameter. Those of the



(a) The R/S Statistic

(b) The VT Plot

(c) Whittle's MLE

Figure 5.26: Mean estimated Hurst values of the dFGN+WM algorithm.

'Hybrid-255' and 'Hybrid-1023' schemes sharply increase and are significantly more than the desired value. The result of the 'Hybrid-1023' scheme even reaches 0.7. Normally the mean results of the dFGN+WM algorithm with $H = 0.7$ increase as the size of subtraces generated by the WM function increases. The exceptional case is the 'Hybrid-31' scheme in which its mean result is lower than those of the 'Hybrid-7' and 'Hybrid-1023' schemes. The maximum mean result is located on the 'Hybrid-7' scheme which has the best result to fit the target Hurst value, 0.9. Overall, the 'Hybrid-7' scheme of the dFGN+WM algorithm exhibits the best result. However, the analytical results cannot fully reflect the actual Hurst values from the previous analysis. In addition, the confidence intervals of all mean results are very large. It might mislead the result of the Hurst parameter if there is only one trace.

The mean results using Whittle's MLE are depicted in Fig. 5.26 (c). The mean results of the dFGN algorithm with the three Hurst parameters fit the reference lines very well. The 'Hybrid-1' scheme of the dFGN+WM algorithm with $H = 0.7$ is just slightly less than the target Hurst value. For the short-range dependence, the mean result is higher than the target Hurst value while that of $H = 0.9$ is much smaller than its target Hurst parameter. The 'Hybrid-7' scheme of the dFGN+WM algorithm is slightly worse than the dFGN algorithm and almost the same as the WM function. Its mean values almost lie on the reference lines for each Hurst setting. For the 'Hybrid-31' scheme the mean results are much worse than those of the 'Hybrid-7' scheme. Its mean results of $H = 0.7$ is higher than the reference line and its 95% confidence interval is significantly larger in comparison to the confidence intervals of the dFGN algorithm and the WM function. The mean result of $H = 0.9$ is much lower than the reference line and its confidence interval is so huge that this combination scheme is not worth concerning. The mean results of

the 'Hybrid-255' and 'Hybrid-1023' schemes are all near 0.5. Their confidence intervals are too large to be of value. Therefore, the 'Hybrid-7' scheme of this hybrid algorithm is the only combination scheme that is worth using to generate FBM samples.

A thorough investigation of the various effect has led to an understanding of the reasons why some of the results are useful and other are disappointing. The nature of the mechanism used has an effect on the outcome. To improve the situation would require a totally different approach and is an avenue for further research.

When comparing the dFGN+RMD and the dFGN+WM algorithms, the computational time of the dFGN+RMD algorithm is slightly better than that of the dFGN+WM algorithm. The dFGN+RMD algorithm with 5 levels of depth is the best choice of this hybrid algorithm, while the 'Hybrid-7' scheme is the best choice of the dFGN+WM algorithm. Thus, the dFGN+RMD algorithm is the best solution for the aspect of computational effort. When examining the results of these two hybrid algorithms with Whittle's MLE from Table D.7 and Table D.9, the dFGN+RMD algorithm is more accurate as $H = 0.5$, but the dFGN+WM algorithm has better results as $H = 0.7$ and 0.9. However, when comparing these two hybrid approaches with the 'Hybrid-7' scheme, the results of the dFGN+WM algorithm do not show any improvement. Therefore, the dFGN+RMD algorithm is the best choice to generate FBM samples fast, accurately, and on-the-fly.

5.7 Summary

The investigation of the hybrid approaches has demonstrated that the simplest combination scheme can reduce by about half, the simulation time of the dFGN algorithm. A higher level of depth will significantly reduce the computational time further. When the level of depth is set as 5, the simulation time of the dFGN+RMD algorithm is reduced to about $\frac{1}{30}$ of that of the dFGN algorithm. The computational effort is just less than twice the simulation time of the RMD algorithm if the level of depth is 10. The dFGN+SRA and dFGN+WM algorithms also significantly reduce the computational effort.

The statistical analysis shows that 5 levels of depth is the most appropriate setting for the dFGN+RMD algorithm. The results of the Q-Q plots indicate that the sample paths are not sufficiently close to the normal distribution if the level of depth is higher than 7 is used. When the level of depth is larger than 5, the results using Whittle's MLE for the hybrid are worse than that of the RMD algorithm. Fig. 5.8 shows that with a higher Hurst parameter, a level of depth larger than 5 will have a sharp drop when the aggregate level increases. In addition, the mean results using Whittle's MLE also indicate that for a level of depth larger than 5 there is no significant improvement in approaching the reference line when the Hurst parameter is set as 0.7 and 0.9.

From the analysis of the dFGN+SRA algorithm, it is found that the SRA algorithm does not overcome the disadvantage of the RMD algorithm. The analytical results of the dFGN+SRA algorithm are much worse than those of the dFGN+RMD algorithm, especially when the level of depth increases.

For the dFGN+WM algorithm, the 'Hybrid-7' scheme is the only case which is worth considering. However, when comparing it to the similar scheme of the dFGN+RMD algorithm, this algorithm does not significantly improve the accuracy while being slower. Thus, the dFGN+RMD algorithm with 5

levels of depth is the best choice to generate fast and accurate samples on-the-fly.

Several algorithms and their hybrid approaches have been investigated in the previous and current chapters. The dFGN algorithm has proved to be the best one in terms of quality while the RMD algorithm is good in terms of speed. Their combination represents a way to reduce the computational effort and improve the quality of traces. In addition, the hybrid algorithm can generate samples on-the-fly. It can be used for a long-term network simulation which has fractal inputs.

In the next chapter, the marginal distribution model and the storage model are introduced and applied to generate fractal traffic to enable further investigation to be carried out.

Chapter 6

Generating Self-Similar Traffic

The previous chapters have focused on the generation of FGN or FBM traces. For network simulations, the counts of arrivals must be non-negative and integer. However, FBM is not such a stochastic process. Transformations must be carried out to convert the self-similar stochastic process to self-similar traffic. The marginal distribution model and the storage model are two models which can transform the FBM or FGN samples to self-similar counting processes.

This chapter is divided into four parts. Sec. 6.1 and Sec. 6.3 introduce the marginal model and the storage model, respectively. Sec. 6.2 and Sec. 6.4 examine their numerical results using three Hurst estimators.

6.1 The Marginal Distribution Model

The marginal distribution model makes use of the invariant property of transferring a distribution to another distribution in which several characteristics of the original samples remain the same. All the FGN traces can be modeled by the normal distribution, therefore the Q-Q plot has been used to examine the generated samples. Using the marginal distribution model for the transformation from a normal distribution to an arbitrary distribution will maintain the degree of correlation of the samples. Exponential, gamma, and Pareto distributions are applied to generate self-similar traffic streams.

6.1.1 Arbitrary Marginal Distribution

Let $G_H(t)$ be a normalized fractional Gaussian noise, i.e., zero mean, unit variance. For each epoch of $G_H(t)$, $G_H[k]$ can be modeled by a normal distribution. Let $F_N(x)$ be the marginal cumulative probability function. Let $F_Y(y)$ be a marginal cumulative probability function corresponding to a process $Y[k]$. The process $Y[k]$ can be generated by

$$Y[k] = F_Y^{-1}(F_N(G_H[k])),\ k = 1,2,\cdots, \tag{6.1}$$

with the desired marginal cumulative probability function $F_Y[y]$ from $G_H[k]$.

For real applications there are two possible methods to model the cumulative probability function $F_Y(y)$. It can be obtained either by modeling an empirical distribution using parametric mathematical

functions or by inverting the empirical distribution from traffic measures. The invariance property the marginal cumulative distribution $F_Y(y)$ has the same Hurst parameter as $F_G(x)$ [PJS92,ST94,HDLK95b].

The self-similarity with the addition of some 'small scale structural assumptions', consisting of a sequence $(t_1,n_1),(t_2,n_2),\cdots$, where the t_i are interarrival times and the n_i are number of arrivals, can be generated by the following procedure introduced in [DS96]:

1. Generating fractional Gaussian noise $G_H[k]$ with zero mean and unit variance.

2. Mapping the random variable $G_H[k]$ to an arbitrary distributed random variable $Y[k]$ by letting

$$Y[k] = F_Y^{-1}(F_N(G_H[k])) \tag{6.2}$$

   where F_N is the cumulative distribution of the standard Gaussian and F_Y^{-1} is the inverse of arbitrary distribution.

3. Computing the interarrival arrivals for the time interval $k\tau \leq t \leq (k+1)\tau$ by letting the number of arrivals be

$$m_k = \left\lceil X[k] - \frac{1}{2} \right\rceil \tag{6.3}$$

If the time scale is significantly small, every time interval can either have one or zero arrival, however, this method is not efficient. It is reasonable to select an appropriate time scale depending on the arrival rate and the requirements of the simulation. Furthermore, the arrivals which occur during a time interval can be assumed to follow a uniform distribution. More elaborate small scale models may be chosen, such as the Poisson processes with the mean rate calculated by the generated packets and the time interval length. For different data traffic, packet sizes can use the mean sizes or incorporate with some other random distributions from the real data measurements.

6.1.2 Exponential Distribution

The simplest distribution for generating self-similar traffic is an exponential distribution [DS96]. Let $N(t)$ be the total number of arrivals put on the network in the time interval $[0,t)$. If $N(t)$ has stationary increments and $E[N(1)] < \infty$, then $E[N(t)] = E[N(1)]t$. Let $E[N(1)] = \lambda$, the

$$E[N(t)] = \lambda t. \tag{6.4}$$

The constant is the traffic rate measured in arrivals per unit time. Here, the unit of arrivals could be bit, byte, frame, etc. The counting process $N(t)$ must be a non-negative integer and non-decreasing. Assume that $Y(t) = N(t) - \lambda t$ is an LRD process, then the counting process $N(t)$ is also a self-similar process. By the scaling property, $N(t) = Y(t) + \lambda t$ has the same distribution as $t^H Y(1) + \lambda t$ which has the same distribution as $Y(1)$, except for a dilation and a shift.

In self-similar traffic models the distribution of $Y(t)$ is popularly chosen as a Gaussian random variable. However, Gaussian distributions violate the non-decreasing requirement for $N(t)$ because any

Gaussian random variable assumes negative values with positive probability. The difference between two epochs of $N(t)$,

$$N(t+\tau) - N(t) = Y(t+\tau) - Y(t) + \lambda\tau, \tag{6.5}$$

has the same distribution as the Gaussian random variable $\tau^H Y(1) + \lambda\tau$. Therefore, the exponential distribution is a distribution that follows the non-negative and non-decreasing requirements.

To apply the algorithm described in Sec. 6.1.1 consider an exponential distribution with mean $\mu = \lambda\tau$. The probability density function of the exponential distribution is

$$f_E(x) = \mu e^{-\mu x}, \tag{6.6}$$

and its corresponding cumulative probability function is

$$P(x) = F_E(x) = 1 - e^{-\mu x}. \tag{6.7}$$

Therefore, the inverse of the exponential distribution can be expressed as

$$F_E^{-1}(x) = -\mu \log(1 - P(x)). \tag{6.8}$$

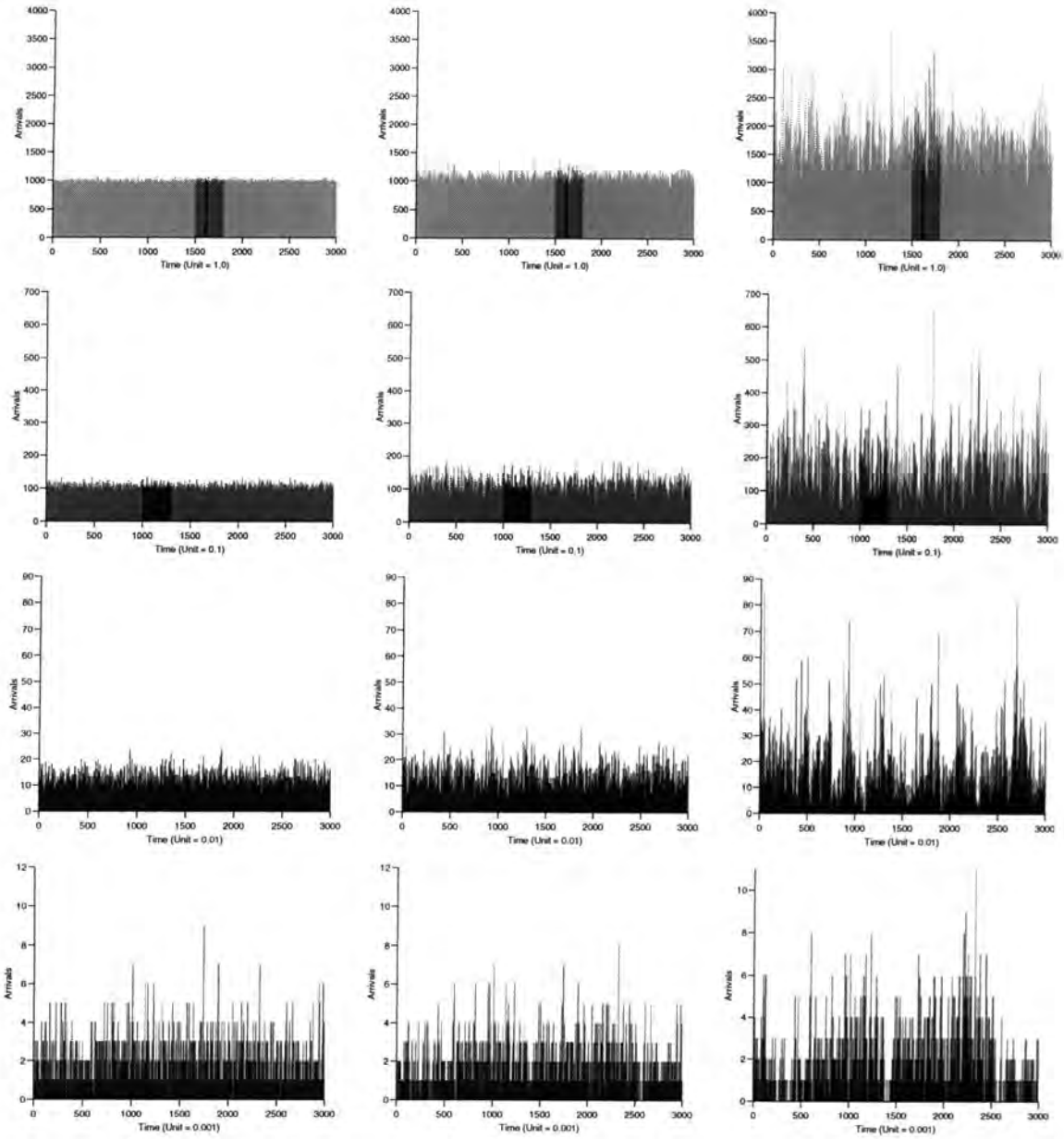
Fig. 6.1 illustrates the generated self-similar traffic. The results displayed left, middle, and right columns use different Hurst parameters of $H = 0.5, 0.7$, and 0.9, respectively. These traffic traces are generated with $\lambda = 1000$ and $\tau = 0.001$. Different grey levels of these diagrams indicate the same segments of traffic on these different time scales. The bottom row of the diagrams shows the outcomes of the simulations. The lower middle row illustrates the 0.01 time unit, i.e., every ten outcomes of the synthetic traces are summed as a single outcome. For different Hurst parameters the diagrams exhibit very different results. In Appendix C the measurement of the burstiness of self-similar traffic generated by the model is discussed.

6.1.3 Gamma Distribution

From the measurement of different traffic sources considerable evidence indicates that the arrival traffic could be modeled by Gamma distribution. Meier-Hellstern *et al* monitored eight ISDN data users [MHWYH91]. The authors found that the interarrival histogram can be modeled by three different distributions within different states. Two of them exhibit gamma phenomenon. Heyman *et al* analyzed a long sequence of real video conference data [HTL92], and discovered that the number of cells per frame is a stationary stochastic process with a gamma distribution rather than a normal distribution. Garrett and Willinger analyzed a two-hour long empirical sample of VBR video [Gar93, GW94]. They revealed that the gamma distribution matches the data very well except in the extreme tail.

A random variable X has a gamma distribution if its probability density function is of form [JK70, Pap91, HTL92, PFTV92, GW94]

$$p_X(x) = \frac{\lambda^s (x-\gamma)^{s-1} \exp[-(x-\gamma)\beta]}{\beta^s \Gamma(s)}, \quad s > 0, \beta > 0, x > \gamma, \lambda > 0. \tag{6.9}$$



(a) $H = 0.5$ (b) $H = 0.7$ (c) $H = 0.9$

Figure 6.1: Synthetic traces generated by the marginal distribution model with exponential distribution on four different time scale.

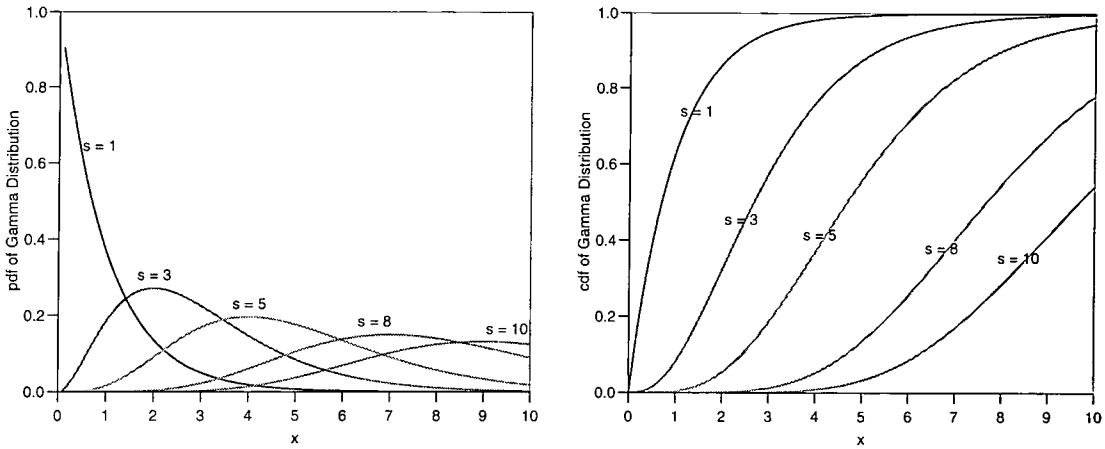


Figure 6.2: The probability density and cumulative distribution functions of gamma distribution with different shape parameters.

The standard form of the distribution is obtained by putting $\beta = 1$ and $\gamma = 0$, given

$$f_\Gamma(x) = \frac{\lambda^s x^{s-1} e^{-x}}{\Gamma(s)} \quad x > 0. \tag{6.10}$$

The exponential distribution can be obtained by setting $s = 1$. The quantity

$$\Gamma(s) = \int_0^\infty t^s e^{-t} dt \tag{6.11}$$

is sometimes called an *incomplete gamma function.* Parameter s and λ are called the *shape* and *scale* parameters, respectively. The cumulative distribution function of gamma distribution is

$$F_\Gamma(x) = Pr[X \leq x] = \frac{\int_0^x t^{s-1} e^{-t} dt}{\Gamma(s)}. \tag{6.12}$$

The mean and variance of the distribution are s/λ and s/λ^2, respectively. When s is an integer, a gamma random variable is the sum of s i.i.d. exponential random variables. In Fig. 6.2 the probability density and cumulative distribution functions are depicted with different shape parameters. The curves with $s = 1$ are the same as the exponential distribution while the others are the aggregated distributions with s independent and identically exponential distributions.

The discrete version of a gamma distribution is negative binomial distribution whose probability density function is defined as [JK92, HTL92]

$$P[X = x] = \binom{k+x-1}{k} p^x q^k, \quad x = 0, 1, 2, \cdots, \tag{6.13}$$

where $0 < p < 1, q = 1 - p$, and $k > 0$. The mean and variance of this distribution are

$$\mu = \frac{k(1-p)}{p} \quad \text{and} \quad \sigma^2 = \frac{k(1-p)}{p^2}. \tag{6.14}$$

When k is an integer, a negative binomial random variable is the sum of k i.i.d. geometric random variables with success probability p.

There is no direct formula to calculate the inverse cumulative marginal distribution of the gamma distribution because no closed form of the gamma function exists . The easiest way is to obtain the

results from its discrete version, the negative binomial distribution. When the cumulative normal function is applied to achieve the probability, the obtained value is the success probability p in the inverse function of the cumulative negative binomial distribution. The scale parameter s is the same as the parameter k in the negative binomial random variable. It can be obtained from the measurements of empirical data, such as $s = 3.006$ for video teleconference traffic, 8 or 6 for ISDN data, 19.75 for interframe VBR video data, or 8.0 for MPEG video data [HTL92, Gar93, GW94].

Similar to the traffic traces depicted in Fig. 6.1, the traffic traces generated by the marginal distribution model with the gamma distribution is demonstrated in Fig. 6.3. Different grey levels indicate the same segments of traffic on the different time scales. All the traces are generated with $s = 3, \lambda = 1000$ and $t = 0.001$ unit of time. The left, middle, and right columns show $H = 0.5, 0.7$, and 0.9, respectively.

6.1.4 Pareto Distribution

Increasing evidence illustrates that many traffic sources are heavy-tailed, i.e., the variance of the observed data is infinity. Pareto distribution follows this type of distribution. In [MHWYH91] the tail part of the gathered ISDN data is modeled by this distribution. The analysis of interframe VBR and MPEG video data found that the extreme tail of the log-log graph of complementary cumulative distribution fits Pareto distribution [GW94]. It also plays an important role both in TELNET packet interarrivals and in the size of FTP data bursts [PF95]. A synthetic trace generated by an ON/OFF source with Pareto ON- and OFF-periods is practically indistinguishable from the real traffic trace, while the traffic generated by the ON/OFF source with the exponential distribution appears in a more or less uniform manner [TWS97]. In addition, the heavy-tailed distribution has been proposed to model telephone call holding times [DMRW94].

Pareto distribution is named after an economist, Vilfredo Pareto, who investigated the distribution of income over a population [JK70, Arn83]. It has been used to model the sizes of asteroids, islands, cities, extinction events, and also flood frequency analysis [Man63, BR95, MRR97a, MRR97b]. The classical form of the distribution is

$$F_P(x) = 1 - (x/k)^{-\alpha}, \quad x > \sigma, \tag{6.15}$$

where the *scale parameter* k and Pareto's *index of inequality* (or known as the *shape parameter*) α are positive. The corresponding probability density function is expressed as

$$f_P(x) = \alpha k^{-\alpha} x^{\alpha-1}. \tag{6.16}$$

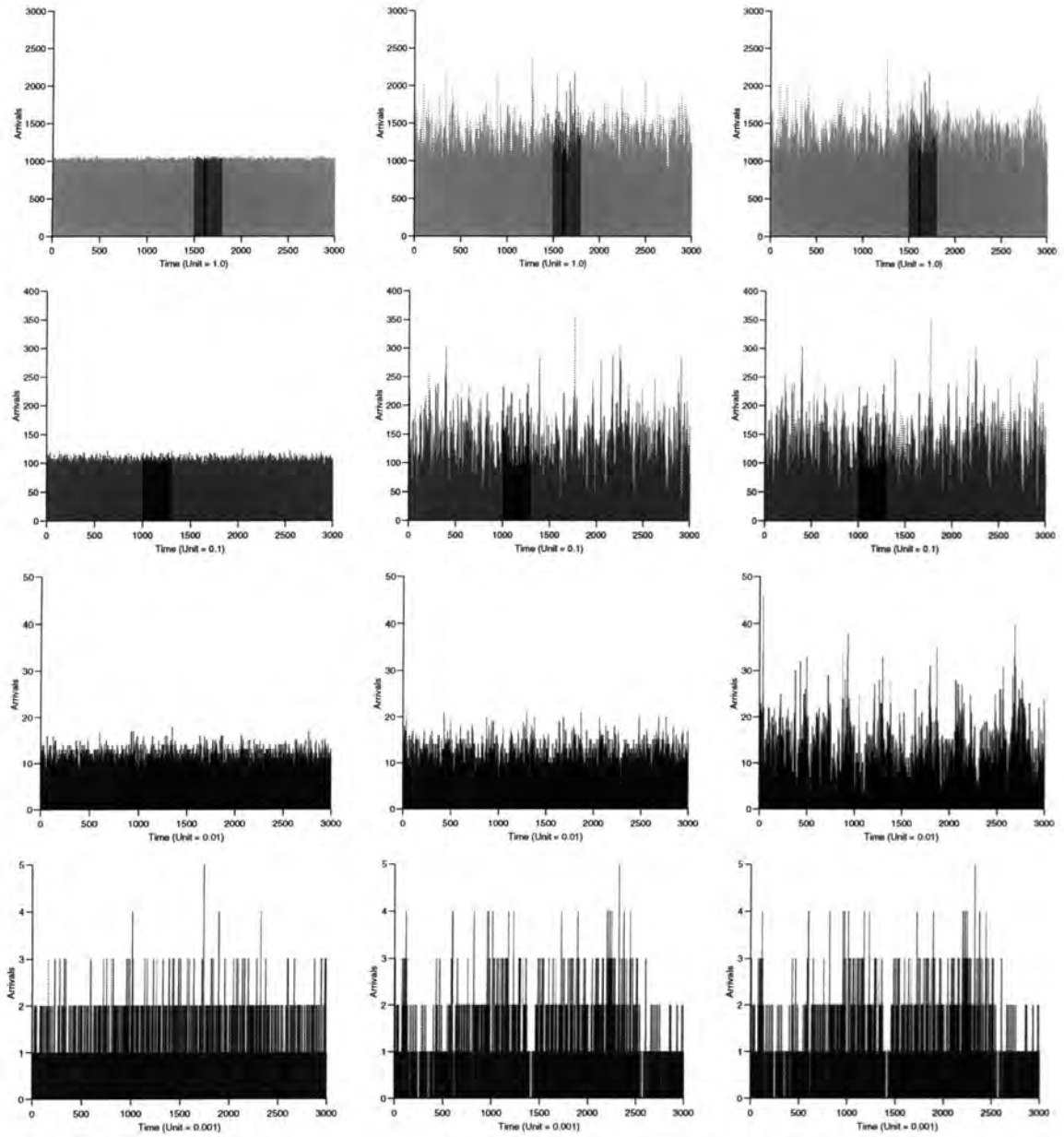
The mean and variance of the Pareto distribution are given by

$$\mu = \alpha k/(\alpha - 1), \quad \text{if } \alpha > 1, \tag{6.17}$$

and

$$\sigma^2 = \alpha k^2 (\alpha - 1)^{-2} (\alpha - 2)^{-1}, \quad \text{if } \alpha > 2 \tag{6.18}$$

Therefore, the distribution has an infinite mean if $\alpha < 1$ and infinite variance if $\alpha < 2$. In fact, this phenomenon coincides with the infinite variance hypothesis that Mandelbrot proposed to account for the



(a) $H = 0.5$ (b) $H = 0.7$ (c) $H = 0.9$

Figure 6.3: Synthetic traces generated by the marginal distribution mode with the gamma distribution on four different time scale.

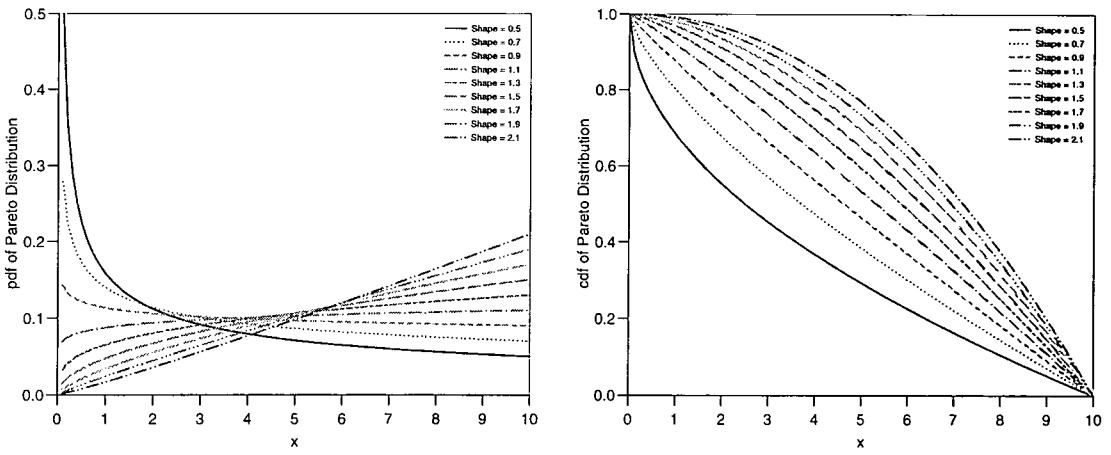


Figure 6.4: The probability density and cumulative distribution functions of the Pareto distribution with different shape parameters.

erratic variability of the sample variances without giving up stationarity [Man69]. Fig. 6.4 illustrates the probability density and cumulative distribution functions. All the curves have a different shape parameter but the same scale parameter, $k = 10$.

Zipf distribution is the discrete version of the Pareto distribution [Fel71, Arn83, JK92]. Its probability function is

$$P[X = x] = \frac{x^{-(\rho+1)}}{\eta(\rho+1)}, x = 1, 2, \cdots, \tag{6.19}$$

where $\eta(\cdot)$ denotes the Riemann zeta function and $\rho > 0$. Very often it is called the *Riemann zeta distribution*. Similar to the Pareto distribution, its moment is infinite if the order of moments is larger than ρ. The use of this distribution has been connected with the counting of letters in linguistics and the number of insurance polices held by individuals. In engineering applications it is used to model the amount of CPU time consumed by an arbitrary process [LO86] and the sizes of ISDN data [MHWYH91].

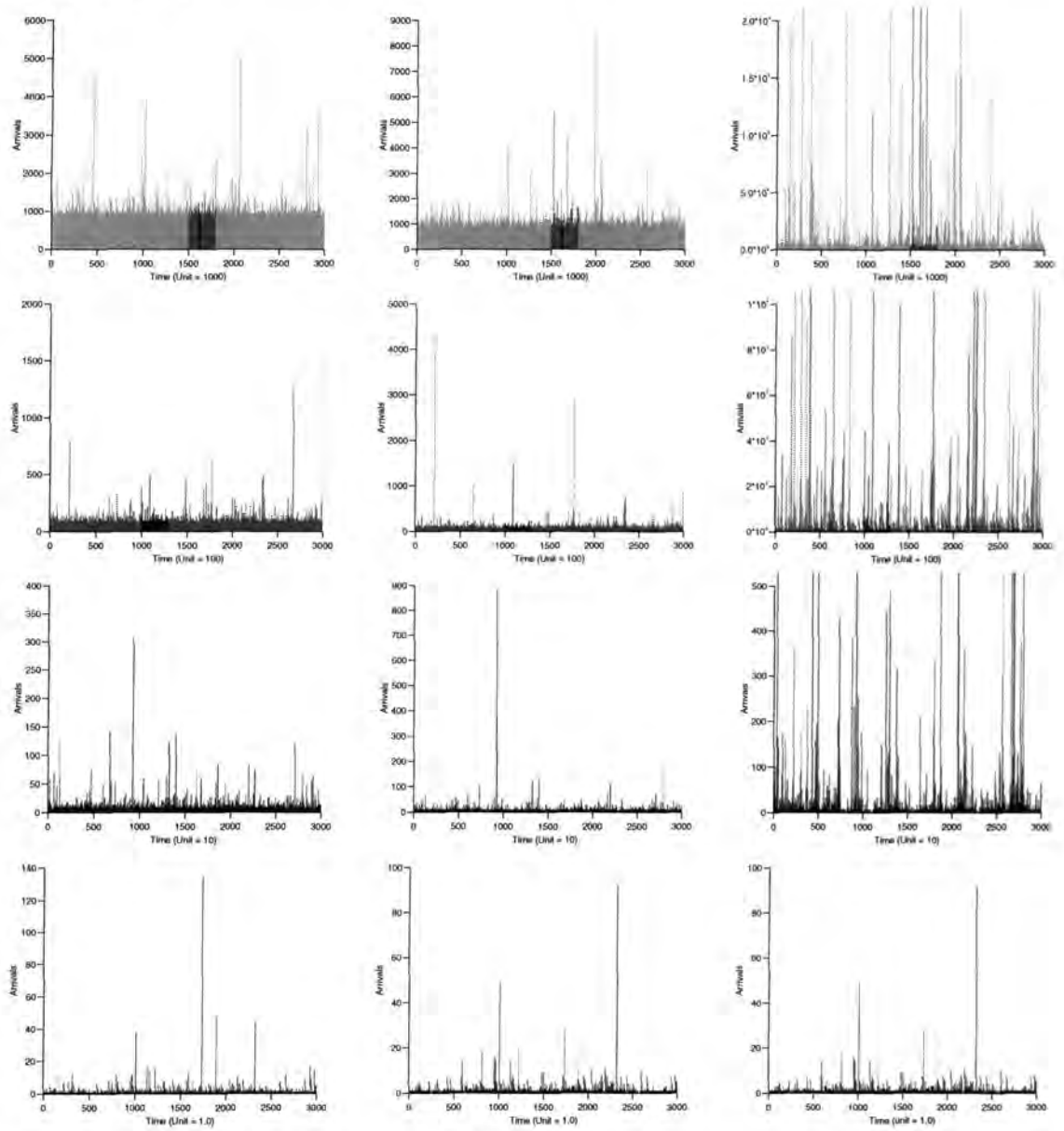
In practice, the measures of inequality can be obtained by several methods [Arn83]. Most of the measures are well defined if the parent population has an infinite mean. The simplest way to measure the shape parameter is to take the survival function of Pareto distribution, i.e., $1 - F_P(x)$. Then, simply applying the logarithm to both sides, the slope of the plotting curve of the logarithmic survival function versus logarithmic x is the scale parameter. A number of measurements of real traffic traces have been undertaken. For instance, the shape parameter of the extreme tail of ISDN data is 0.96 [MHWYH91]. The index of the inequality of interframe video data is 11.2 and the shape parameter of the MPEG video data trace is 0.85 [GW94].

In addition, the Pareto traffic model has been proposed to model Ethernet traffic traces [Gor95, GH97]. While self-similar traffic models represent the block packet count, the Pareto model is used to model the marginal interarrival time distribution. The relationship of the shape parameter and the Hurst parameter can be formulated as

$$H = \begin{cases} (1+\alpha)/2, & 0 < \alpha < 1, \\ (3-\alpha)/2, & 1 < \alpha < 2, \\ 1/2, & \alpha > 2. \end{cases} \tag{6.20}$$

The interarrival times of three Ethernet traffic traces are analyzed to check the fit of the Pareto model. Their shape parameters are 1.16, 1.10 and 1.10, respectively.

As with the two previous distributions, the synthetic traffic trace of the marginal distribution model with the Pareto distribution is shown in Fig. 6.5. Similar to the previous traffic traces, different grey levels indicate the same segments of traffic on the different time scales. The left column illustrates the short range dependent traffic trace while the middle and right column are LRD traffic traces with $H = 0.7$ and 0.9, respectively.



(a) $H = 0.5$ (b) $H = 0.7$ (c) $H = 0.9$

Figure 6.5: Synthetic traces generated by the marginal distribution mode with the Pareto distribution on four different time scale.

6.2 Numerical Results of the Marginal Distribution Model

In the following subsections the effects of the transformation from the FGN samples to self-similar traffic with the marginal distribution model are investigated. The changing of different parameters, such as the Hurst parameter, the shape parameter in gamma and Pareto distributions, will be studied. All the self-similar traffic traces are generated from the FGN samples that are produced by the dFGN algorithm with $M = 1000$ and $n = 30$. With the FGN traces, the distortion of the Hurst parameter in generating FGN traces can be minimized.

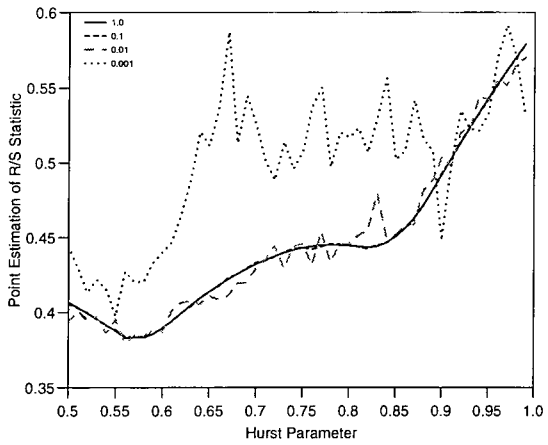
6.2.1 Exponential Distribution

The most attractive property of the exponential distribution is that its mean is associated with the arrival rate and time scale. Therefore, the 'small scale structural assumptions' can be applied easily with the distribution. The results of the estimators, when the change of the Hurst parameter is applied, is displayed in Fig. 6.6. The results using the R/S statistic with $t = 0.0001$ are not available because there are traces in which the slopes of the R/S plots cannot be evaluated (see Table D.12). The curves of the R/S statistic are nonlinear, especially when the time scale is smaller than 0.1. The results using the VT plot exhibit significantly smooth curves. However, for time scales smaller than 0.001 the curve is not as smooth as the others. For both estimators, the results are significantly lower than the target values. The curves of Whittle's MLE exhibit significantly better results although there exists distortions when transferring the traffic traces from the FBM traces. When the time scale decreases as low as $t = 0.0001$, the curve is significantly lower than the target Hurst value as the Hurst parameter increases.

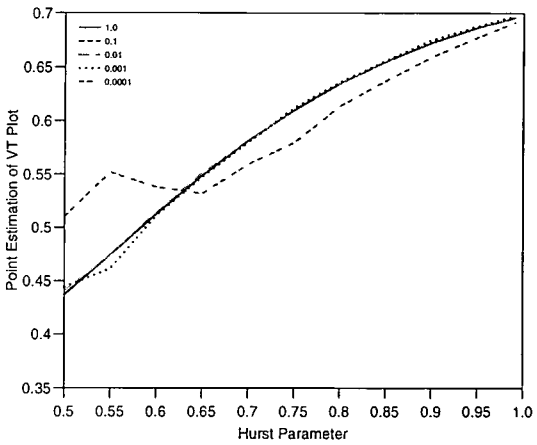
The mean arrival rates of the output of the marginal distribution models with the exponential distribution is depicted in Fig. 6.7. The values and their 95% confidence intervals of the generated traffic traces are listed in Table D.4. All the traffic traces are generated with 1000 arrivals per unit time. With different observed time scales, the confidence intervals increase as the Hurst parameter increases. For the observed time scales, $1.0, 0.1$, and 0.01, the mean rates are reasonably accurate when compared with the input rate, while those of the observed time scale of 0.001 are slightly smaller. All of them are acceptable, although the mean rate of high self-similar traffic traces are about 20% higher than the input rate. However, the mean rate of the smallest observed time scale, 0.0001, is significantly smaller than the input rate. Therefore, the small scale structural assumptions cannot apply to the cases where there is less than one arrival per observed time slot.

6.2.2 Gamma Distribution

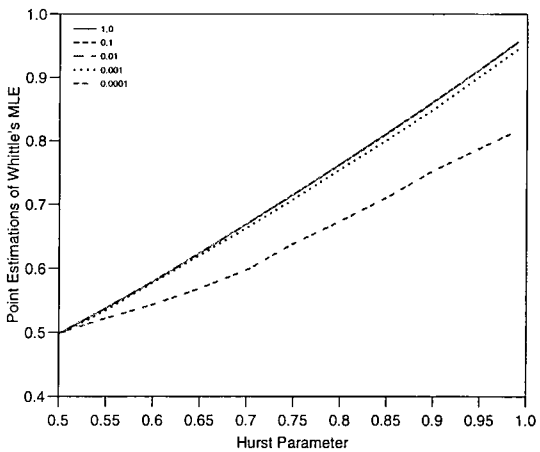
To present the characteristics of network traffic the average arrivals during an observed time slot t can be expressed as st/λ. Thus, the small scale structural assumption can also be applied to the marginal distribution model with the gamma distribution. The arrival rate is 1000 arrivals per unit time. Generating traffic traces with a gamma distribution gives the mean output arrival rates depicted in Fig. 6.8. In this diagram, for the time scale larger than 0.001 the mean arrival rates of different shape and Hurst parameters show no significant differences. For $H = 0.5$ and 0.7, the mean output rates are almost the



(a) The R/S Statistic



(b) The VT Plot



(c) Whittle's MLE

Figure 6.6: The results of the marginal distribution model with the exponential distribution.

same as the input 1000 arrivals per unit time. However, in the case of the highest degree of self-similarity, the rate is about 5% to 10% higher than the input. The bar char shows that the mean arrival rates decrease as the shape parameter increases. All the values shown in Fig. 6.8 and their 95% confidence intervals are recorded in Table D.5. In addition, two more shape parameters, $s = 2$ and 4, also be found in Table D.5.

The relationship of the Hurst parameter and shape parameter is shown in Fig. 6.9. The associated data of part of the diagram is compiled in Table D.10 and Table D.11. In Table D.10, the mean, maximum, and minimum results of the exponential distribution are shown, while those of the gamma distribution with $s = 3$ are recored in Table D.11. The left column of diagrams is generated with $t = 1.0$ while the lower row is with $t = 0.001$. For the R/S statistic, the mean results increase as the shape parameter increases. However, almost all of them are between 0.6 and 0.8. All the 95% confidence intervals are very wide. The confidence interval is shortened as the observed time slot decreases and the shape parameter increases. However, no matter what observed time slot and shape parameter are set, none of the mean results can accurately present the target Hurst parameter.

The mean results using the VT plot show no significant difference between the different shape parameters and observed time slots. In Fig. 6.9 (c) and (d) all the traffic traces with the same Hurst parameter but different shape parameters and observed time scales are very close to the results. However, in both diagrams none of the mean values are over 0.7. For $H = 0.9$ even the confidence intervals are significantly

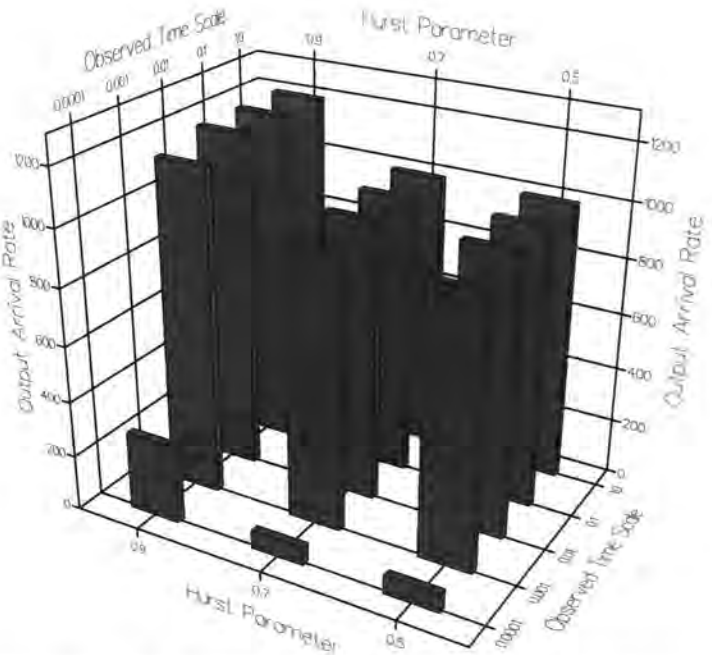


Figure 6.7: Output rate of the marginal distribution model with the exponential distribution.

lower than the target value.

For the results using Whittle's MLE, the confidence intervals are very small, so they can be ignored, see Fig. 6.9 (e) and (f). When the traffic traces are uncorrelated all the mean results are on the reference line, $H = 0.5$. When the Hurst parameter is set as 0.7 and 0.9 the mean values of the larger observed time slot have better results than the other two estimators. For the observed time $t = 1.0$ the values gradually approach to the reference lines when the shape parameter increases. In contrast, the results part from the reference lines when the shape parameter increases for $t = 0.001$.

A selected random seed is used to generate FGN sample traces with different Hurst parameters. Then, transferring the traces to traffic streams with the marginal gamma distribution model with different shape parameters, the arrival rates and results with the three Hurst estimators can be illustrated in Fig. 6.10. All the traffic traces are generated with the observed time slot $t = 1.0$ and 1000 arrivals per unit time. From Fig 6.10 (a) most of the arrival rates are under 1040. Increasing the Hurst parameter results in the arrival rate increasing. However, the shape parameter has the opposite effect. When the shape parameter increases the arrival rate reduces to a lower value.

In Fig. 6.10 (b) all the results using the R/S statistic are less than 0.73. When the shape parameter is near 1, i.e., the exponential distribution, the result carries from 0.4 to 0.58 and its curve exhibits two concave points. For the larger shape parameter the surface exhibits a simpler concave curve. When $s = 5$ the result changes from 0.62 to 0.73 and the minimum value locates near $H = 0.7$. Thus, the curves with different shape parameters are quite flat. Therefore, the R/S statistic could not represent the input Hurst parameters.

The results using the VT plot and Whittle's MLE are illustrated in Fig. 6.10 (c) and (d), respectively. In both cases the shapes are very smooth. In Fig. 6.10 (c) the results with fixed Hurst parameter slightly increase when the shape parameter increases. However, the change is not significant and becomes smaller

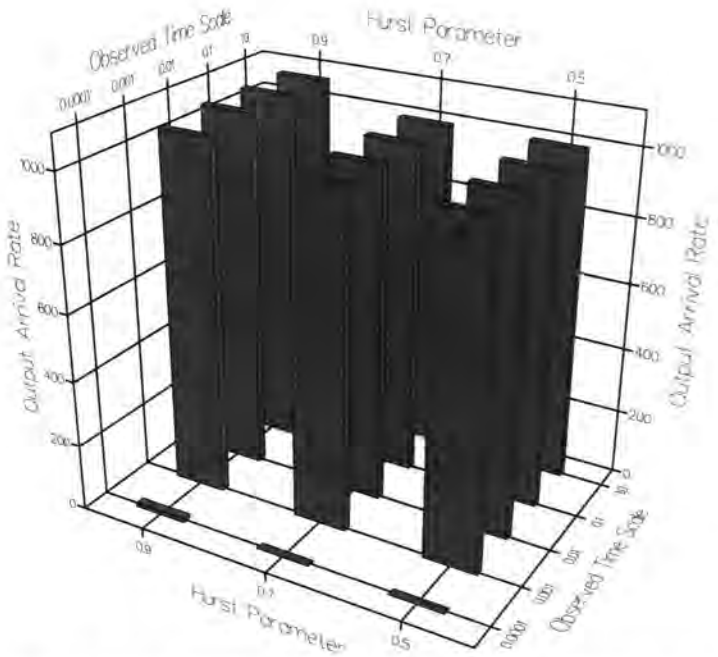


Figure 6.8: Output rate of the marginal distribution model with the gamma distribution with $s = 3$.

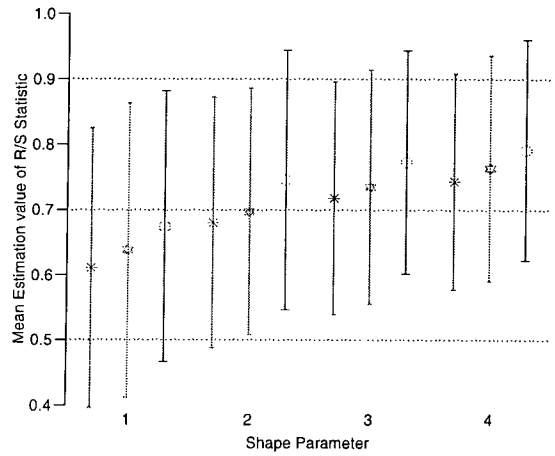
as the Hurst parameter increases. Although the shape of the VT plot is smooth, the results are far less than the target values. The maximum value in this diagram is located on $H = 0.99$ and $s = 1$ with the value 0.7. The results of Whittle MLE approach the target Hurst parameters far better. When fixing the Hurst parameter the result increases as the shape parameter increases. However, the change is not so significant when the traffic traces are uncorrelated or nearly uncorrelated. From Fig. 6.9 (c) and this diagram it is shown that the increase in the shape parameter leads to better results.

Based on the analysis of the marginal distribution model with the exponential and gamma distribution, the R/S statistic is not suitable for the estimation of the traffic traces. The R/S statistic extracts the slope of the pox plot of the range versus the standard deviation. The difference between the maximum and minimum values of traffic traces will highly affect the result of the estimator. In addition, when the Hurst parameter is near 1 the exponential distribution is not appropriate to generate the traffic traces because the output rate will be significantly higher than the setting rate.

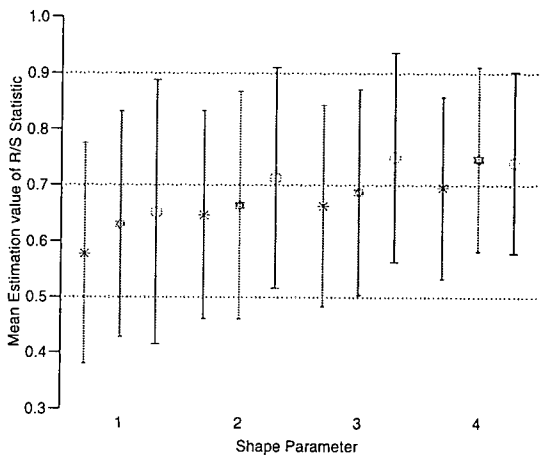
Furthermore, the small scale structural assumption has some limitations. The observed time scale should not be smaller than the arrivals occurring during a time slot. If there is an arrival on average during an observed time slot, the distortion of the Hurst effect will increase with the higher shape parameters. Therefore, to obtain high quality self-similar traffic streams an appropriate observed time scale must be chosen in which there are at least one arrival per observed time slot.

6.2.3 Pareto Distribution

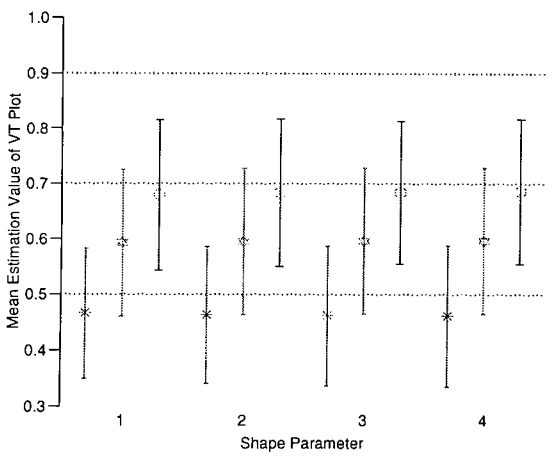
The Pareto distribution is a heavily-tailed distribution. Its mean and variance exist only if its shape parameter is greater than 1 and 2, respectively. It is not appropriate to be incorporated with the small scale structural assumption. In theory, the mean and variance of this distribution is expressed in Eqs. (6.17) and (6.18). However, in practice the simulation results vary with the shape parameter. In Table 6.1 the



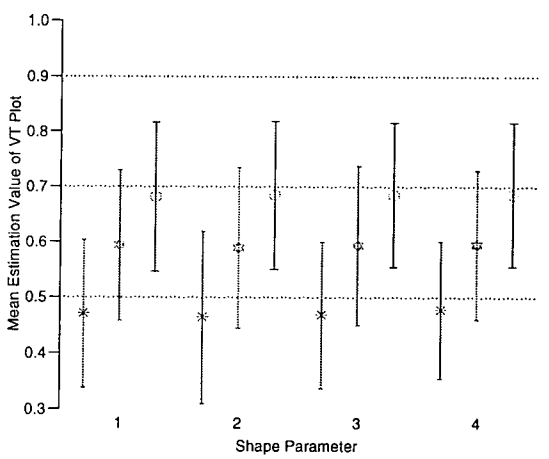
(a) The R/S Statistic ($t = 1.0$)



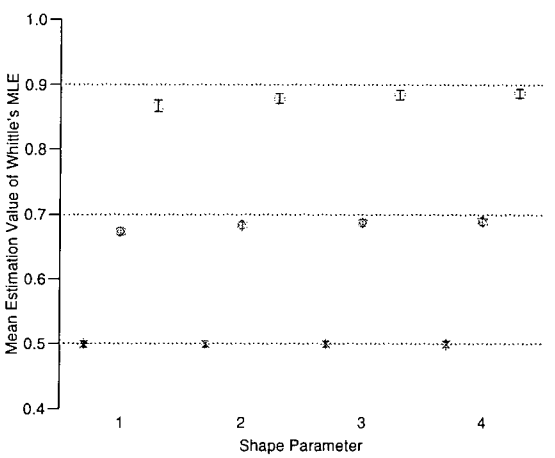
(b) The R/S Statistic ($t = 0.001$)



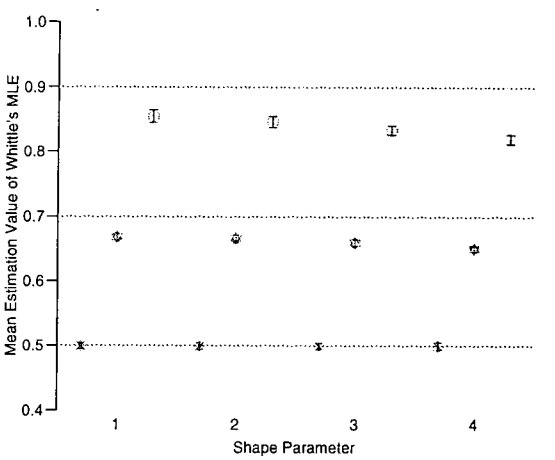
(c) The VT Plot ($t = 1.0$)



(d) The VT Plot ($t = 0.001$)

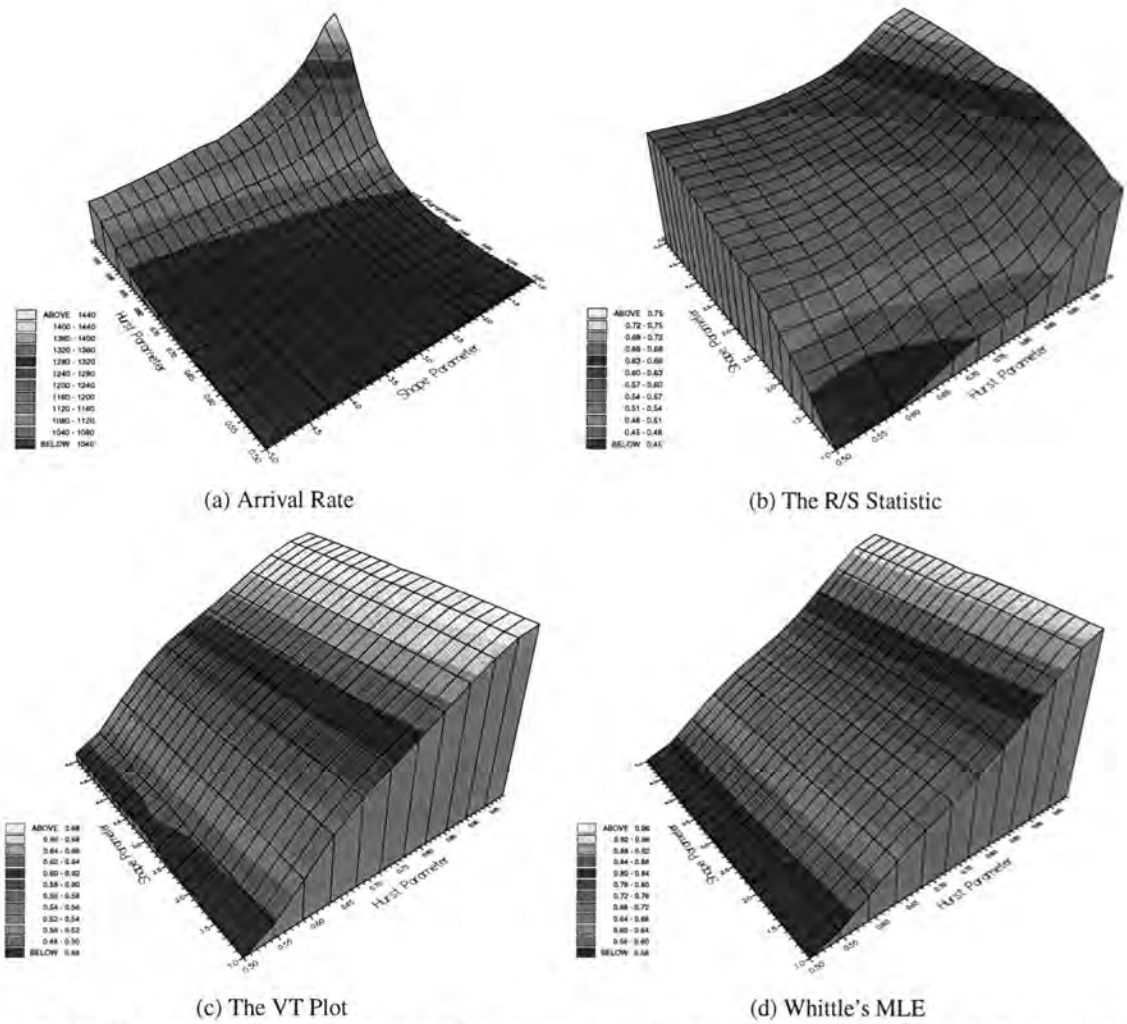


(e) Whittle's MLE ($t = 1.0$)



(f) Whittle's MLE ($t = 0.001$)

Figure 6.9: The mean Hurst values of the marginal distribution model with the gamma distribution.



(a) Arrival Rate

(b) The R/S Statistic

(c) The VT Plot

(d) Whittle's MLE

Figure 6.10: The arrival rate and estimated Hurst values of the marginal gamma distribution model with different Hurst parameters and shape parameters.

combination of the scale and shape parameters are related to the unit arrival rate. For different Hurst parameters the output rate of the traffic traces is highly dependent on the shape and Hurst parameters. For $H = 0.5$ and 0.7 the rate increases as the shape parameter increases but it decreases as $H = 0.9$. None of the output rates of the different shape and scale settings fit well with the theoretical mean rate. Therefore, investigating this model is not appropriate to set a fixed arrival rate. All the traffic traces in the following study are generated with a fixed scale parameter $k = 10$.

The relationship between the results and the shape parameter is demonstrated in Fig. 6.11. From these three diagrams it is apparent that all the curves are non-tractable. In Fig. 6.11 (a) the results using the R/S statistic with $\alpha = 1.5$ and 2.5 have peak values near $H = 0.7$. When the Hurst parameter is near 1 both of the results are lower than 0.2. For $\alpha = 0.5$ the estimate results are almost under 0.2. The results using the VT plot are very low as well. The results using the VT plot are significantly lower than their target Hurst values. All the curves in Fig. 6.11 (b) fluctuate considerably, thus they do not convey any reasonable information. The results using Whittle's MLE in Fig. 6.11 (c) with $\alpha = 0.5$ exhibits no correlation when the Hurst parameter is lower than 0.8. All of the results are close to 0.8 when the Hurst parameter is near 1. From these three diagrams none of the results can represent the original Hurst

Shape Parameter	Scale Parameter	Output Rate		
		H = 0.5	H = 0.7	H = 0.9
1.2	1/6	0.748 ± 0.29	0.738 ± 0.29	25.78 ± 72.2
1.6	3/8	0.959 ± 0.042	0.956 ± 0.057	3.308 ± 3.04
2.0	1/2	1.232 ± 0.015	1.231 ± 0.026	1.965 ± 0.51

Table 6.1: Output rate of marginal Pareto distribution model with fixed input rate.

Shape Parameter	Ideal Rate	Hurst Parameter		
		0.5	0.7	0.9
0.5	∞	$271x10^3 \pm 147x10^3$	$247x10^3 \pm 171x10^3$	$501x10^6 \pm 271x10^6$
1.5	30	29.63 ± 1.90	29.52 ± 2.03	141.5 ± 174.7
2.5	16.67	16.65 ± 0.15	16.64 ± 0.41	22.17 ± 3.29

Table 6.2: Output rate of the marginal distribution model with the Pareto distribution.

settings.

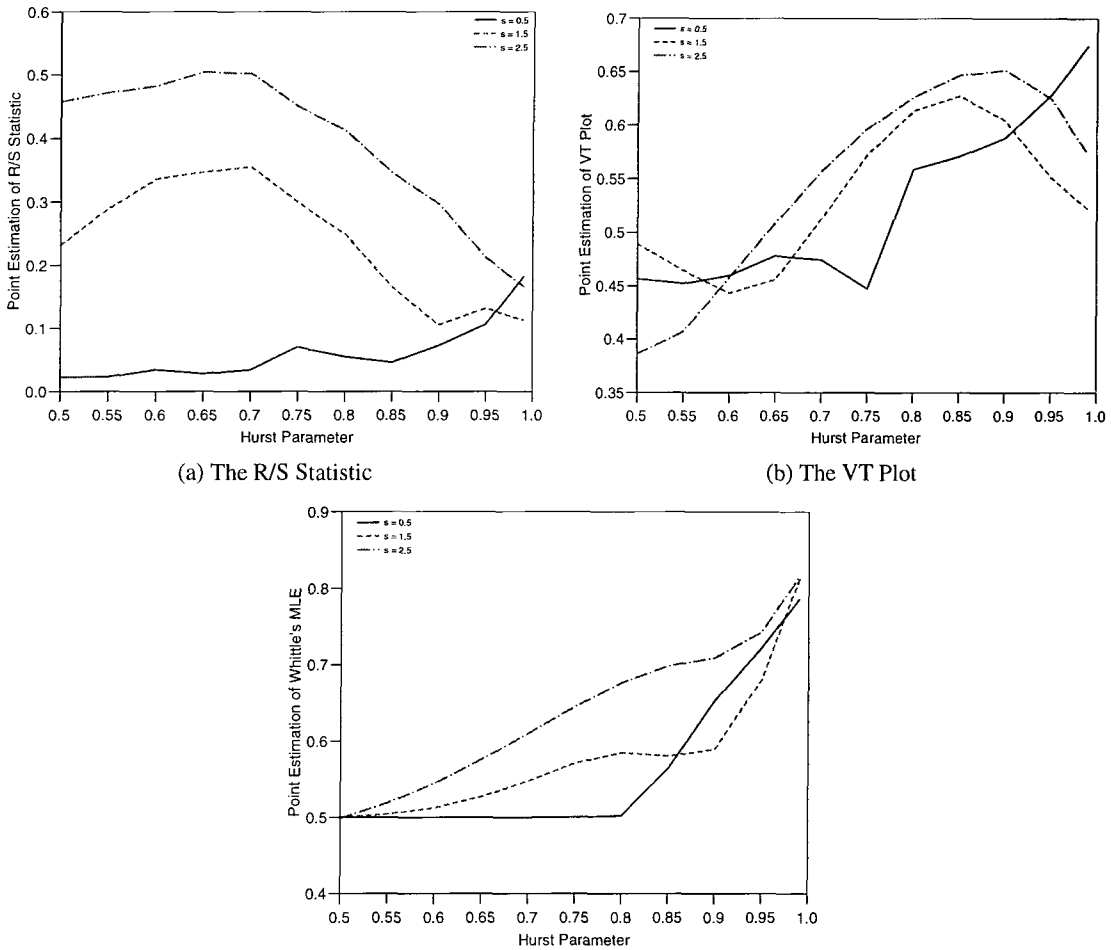
After examining the relation of the shape and Hurst parameters the mean output rate of the traffic traces are listed in Table. 6.2. The second column lists the theoretical mean of the traffic traces. For the shape parameter $\alpha = 0.5$ the theoretical mean value is infinity. The first row of the output rate is very high and in addition their confidence intervals are very large. Thus, it matches the theoretical assumption well. When $\alpha = 1.5$ and 2.5 the output rate of $H = 0.5$ and 0.7 are very close to the ideal values. However, for $H = 0.9$ the output rate are larger than the expected values. Especially, when $\alpha = 1.5$ the mean output rate is almost five times the ideal value.

This analysis shows that the marginal distribution model with the Pareto distribution is not suitable for generating the self-similar traffic traces. Although the FGN traces show high quality of self-similarity, the output traffic streams are highly distorted by the transformation. Even the results using Whittle's MLE and the target Hurst parameters are significantly different.

As with previous study, twenty different values of random seed are applied to examine the statistical robustness of the results with three Hurst estimators. The mean results and their 95% confidence intervals are illustrated in Fig. 6.12. All the symbols of the star, asterisk, and hexagon are the mean Hurst results. The barlines on top of the symbols are their 95% confidence intervals. The three dotted lines in each graph are the target Hurst parameters, $H = 0.5$, 0.7, and 0.9, respectively. The associated data is compiled in Table D.12.

For the R/S statistic all the mean results fail to represent actual Hurst values. The mean results with $\alpha = 0.5$ and 1.5 are significantly lower than the desired Hurst values. The mean results with $\alpha = 0.5$ are very close to zero as the Hurst parameters are 0.5 and 0.7. Although the mean results with $\alpha = 2.5$ show a better result, the mean values are significant higher and lower than the target Hurst parameters, $H = 0.5$ and 0.9, respectively.

The mean results using the VT plot in Fig. 6.12 (b) are significantly lower than the target Hurst values as the samples are not uncorrelated. The confidence intervals with $H = 0.9$ and $\alpha = 01.5$ are significantly



(a) The R/S Statistic

(b) The VT Plot

(c) Whittle's MLE

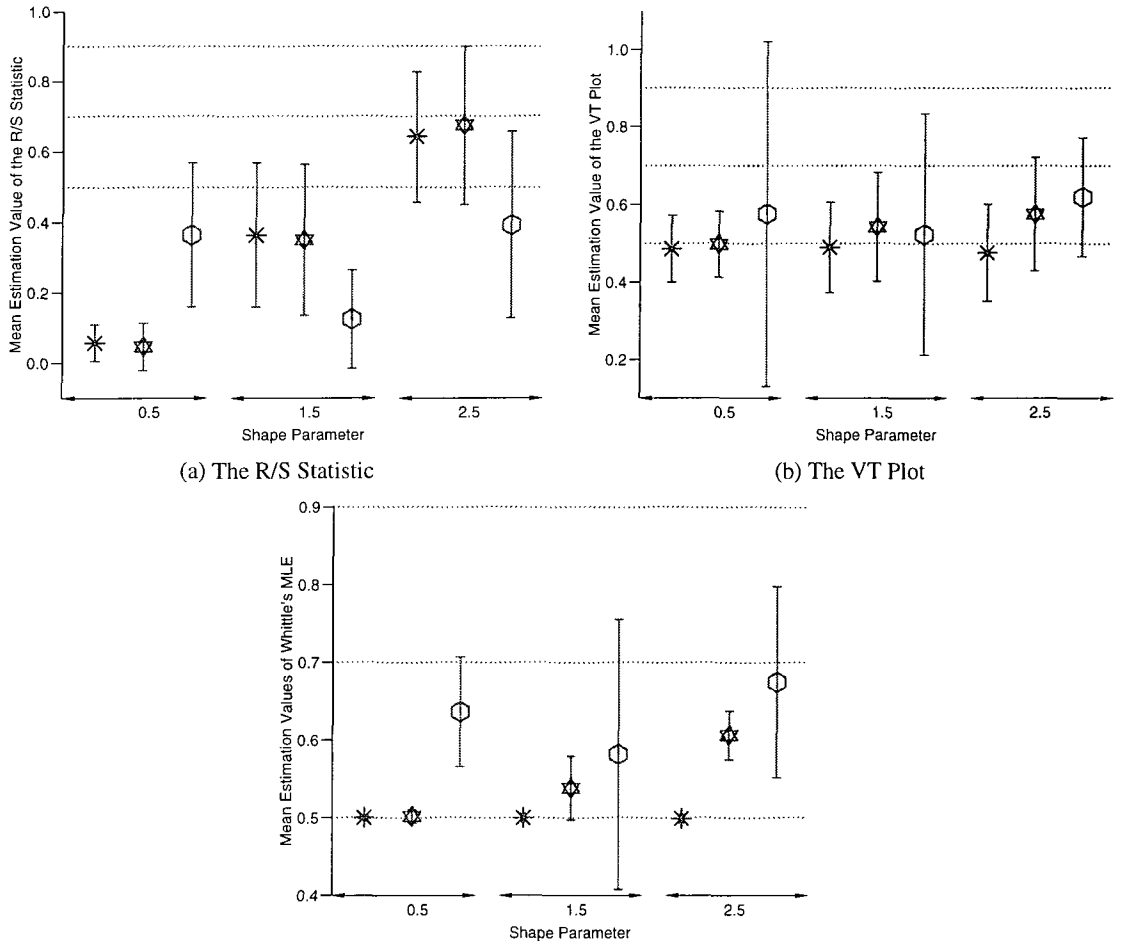
Figure 6.11: The results of the marginal distribution model with the Pareto distribution.

large. It almost covers the whole range from 0 to 1. Although the mean values of $H = 0.5$ are very close to the target Hurst value, the other results suggest that the VT plot cannot accurately represent the actual Hurst values.

The mean results using Whittle's MLE are depicted in Fig. 6.12 (c). All the mean results are significantly lower than the desired Hurst values, except in the case of $H = 0.5$. The confidence intervals of $H = 0.9$ are significantly large. It has been shown that Whittle's MLE is the most accurate estimator. Therefore, from this study it is apparent that the model cannot generate accurate traffic traces.

6.3 The Storage Model

After revealing the fractal nature of bursty data, a number of research studies have been undertaken to transform the self-similar stochastic process to self-similar traffic. Although the marginal distribution model is mathematically perfect, it does not convey any physical interpretation. The difficulties in transforming the self-similar stochastic process to self-similar traffic is due to the characteristic of the traffic stream being a counting process rather than a continuous-time stochastic process. The incorporation of a fluid flow model was first proposed by Norros in [Nor94] in which the arrival traffic consists of fluid bursts with Poisson arrivals. Based on Norros' proposal, more precise models, mainly collabo-



(a) The R/S Statistic

(b) The VT Plot

(c) Whittle's MLE

Figure 6.12: The mean Hurst values of the marginal distribution model with the Pareto distribution.

rating with the ON/OFF models, have been proposed to interpret the self-similar phenomenon in data traffic [WTSW95, Pru95, BRSV96, HRS96, WTSW97].

The following derivation is almost entirely based on the work having been done in [WTSW97] and [TWS97]. First, suppose that there is a single source that emits a burst occasionally. Let $\{W(t), t \geq 0\}$be the workload process of the source. In the ON period, there is a burst generated by the source while there is nothing emitted during the OFF period. The workload process is a stationary binary time series, i.e., $W(t)$ is equal to 1 or 0 according to whether the source is in the ON or the OFF period. The length of the ON and OFF periods follow independent distributions and might have different distributions. The ON/OFF model is a strict ON/OFF process, i.e., the OFF period must be followed by an ON period, and vice versa.

Suppose that there are M i.i.d. sources. Let $W^{(m)}(t)$ be the workload process of the m^{th} source. The superposition or cumulative counting process at time t is $\sum_{m=1}^{M} W^{(m)}(t)$. The aggregated cumulative arrival counts in the interval $[0, Tt)$ can be expressed as

$$W_M(Tt) = \int_0^{Tt} (\sum_{m=1}^{M} W^{(m)}(u))du, \tag{6.21}$$

where T is a factor to gain the scaling time. Based on the findings of the self-similar nature of Ethernet traffic the distributions of the ON/OFF periods must have *infinite variance syndrome* (the 'Noah Ef-

fect') or *heavy-tailed behavior* when $T \to \infty$ and $M \to \infty$ [Man69, TWS97]. Evidence indicates that this hypothesis of the derivative is appropriate.

The analysis of token-ring traffic shows that the distribution of interarrival times is highly skewed to the right (i.e. with a long tail on the right) [JR86]. The ISDN packet traffic has an extreme tail that can fit with Pareto distributions with infinite mean [MHWYH91]. In [DMRW94], the analysis of CCSN/SS7 data traffic indicates that the call holding time is a heavy-tailed distribution. In [PF95], the analysis of 24 wide-area traces claims that the Poisson-based processes fail to model them due to the heavy-tailed behavior. The file size and transmission times of the World Wide Web are due to heavy-tailed distributions [CB96, CTB97]. In addition, a series of papers reveal that Ethernet traffic or LRD VBR video data use stochastic processes with slow decaying variance to model the measuring datasets [LTWW93, LTWW94, BSTW95, Pru95, Wil95, EGW94, ENW96, EPW95, LWTW94, WTLW95].

According to the observed evidence let $f_1(x), F_1(x)$, and $\overline{F}_1(x) = 1 - F_1$ be the probability density, cumulative distribution, and survival functions, respectively, of an ON period, respectively. Similarly, $f_2(x), F_2(x)$, and $\overline{F}_2(x)$ correspond to an OFF period. μ_1, σ_1^2, μ_2 and σ_2^2 are the mean lengths and variances of the ON and OFF periods. As a heavy-tailed distribution, assume that

$$\text{either } \overline{F}_1(x) \sim l_1 x^{-\alpha_1} L_1(x) \text{ with } 1 < \alpha_1 < 2 \text{ or } \sigma_1^2 < \infty$$

and

$$\text{either } \overline{F}_2(x) \sim l_2 x^{-\alpha_2} L_2(x) \text{ with } 1 < \alpha_2 < 2 \text{ or } \sigma_2^2 < \infty$$

where $l_j > 0$ is a constant and $L_j(x) < 0$ is a slowly varying function at infinity, i.e., $\lim_{x\to\infty} L_j(tx)/L_j(x) = 1$ for any $t > 0$. Therefore, $L_j(x)$ could be asymptotic or a constant, $\log x$, $(\log x)^{-1}$, etc. The mean lengths μ_1 and μ_1 must be finite but the variance σ_1^2 and σ_2^2 are infinite when $\alpha_j < 2$. Therefore, F_1 and F_2 could have a Pareto distribution in which the index j is 1 for an ON period, and 2 for an OFF period.

For convenience, let $a_j = l_j(\Gamma(2-\alpha_j))/(\alpha_j)$ when $1 < \alpha_j < 2$. For the case of $\sigma_j^2 < \infty$, let $\alpha_j = 2, L_j = 1$, and $a_j = \sigma_j^2$. Let

$$\Lambda = \lim_{t\to\infty} t^{\alpha_2-\alpha_1} \frac{L_1(t)}{L_2(x)} \tag{6.22}$$

Let σ_{lim}^2 be the normalized factor which can be expressed as

$$\sigma_{\text{lim}}^2 = \begin{cases} \frac{2(\mu_2 a_1 \Lambda + \mu_1^2 a_2)}{(\mu_1+\mu_2)^3 \Gamma(4-\alpha)}, & \text{if } 0 < \Lambda < \infty, \\ \frac{2\mu_1 a_2}{(\mu_1+\mu_2)^3 \Gamma(4-\alpha_{\min})}, & \text{if } \Lambda = 0, \\ \frac{2\mu_2 a_1}{(\mu_1+\mu_2)^3 \Gamma(4-\alpha_{\min})}, & \text{if } \Lambda = \infty, \end{cases} \tag{6.23}$$

where $\alpha = \alpha_1 = \alpha_2$. Similarly, the limiting constant $L = L_2$ if $0 < \Lambda < \infty$, and $L = L_1$ if $\Lambda = \infty$ or if $\Lambda = 0$. The derivative leads to the following conclusion [WTSW97]:

> For large M and T, the aggregated cumulative arrival process $\{W_M(Tt), t > 0\}$ behaves statistically like
>
> $$TM\frac{\mu_1}{\mu_1+\mu_2}t + T^H\sqrt{L(T)M}\sigma_{\text{lim}}B_H(t) \tag{6.24}$$

where $H = (3 - \alpha_{min})/2$. More precisely,

$$\mathcal{L}\lim_{T\to\infty} \mathcal{L}\lim_{M\to\infty} \frac{(W_M(Tt) - TM\frac{\mu_1}{\mu_1+\mu_2}t)}{T^H L^{1/2}(T) M^{1/2}} = \sigma_{\lim} B_H(t) \tag{6.25}$$

where $\mathcal{L}\lim$ means convergence in the sense of finite-dimensional distributions.

Instinctively, Eq. (6.24) is composed of two parts: the mean level and the fluctuations. The mean level decided by $TM(\mu_1/(\mu_1+\mu_2))t$ provides the main contribution for large T and M. It determines the 'quantity' of the traffic trace. The fluctuations, involved with the fractional Brownian motion $\sigma^2_{\lim} B_H(t)$ scaled by a lower order factor $T^H L^{1/2}(T) M^{1/2}$, control the 'quality' part of the traffic trace. The storage model has been pictorially proved in Fig. 2.2 (c) which is illustrated by 500 ON/OFF sources with $\alpha_1 = \alpha_2 = 1.2$ simulated on a parallel processing machine with 16384 processors [WTSW95, WTSW97, TWS97].

Although the above derivative covers the gap between the counting processes and self-similar traffic models, it involves handling too many unknown parameters, such as how large T and M should be, what the slowly varying functions $L_j(x)$ are, etc. In [TG97], it has been shown that $L_j(x)$ is a constant rather than a slowly decaying function. Therefore, a Pareto distribution might be the best candidate to model the ON/OFF sources because of its heavy-tailed property. However, it is still too complicated to handle.

Next, the analysis provides a very attractive model that can be used to analyze the queueing problems. From the simulation aspect a simpler model that does not involve so many unknown factors requires to reviewing. In [Nor94, Nor95b, Nor95a], Norros proposed a storage model which could interpret the physical meaning of self-similar traffic models. With a fractional Brown motion $B_H(t)$, a stationary storage process can be expressed as

$$A(t) = mt + \sqrt{am} B_H(t), \quad t \in (-\infty, \infty) \tag{6.26}$$

The model involves three parameters, the mean rate, the variance coefficient, and the Hurst parameter, m, a, and H, respectively.

Norros interpreted the storage model as a fluid burst model in which the burst length has infinite variance but the interarrival times follow an exponential distribution with arrival rate λ. Similar to the storage model incorporated with the heavy-tailed distribution ON/OFF model the first part of Eq. (6.26) characterizes the quantity of the traffic while the second part involved with H and a characterizes the quality of the traffic.

In this model the only unknown parameter is the variance coefficient. From the observation of the fluid burst model, Norros claimed that a is independent of λ. When $H = 1/2$ the variance coefficient is the index of dispersion (variance over mean) of $A(t)$ for large t. When $H > 1/2$ the variance coefficient is determined by three factors: r^{2H-1} giving an interesting expression for a's dependence on the burst transmission rate r, a factor of the form C^{2H} where C is a constant, and the truncated distribution of the burst size. However, when comparing Eqs. (6.24) and (6.26) it is more appropriate to interpret the storage model as aggregation of many ON/OFF sources with heavy-tailed distributions. Therefore, the quantity parts can be obtained as $m = (\mu_1/(\mu_1+\mu_2))M$. For the quality part the variance coefficient can

be treated as

$$am \sim T^{2H} ML(T)\sigma_{\text{lim}}^2. \tag{6.27}$$

Therefore, a is determined by the number of sources M, the time factor T, the limiting constant L, and the normalization factor σ_{lim}^2/m.

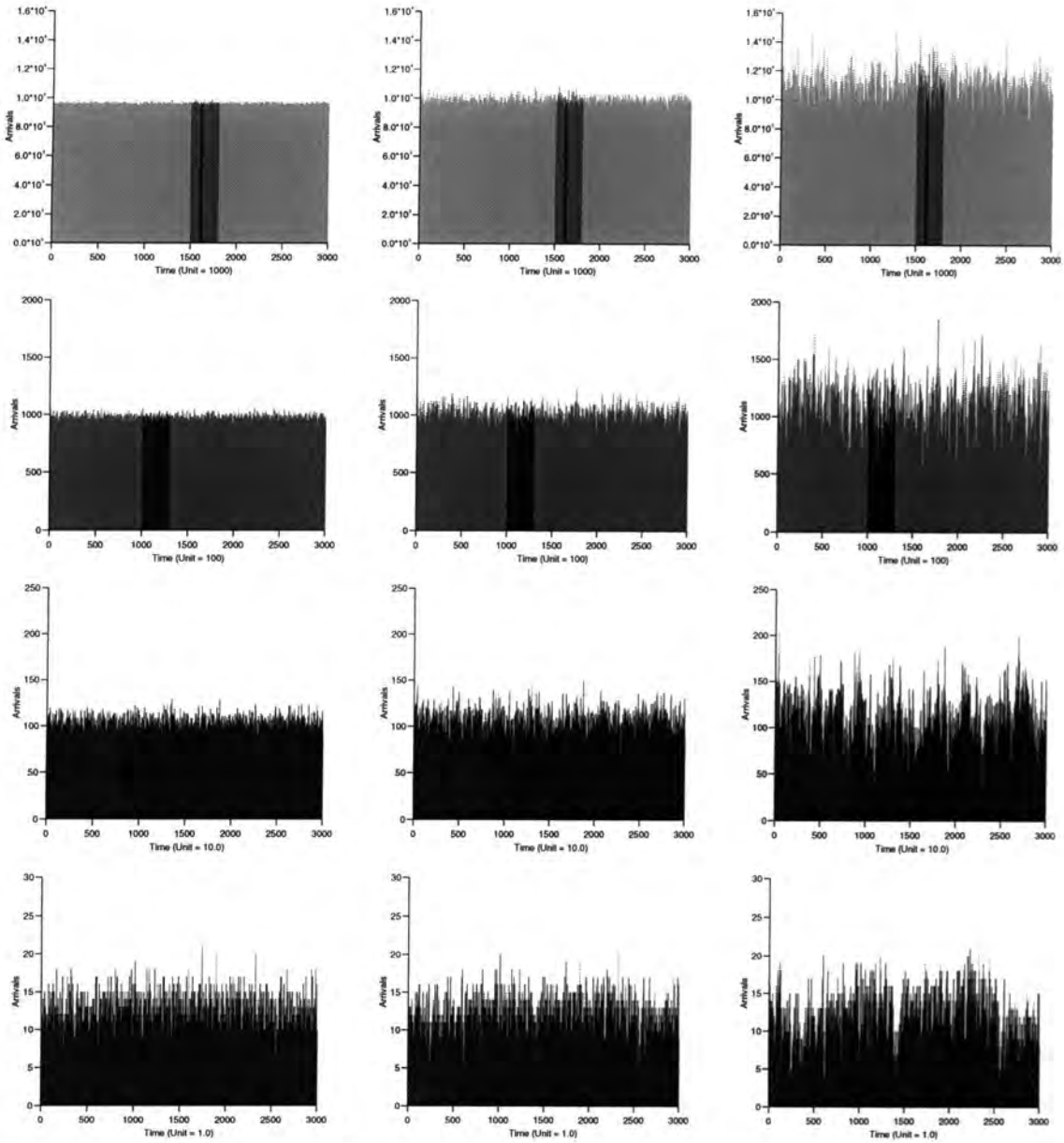
Fig. 6.13 illustrates the generated self-similar traffic. The results displayed left, middle, and right columns use different Hurst parameters of $H = 0.5, 0.7$, and 0.9, respectively. These traffic traces are generated with $\lambda = 10$ and $a = 1$. Different grey levels of these diagrams indicated the same segments of traffic on these different time scales. The bottom row of the diagrams shows the outcomes of the simulations. The lower middle row illustrates the 0.01 time unit, i.e., every ten outcomes of the synthetic traces are summed as a single outcome. For different Hurst parameters the diagrams exhibit very different results.

6.4 Numerical Results of the Storage Model

The following simulations of the storage model are based on Eq. (6.26). Although Eq. (6.24) has more rigorous mathematical derivation, there are several unknown factors, such as $L(\cdot)$, which make it too complicated to handle. However, it is very useful for analysis in order to understand the queueing behavior or the network performance. Here, the major concern is to generate a high quality of self-similar traffic traces that can be applied to network simulations.

To generate the self-similar traffic traces the arrival rate is set as 1000 arrivals per unit time; i.e, $m = 1000$. All the FBM traces are generated by the dFGN algorithm with $M = 1000$ and $n = 30$. The choice of the FBM traces is due to reducing the effect of the inaccuracy of the Hurst estimation. The output rate of the traffic traces generated by the storage model with the above settings is illustrated in Fig. 6.14. Table D.6 compiles the numerical data of the diagrams in accompany with their 95% confidence intervals. There are five different variance coefficients, $a = 200, 400, 600, 800$, and 1000. As expected increasing the variance coefficient causes an increase in the output rate, also the higher the Hurst parameter, the larger the output rate. These results are due to the truncation of negative counts of arrivals.

Five different variance coefficients are applied to examine the Hurst values of the traffic traces in Fig. 6.15. In this figure the mean results and their 95% confidence intervals are depicted with different variance coefficients $a = 200, 400, 600, 800$, and 1000. The associated data of this diagrams is compiled in Table D.13. With the same Hurst parameter the results using the R/S statistic gradually decrease as the variance coefficient increases. However, the different Hurst parameters do not change when the variance coefficient is fixed. In addition, the confidence intervals of the results using the R/S statistic are very wide and become even wider as the variance coefficient increases. In Fig. 6.15 (b) and (c) the results using the VT plot and Whittle's MLE are not affected by the change of the variance coefficient at $H = 0.5$. When $H = 0.7$ and 0.9 the results gradually decrease as the variance coefficient increases. However, the change is so small that can be ignored.



(a) $H = 0.5$ (b) $H = 0.7$ (c) $H = 0.9$

Figure 6.13: Synthetic traces generated by the storage mode on four different time scale.

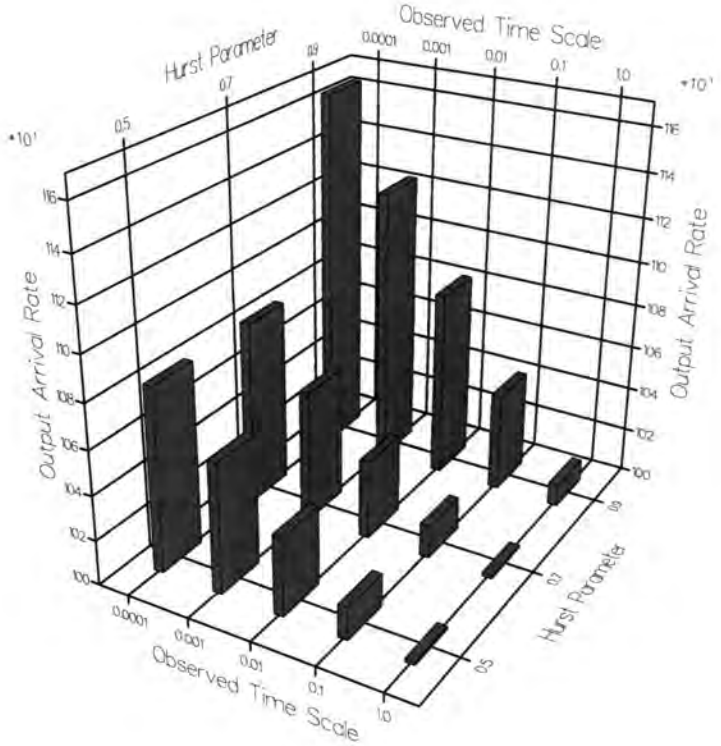
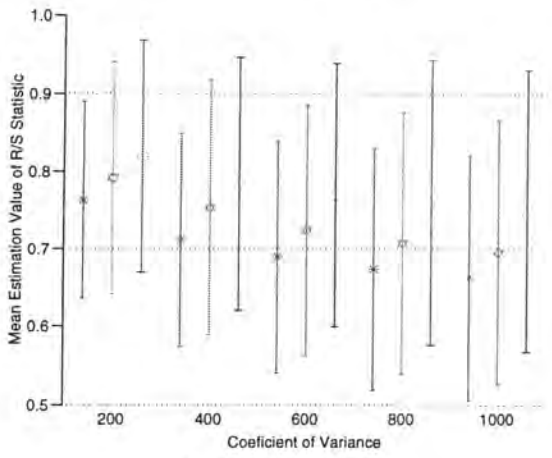
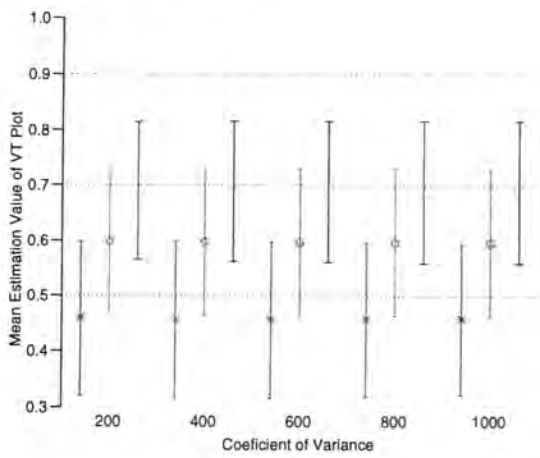


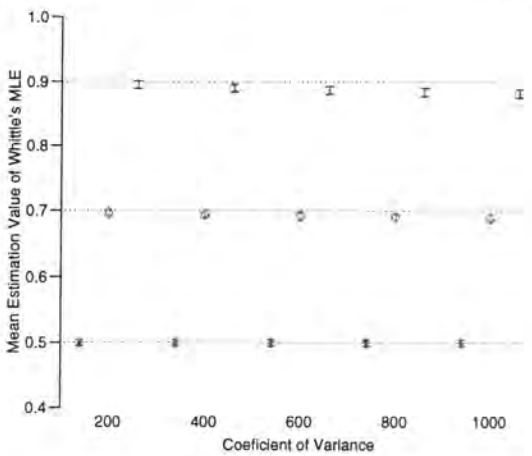
Figure 6.14: Output rate of the storage model.



(a) The R/S Statistic



(b) The VT Plot



(c) Whittle's MLE

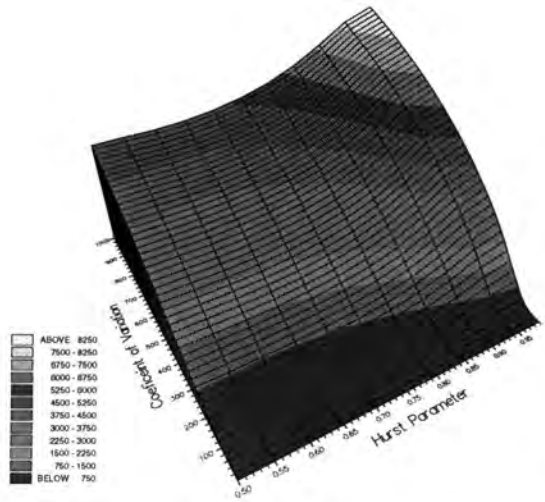
Figure 6.15: The mean results of the storage model.

The effect of the variance coefficient and Hurst parameter in the storage model is illustrated in Fig. 6.16. All the traffic traces analyzed in this figure are generated by the dFGN algorithm with the same random seed but different Hurst parameters and variance coefficients. The first diagram depicts the count of truncated events, i.e., the negative number of arrivals in 32768 samples. The counts increase as the Hurst parameter and the variance coefficient increases. When $H = 0.99$ and $a = 1000$ there are 8559 negative arrivals. The counts of events with negative arrivals causes the output rate of the traffic traces to be considerably higher than the input rate in Fig. 6.16 (b). The highest output rate, 1313.29, occurs at $H = 0.99$ and $a = 1000$.

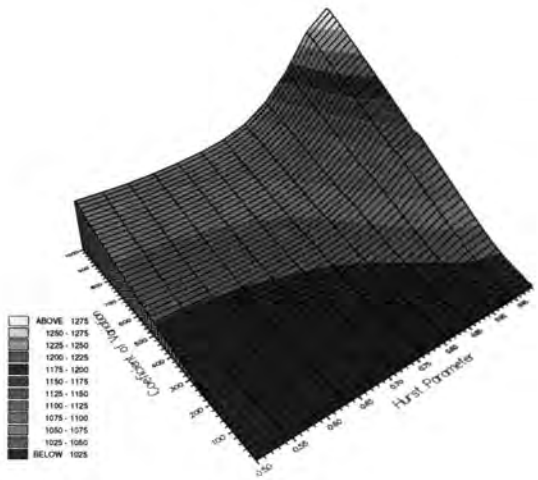
The results using the R/S statistic, the VT plot, and Whittle's MLE are depicted in Fig. 6.16 (c), (d), and (e), respectively. The shape of the results using the R/S statistic exhibit a nonlinear surface. For the variance coefficient less than 40 the results using the R/S statistic are very high and do not change as the Hurst parameter changes. For higher Hurst parameters changing the variance coefficient exhibits nonlinear curves. However, for higher variance coefficients changing the Hurst parameter exhibits more tractable curves. In comparison to the results using the R/S statistic those using the VT and Whittle's MLE are far smoother. In spite of being smooth, the results using the VT plot do not fit the target Hurst parameters. The results increase as the variance coefficient decreases. The values using Whittle's MLE exhibit significantly better results because they fit the ideal Hurst values well. The changing of the variance coefficients do not affect the results while the Hurst parameter is near 0.5. However, as the Hurst parameter increases, the smaller variance coefficient show a better result.

In this model the results using Whittle's MLE fit the target Hurst values very well no matter what the variance coefficient is. However, another problem is raised when the traffic traces are applied to examine queueing behavior and network performance. In Fig. 6.17 three different traffic traces are depicted. All the traffic streams are generated with the same FGN sample trace whose Hurst parameter is 0.9 but with different variance coefficients. The traffic traces in the first diagram are generated with $\alpha = 1$. The second and third diagrams are generated with $\alpha = 500$ and 1000, respectively. From the diagrams it is apparent that the traffic streams are very different. The arrival rate of the three traffic traces are 1000.48, 1079.49, and 2282.29, respectively. The corresponding results using Whittle's MLE are 0.899, 0.887, and 0.879. Although the results slightly decrease as the variance coefficient increases, the difference between them are not significant. Such different traffic traces with almost the same results might cause the misuse of this model.

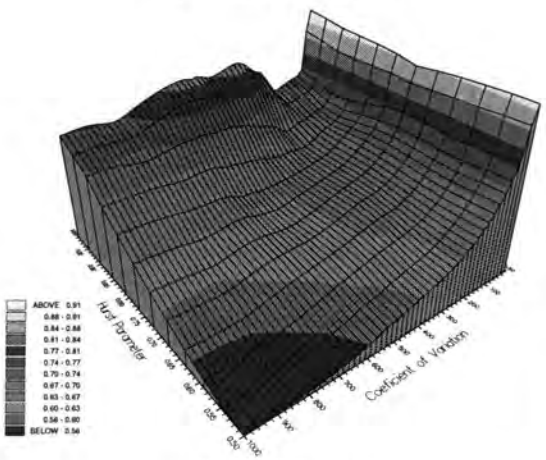
Two possibilities create this situation. First, Whittle's MLE might not be appropriate to estimate the Hurst parameter. Secondly, the Hurst parameter cannot fully represent the fluctuations of the traffic. The first cause is not very convincing because Whittle's MLE has a very rigorous mathematical derivation. Secondly, the Hurst parameter concerns the correlation of samples not the values. Although the traffic traces in Fig. 6.17 are quite different, the samples themselves have a similar degree of correlation. The term σ^2_{lim} in Eq. (6.24) might be the key factor which affects the fluctuation of the traffic traces. The study of this problem is a candidate for further research.



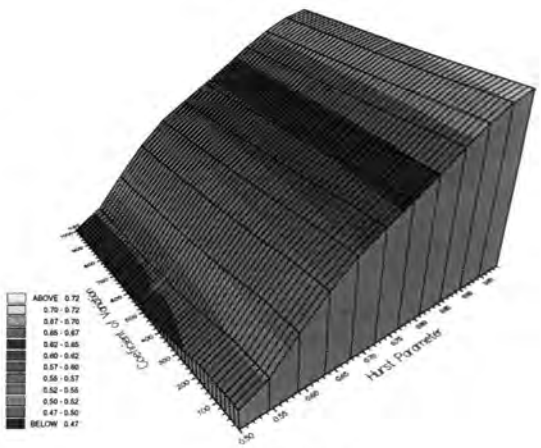
(a) Discard Event Count



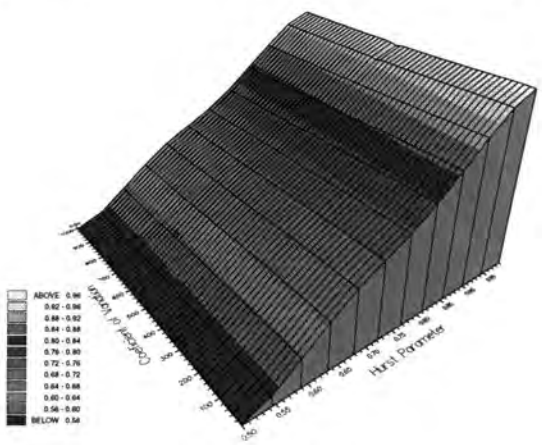
(b) Arrival Rate



(c) The R/S Statistic

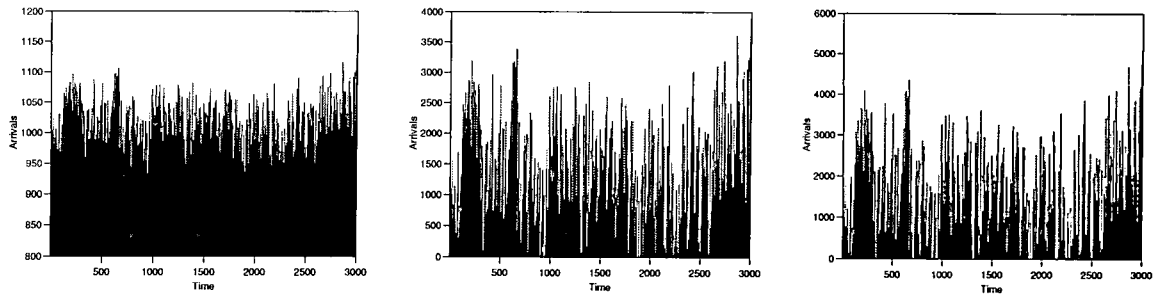


(d) The VT Plot



(e) Whittle's MLE

Figure 6.16: The arrival rate and estimated Hurst values of the storage model with different Hurst parameters and variance coefficients.



(a) $\alpha = 1$ (b) $\alpha = 500$ (c) $\alpha = 1000$

Figure 6.17: Traffic traces generated by the storage model with different variance coefficients.

6.5 Summary

In this chapter two models to transform the FBM processes to self-similar traffic have been investigated. The marginal distribution models use the invariant property to transform the non-integer, non-positive FBM samples to self-similar traffic with an arbitrary distribution. The models have been studied using an exponential, a gamma, and a Pareto distribution for the transformation processes. The distributions are obtained from the traffic measurements.

The results of the exponential and gamma distributions can be discussed together, because the exponential distribution is a special case of gamma distribution. Using these two distributions, the small scale structural assumptions can be applied to restrict arrivals to either one or none in an observed time slot. Investigation of the mean arrival rate of the traffic traces generated by the models with the gamma distribution shows that the output rate increases as the Hurst parameter increases. When the shape parameter of the gamma distribution increases, the output rate decreases. Therefore, a better result in terms of the accurate output rate is obtained as a larger shape parameter is used.

For the marginal distribution models with the gamma distribution, the VT plot and the R/S statistic do not give a good representation of the degree of self-similarity. The results of the VT plot are significantly lower than the Hurst settings, while the results using the R/S statistic shows almost constant values as the Hurst parameter changes. The values using Whittle's MLE exhibits a more reasonable result. However, if the observed time scale is too small, the results using Whittle's MLE is significantly lower than the target Hurst parameters.

The investigation of the marginal distribution model with the Pareto distribution shows that the distribution is not suitable for generating self-similar traffic. None the results accurately represent the actual Hurst values. This distribution has been proved in [TWS97] to be more appropriate for the modeling the ON/OFF traffic sources than their aggregated self-similar traffic.

The storage model has a more plausible physical interpretation and mathematical derivation. The model can be derived from the aggregation of many ON/OFF traffic sources with heavy-tailed distributions. It can also be seen as a fluid burst model to interpret real traffic. A concise version of the storage model was chosen for investigation because it has few parameters. In this model the traffic consists of two main parts: the mean level and the fluctuations. The former provides the mean arrival rate of the traffic while the latter controls the 'quality' of the traffic. The fluctuations are obtained from the FBM processes scaled by a weighting function of which the variance coefficient is the main parameter. The

effect of the parameter was investigated.

For this transformation the output rate of the generated self-similar traffic traces increases as the Hurst parameter and the variance coefficient increases. The result using the R/S statistic cannot represent actual Hurst parameters because this estimator is highly affected by the mean level of the traffic traces. The results using the VT plot are not affected by changing the variance coefficient, and are significantly lower than the actual Hurst settings. The results using Whittle's MLE accurately represent the desired Hurst parameters. In addition, the results are not significantly affected by changing the variance coefficient. The mean rates of the traffic traces are the only factor to consider of the value of variance coefficient because of the truncations of negative arrivals.

This exhaustive analysis has shown that the storage models is the preferred transformation producing accurate results. The marginal distribution models with the gamma distribution is a good transformation that could be used with care. It has been shown that the marginal distribution models with the Pareto distribution is not suitable for generating self-similar traffic because of its highly distorted Hurst parameter.

Chapter 7

Conclusions and Suggestions for Future Work

Asynchronous Transfer Mode (ATM), high-speed, cell-relay networks are likely to be used as the backbone of B-ISDN for the interconnection of enterprise networks composed of several LANs. Traffic modeling plays an important role in research because it is fundamental to the analysis and simulation of ATM networks. A great number of traffic models have been proposed to represent the traffic characteristics of different service categories. Most of them either cannot fully represent real traffic or must use complicated approaches. Thus, self-similar traffic models have been proposed as a unified model to capture the characteristics of various types of traffic.

To date, most of the research in this area has focused on the effect on ATM network mechanisms of the traffic models. Very few studies concentrate on the generation of high quality self-similar traffic. Although some research has covered the high-speed generating algorithms, this work has not been completed. This thesis has not only looked into the theoretical derivation of self-similar traffic models but also investigated their suitability in practice. In addition, some novel algorithms to generate self-similar traffic with high efficiency and accuracy for ATM networks have been proposed. This work has filled a gap in the study of self-similar traffic models. During the investigation, several conclusions have been drawn and some possible areas for future study identified. These are detailed below.

7.1 Conclusions

7.1.1 Survey of Traffic Modeling

There are two major categories of traffic models: those with short-range dependence and those with long-range dependence (LRD). Most conventional traffic models are from the former category while the recently proposed self-similar traffic models are the latter. Many real traffic measurements exhibit either strong correlation within the samples or infinite variance of their interarrival processes. While self-similar traffic models represent the aggregated traffic with LRD (the Joseph effect), the packet-train (ON/OFF) traffic models represent interarrival time series with heavy-tailed behavior (the Noah effect, infinite variance syndrome).

Self-similar traffic models have been tested against the six main criteria for traffic modeling. They have been found to satisfy five of the requirements: accuracy or closeness to reality or physical meaning of real traffic; generality of models for various types of traffic; usability for simulation or statistical stability; method of model parameterization; and small number of parameters. Only one of the criteria, analytical tractability, is not fulfilled at this stage of study. In fact this criterion, mainly concerned in the development of conventional traffic models, is not overly useful for self-similar traffic.

In conclusion, self-similar traffic models are more appropriate for the representation of the characteristics of bursty traffic because of the LRD and infinite variance of traffic. These models can satisfy most of the criteria of traffic modeling, except that of analytical tractability.

7.1.2 Foundations of Self-Similar Traffic Models

The stochastic and deterministic models are the two approaches to model the self-similar phenomena in time series. The former are based on the self-similar stochastic processes while the latter use chaotic maps in dynamic systems. Although the chaotic maps directly generate self-similar traffic, the generated samples are pseudo-random rather than stochastic. Therefore, the chaotic maps are very efficiency because of their direct generation of self-similar traffic, but are hard to use as models for network analysis. They are not appropriate for the modeling of stochastic processes but are understood to be efficient models for network simulation. There are two methods which can be used to represent the self-similar stochastic processes: the exactly self-similar FGN processes and the asymptotically self-similar FARIMA processes. The latter has two more parameters which make the models more flexible. However, two additional parameters increase the complexity of the models. The FGN processes are not only truly random and use fewer parameters, but also have closer physical interpretations. For these reasons, this study concludes that the FGN processes can be used to represent the self-similar phenomena more accurately and efficiently.

Many estimators have been proposed to obtain the Hurst parameter. Most of them are derived from the properties of the self-similar stochastic processes. After a thorough review of these estimators, three were selected as a benchmark to evaluate the generated samples because they were considered to be the best representations of the full range of estimation techniques. The simplest one is the variance-time plot which directly approximates from the property: the variance of the arithmetic mean decreases more slowly than the reciprocal of the sample size. Many other estimators are based on the same property but with modifications. The rescaled adjusted range statistic is based on the important scaling property of fractal samples. Whittle's approximate maximum likelihood estimator obtains the Hurst value by optimizing the logarithmic function of the periodogram. This estimator has a more rigorous mathematical derivation in comparison to other estimators.

In conclusion, the study of the self-similar stochastic processes indicates that the FGN processes are the most suitable models to represent self-similar traffic. In addition, three commonly-used estimators and a visual assessment tool can serve as the benchmark of the accuracy of the generated fractal samples.

7.1.3 Algorithms for Generating FBM and FGN samples

A preliminary selection process identified six algorithms which could be used to to generate the FBM and FGN sample traces. The dFGN and fFGN algorithms are directly or approximately derived from the definition of the FBM processes. These two algorithms generate the FGN samples while the RMD and SRA algorithms use the scaling property to generate interpolated FBM samples. The WM function and Whittle's approximate approach generate FBM and FGN samples in the frequency domain rather than the time domain.

Of these algorithms the RMD algorithm is the fastest sample generator while the SRA algorithm is only slightly slower. The main computational effort of the WM function is the inverse FFT, so its computational time is slightly worse than the SRA algorithm. The dFGN algorithm is the worst in terms of speed, but is the only one which can generate the samples on-the-fly.

All the algorithms generate samples aggregated with weighted Gaussian random variables, so their results follow the normal distribution. The Q-Q plot of their samples should fit the normal distribution. The dFGN algorithm and the WM function proved to have the best two results, with their sample traces fitting the normal distribution accurately. The SRA algorithm is slightly better than the fFGN algorithm but significantly better than the RMD algorithm. The Q-Q plot of the Whittle's approximate approach is the worst fit.

The results using the R/S statistic and the VT plot show that the 95% confidence intervals of the mean results of twenty different estimated values are very wide. The dFGN algorithm has the best result and Whittle's approach is the worst using the test of the R/S statistic. In contrast, the result of the dFGN algorithm with the VT plot is one of the worst cases. The confidence intervals of the results using Whittle's MLE are significantly small, therefore can be ignored. With this estimator, the dFGN algorithm is the best one while the WM function is only slightly poorer. The fFGN, RMD, and SRA algorithms are significantly worse than the best two but slightly better than Whittle's approach.

In conclusion, the investigation of the algorithms for generating FBM or FGN samples shows that the dFGN algorithm and the WM function are the best two in terms of quality, while the RMD algorithm, the SRA algorithm, and the WM function are superior in terms of speed of computation. Of these, the dFGN algorithm is the only one that can generate samples 'on-the-fly'.

7.1.4 Three Novel Algorithms for Generating FBM Samples

In order to attempt to overcome the problem of computational time associated with the dFGN algorithm, three novel hybrid algorithms were suggested and investigated.

The combination of the dFGN and RMD algorithms reduces the original computational time of the dFGN algorithm up to 50%. A scheme with 5 levels of depth is the best combination in that it significantly reduces the computational time and retains accuracy within a reasonable range. The combination of the dFGN and SRA algorithms was never better than the dFGN+RMD algorithm because of its approximation errors.

The dFGN+RMD and dFGN+SRA algorithms generate interpolated samples between two endpoints. The combination of the dFGN algorithm and the WM function uses a different approach by viewing the

sample traces as two parts: the main level and the fluctuations. The former is decided by the dFGN algorithm while the latter is determined by the WM function. It was found that the best combination scheme of the hybrid algorithm is to generate 7 WM samples between 2 dFGN endpoints.

Comparison of the dFGN+RMD and dFGN+WM algorithms shows that the former is a better hybrid approach. The accuracy of the best combination scheme of the dFGN+RMD algorithm is slightly worse than that of the dFGN+WM algorithm. However, the computational time of the dFGN+RMD algorithm with the best combination scheme is significantly faster than that of the dFGN+WM algorithm. The best overall scheme is the dFGN+RMD algorithm. This hybrid algorithm is better than the other hybrid approaches that were developed.

7.1.5 Models for Producing Self-Similar Traffic

The marginal distribution model and the storage model are two models which are used to convert the FBM sample traces into self-similar traffic. The marginal distribution models make use of the invariant property to convert the FGN samples from the normal distribution to a desired distribution. Three distributions, the exponential, gamma and Pareto distributions have been used to examine the effect of their parameters and the appropriateness of the distributions themselves. From simulations, the numerical results show that the marginal gamma distribution models with higher shape parameters produce more accurate sample traces. The marginal distribution models with Pareto distributions are not suitable for generating self-similar traffic.

The storage models show that many aggregated ON/OFF traffic models with heavy-tailed distributions can generate self-similar traffic. These models have more physical meaning. The mean level and fluctuations are the two terms used to control the quantity and quality of self-similar traffic traces. The latter is controlled by the variance coefficient while the former is determined by the arrival rate of the traffic. In simulations the numerical results suggest that the smaller variance coefficients generate more accurate traffic traces and produce fewer events whose arrival counts are negative. However, a smaller variance coefficient might not fully reflect the fluctuations of real traffic. The choice of an appropriate variance coefficient is still an open research topic.

On balance, of the two models used to transform the FBM samples to self-similar traffic, the storage model with an appropriate variance coefficient is a better approach.

7.1.6 Suitability of Using Hurst Estimators

Further studies, initiated by the contradictory results obtained between the VT plot and the other two estimators, when changing simulation parameters of the dFGN algorithm, suggest that the Whittle's MLE is an accurate Hurst estimator while the other two estimators cannot accurately obtain the Hurst values. In particular, the VT plot misrepresents the Hurst values as the Hurst parameter approaches one.

Numerical results from the estimation of self-similar traffic suggest that the use of the R/S statistic cannot accurately reflect the Hurst value of the traffic traces. The R/S statistic obtains the Hurst values from the range of the influx over a period of time. The mean level of the storage model controls the quantity of traffic, thus the results using the R/S statistic are highly affected by the mean arrival rate.

Of the three Hurst estimators, it has been shown that the VT plot is not suitable for estimating the Hurst parameter, while the R/S statistic can be used as a reference, only Whittle's MLE can obtain accurate Hurst values from fractal samples and fractal traffic.

7.2 Future Work

Self-similar traffic models offer promise for providing parsimonious descriptions of the essential features of packet traffic, but analytical tractability of the ensuing traffic models for performance analysis remains largely an open problem. This field of research is still in its infancy. In addition, the use of self-similar traffic modeling has theoretical problems as well as the problems of practical implementation.

7.2.1 Mathematical Aspects

Estimation of the Hurst parameter varies with trace generation techniques. Of them, Whittle's MLE is the best estimation technique, but it requires a large buffer to store samples. For greater statistical robustness, aggregation of samples have been proposed. However, the confidence intervals of the results significantly increase as the aggregation level increases. Further work needs to be done to identify accurate, efficient, and reliable estimators for each technique.

Another fundamental problem of self-similar traffic is that of the stationary property of second-order self-similar stochastic processes. The increments of a self-similar stochastic process form a stationary stochastic process. When using self-similar stochastic processes to model traffic with LRD the sample traces must be long due to the slow-decaying property. However, the arrival counts of real traffic cannot be stationary. In spite of explaining the basic problem with local Gaussian random processes in [Man69, Ber94], the model of self-similarity still attracts massive debate about this problem.

7.2.2 Engineering Aspects

The main objective of traffic analysis and modeling is to try to gain a good understanding of the actual dynamics of network traffic and to make use of this knowledge when designing, managing, and controlling existing or future networks [Res97]. The analysis of real traffic can lead to different results depending on the samples gathered. This has previously been a serious problem in the traffic characteristics. A full understanding of mechanisms and effects of different protocols becomes possible because of the improvements of techniques. Strong evidence indicates that there exist such underlying mechanisms of real traffic. Then, the design, management, and control of dynamic networks can be developed as the hierarchical nature of networks are able to model by structural models rather than black box models.

In engineering practice, the significance of the discovery of the self-similar nature of network traffic depends on whether or not features such as LRD have measurable impacts on queueing performance and traffic engineering. Self-similar traffic can impact on a wide range of traffic management related areas, including network dimensioning, quality of service, overload control, call admission, and error monitoring. There is considerable scope for further research on highly relevant problems of practical interest.

Considerable recent research has focused on the modeling and analysis of queueing systems driven by fractal traffic flows. There are several aspects to this research: reproducing the statistical behavior of packet traffic parsimoniously; investigation of the statistical impacts of self-similarity by means of trace-driven simulations; analysis of self-similar traffic models; and fast generation and simulation with self-similar traffic [EPW95].

In this thesis, the work focuses on how quickly self-similar traffic traces can be generated while still accurately representing the Hurst parameter. An improvement for generating FBM/FGN sample traces has been suggested but further investigations require to be undertaken. The RMD and SRA algorithms suffer from the criticism of generating independent subtraces within a single sample trace. In addition, the WM function can accurately represent the Hurst parameter when the degree of self-similarity is high. Although the dFGN+WM algorithm has been proposed to combine on their respective advantages, the results shown were not as good as expected. A different method of combining the WM function with the dFGN algorithm is worth investigating. Other hybrid approaches are also worth investigating.

As mentioned in Chapter 1 the development and design of a control mechanism is one of the most important issues for ATM networks. However, while most applications of fractal models in science and engineering have been based on empirical findings, they have almost exclusively focused on the models' powerful descriptive capabilities; their engineering implications and analysis have been largely ignored, mainly because fractal models are generally viewed as very difficult to analyze. The success of self-similar traffic models in teletraffic theory will not only depend on how well they describe actual network traffic, but also will depend to a large degree on the ability to use the models in network analysis and control.

The measurement of real traffic is another important issue. In Chapter 2, several measurements from ISDN, Ethernet, LAN, and VBR traffic have been summarized. Recently, the measurements from BT's Switched Multimegabit Data Service (SMDS) and Frame Relay networks have been found to be fractal nature as well [MA97]. However, the impact of many different existing or newly launched protocols and networks require further investigation. In addition, the traffic of individual source-destination pairs needs to be specified. The appropriate distributions and their parameters which fit the measured data are important to support the practical usage of traffic models.

Recently several works have dealt with the analysis of an ATM queueing system and the general problem of the relevance of self-similar traffic in practice [Ana95, AZN95, DLRT95, DO95, EGW94, EWP94, EW94, ENW96, MS97b, MS97a, LTG95, PM96, RE96, TG97, VKH96]. However, none of these studies make use of the derivative of the storage model from many ON/OFF traffic sources with heavy-tailed distribution, which provides a rigorous mathematical development and therefore might play an important role in analysis and modeling of queueing systems.

Appendix A

An Overview of Traffic Models

Based on [CPL+94] and [FM94] with other newly developed models this review brings together a summary description of traffic models. Such models are developed in two fundamental ways: either as part of any analytic model, or to drive a discrete-event simulation. All the models proposed are represent the characteristics of real-time and non-real-time traffic, such as voice, data, compressed audio and video traffic. Almost all of the models are based on conventional stochastic processes, the family of Poisson and Markov processes.

A.1 General Description of Traffic Modeling

The simplest traffic model is a traffic source consisting of the single arrival of discrete entries. The mathematical presentation is a *point process*, formed by sequences of arrival instants $T_1, T_2, ..., T_n...$ obtained from the initial trial $T_0 = 0$ on the time axis. Counting and interarrival time processes are two different equivalent descriptions of the point process [Cin75]. Let a *counting process* $\{N(t)\}_{t=0}^{\infty}$ represent, for each t the cumulative number of arrivals up to time t. $N(t)$ must satisfy the following conditions [RS95]:

1. $N(t) \geq 0$.
2. $N(t)$ is integer valued.
3. If $s < t$, then $N(s) \leq N(t)$.
4. For $s < t$, $N(t) - N(s)$ equals the number of events that have occurred in the interval (s,t).

Thus, it is a continuous-time, non-negative integer-valued stochastic process.

An *interarrival time process* is a non-negative random sequence $\{A_n\}_{n=1}^{\infty}$, where $A_n = T_n - T_{n-1}$ is the length of the time interval separating the nth arrival from the previous one. These three descriptions of stochastic process are based on the equality of events:

$$\{N(t) = n\} = \{T_n \leq t < T_{n+1}\} = \{\sum_{k=1}^{n} A_k \leq t < \sum_{k=1}^{n+1} A_k\} \tag{A.1}$$

since $T_n = \sum_{k=1}^{n} A_k$. Except for particular addressing, $\{A_n\}$ is assumed as a stationary sequence and its common variance is finite.

However, the arrivals may contain more than a single unit at a certain instant T_n. This kind of traffic for batch arrivals is called compound traffic. To describe *compound traffic* as an interarrival process, it can be represented by a non-negative random sequence $\{B_n\}_{n=1}^{\infty}$ where B_n is the number of units in the batch at arrival instant T_n.

Discrete-time traffic processes are applied when the time is slotted. From the mathematical aspect, the random variables A_n can assume only integer values, or the random variables N_t change their values at integer time instant T_n. In addition, the *workload* is a general concept describing the amount of work $\{W_n\}$ brought to a system by the n^{th} arriving unit; it usually assumes independent interarrival times and batch sizes.

A.2 Renewal Traffic Models

The mathematical simplicity makes the use of renewal models popular. Renewal models have an important property that in Markov chains and Markov processes if the initial state is i, then the time of successive entrances to that state i; and this fact in turn enables us to obtain many of the limiting results. This 'regeneration' property makes the mathematical expression simpler and may hold in more general situations. In a renewal traffic process, the random variables $\{A_n\}$ are i.i.d. taking result in $[0,\infty)$. Later, a new random variable sequence $\{Z_n\}$ was introduced, where $Z_0 = A_0 = 0, Z_1 = A_1, Z_2 = A_1 + A_2, \ldots$. The stochastic process $\{Z_n, n \in N + \{0\}\}$ is called a *renewal process* or *regenerative process*. Unfortunately, very few cases still keep this property when superimposed other renewal processes. Although the simplicity of mathematical analysis makes renewal models broadly used, there is a severe modeling drawback that the autocorrelation function of $\{A_n\}$ vanishes identically for the nonzero lags. It makes the renewal process hard to be applied to the burst phenomenon in broadband networks.

A.2.1 Poisson Processes

Poisson and Bernoulli processes are typical renewal processes applied for continuous and discrete time, respectively. An arrival process $\{N(t), t \geq 0\}$ is called a *Poisson process* if the following three axioms hold [Cin75]:

1. for almost t, each jump of $N(t)$ is of unit magnitude;
2. for any $t, s \geq 0$, $N(t+s) - N(t)$ is independent of $N(u)$ where $u \leq t$;
3. for any $t, s \geq 0$, the distribution of $N(t+s) - N(t)$ is independent of t.

The counting process satisfies the probability function

$$P\{N(t) = n\} = \frac{(\lambda t)^n}{n!} \exp(-\lambda t), \tag{A.2}$$

and the number of arrivals in disjointed intervals is statistically independent. Equivalently, this renewal process whose interarrival times $\{A_n\}$ are exponentially distributed with rate parameter λ:

$$P\{A_n \leq t\} = 1 - exp(-\lambda t). \tag{A.3}$$

The aggregation of several independent Poisson processes simply sum the rate parameters of each process with the same distribution. In addition, the independent incremental properties renders the Poisson process a memoryless process. Thus, it greatly simplifies queueing problems involving Poisson arrivals.

A.2.2 Bernoulli Processes

Instead of occurring on a continuous time scale, the *Bernoulli process* is a stochastic process $\{N_n\}_{n=1}^{\infty}$ with success probability p provided that

1. $N_1, T_2, \ldots$ are independent, and
2. $P\{N_n = 1\} = p, P\{N_n = 0\} = 1 - p$ for all n.

Applying the definition to cell streams means that there is a cell arrival with probability p, and no cell arrival with probability q at each time instance. From the definition of the Bernoulli process the corresponding number of arrivals is binomial,

$$P\{N_k = n\} = \binom{k}{n} p^n (1-p)^{k-n}, \tag{A.4}$$

where $0 \leq n \leq k$. The interarrival distribution is geometric with parameter p:

$$P\{A_n = j\} = p(1-p)^j, \tag{A.5}$$

where j is a non-negative integer. Therefore, this process is also known as a *geometric process*. With the same property of the Poisson process, the Bernoulli process is also memoryless.

The extension of the Bernoulli process with batch arrivals is also a renewal process. In this case, the number of arrivals at a time instant may be greater than 1. Let P_k denote the probability that k cells arrive at a given time instant. Then, the number of arrivals is independent of the past, preserving the memoryless property of Bernoulli process.

A.3 General Modulated Deterministic Traffic Models

The General Modulated Deterministic Process (GMDP) is derived from finite state machine (FSM) having N states [CPL+94, Sai94]. In each state, cells are generated with constant interarrival time d_i, where the index i indicates the state. Let X_i be the number of the cells which are emitted in state i. X_i may have a general discrete distribution $f_i(k) = P[X_i = k]$. The GMDP could include silent states where no calls are generated, and the duration of these states may also have a general discrete distribution. The changing states of the state machine are governed by a $N \times N$ transition matrix $P = [p_{ij}]$, where p_{ij} is the probability that at the end of its sojourn time in state i, the source moves to state j, $i \neq j$. Fig. A.1 is a GMDP example with three-state FSM.

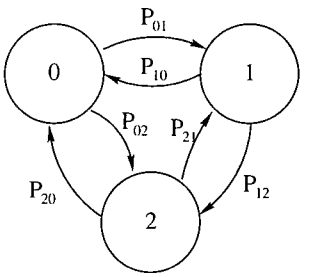


Figure A.1: A three-state FSM example of the general modulated deterministic traffic model.

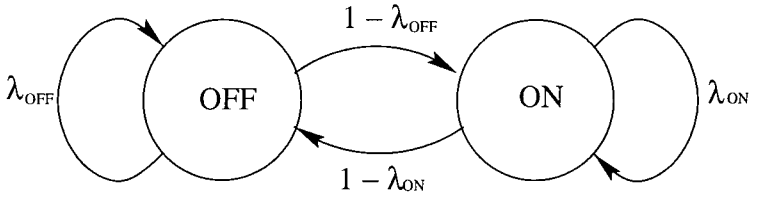


Figure A.2: The state diagram of an ON/OFF traffic model.

A.4 ON/OFF Traffic Models

The ON/OFF traffic model, which is also known as talk-spurt/silent model, burst/silent model, or sporadic models, was proposed by [Gru82, HL86, SW86]. It is a two states Markov model which alternates between phases of ON and OFF. Within an ON state, cells are emitted with a constant interarrival time. Originally, the model was introduced in the continuous-time domain; that is, the burst and silent duration as well as the cell distance within the ON state are continuous-time variables. In this case the sojourn times in both states have a negative exponential distribution with mean $1/\lambda_{ON}$ for ON state and $1/\lambda_{OFF}$ for OFF state. Fig. A.2 is the state transition diagram of this two phases traffic model. Other interarrival time distributions and characteristics of this model can be found in different research work, such as [HL86] and [SW86].

Similarly, this model is also used in the discrete-time domain, where the minimum cell distance T and the silent duration are integer multiples of a basic time unit, which is usually the cell transmission time. The sojourn time in the OFF state and the number of cells in an ON state have a geometric distribution. Therefore, the model is a special case of a two-state Markov modulate deterministic process with one silent state.

A.5 Markov-Modulated Traffic Models

Markov-modulated models introduce the very important idea that an auxiliary Markov process is evolving in time and its current state controls the probability law of the component state space [MH89, HL86, FM94, LL92, Onv95, Sai94]. Let $M = \{M(t), t \geq 0\}$ become a continuous-time Markov process, with state space $E = \{1, 2, ..., m\}$. Now assume that M is in state k, the probability law of traffic arrivals is completely determined by k, and this holds for every $1 \leq k \leq m$. Thus, the probability law for arrivals is modulated by the state of M. This modulating process can be far more sophisticated than continuous-time

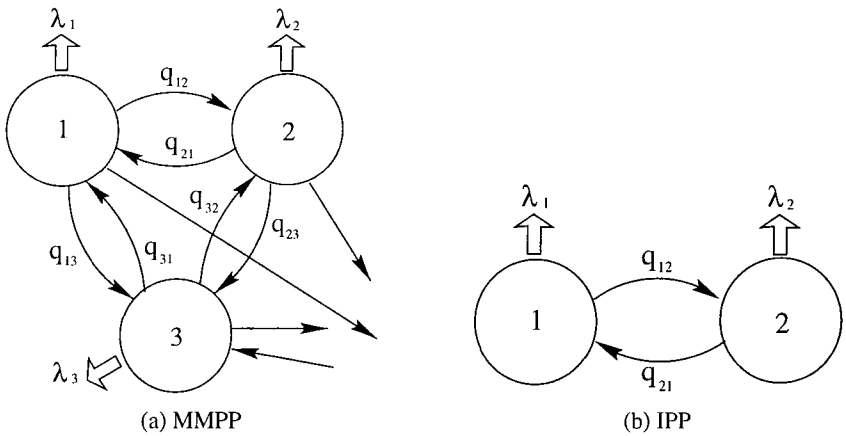


Figure A.3: State transition diagram of the Markov modulated Poisson process and the interrupted Poisson process.

or discrete-time Markov processes, but such models are far less analytically tractable.

A.5.1 Markov-Modulate Poisson Process

The Markov-modulated Poisson process (MMPP) is the most commonly-used Markov-modulated model, and has been extensively used to model various ATM traffic sources, such as voice and video, as well as characterizing the superposed traffic [HL86, SKY91, CPL+94, Sai94]. MMPP combines the simplicity of the modulating Markov process with that of the modulated Poisson process. It has the property of capturing both the time-varying arrival rates and correlations between the interarrival times. An MMPP is a doubly stochastic Poisson process. The arrivals occur in a Poisson manner with a rate that varies according to a k-state Markov chain, which is independent of the arrival process. Fig. A.3(a) depicts a general MMPP state transition diagram. In this diagram, λ_i denotes the arrival rate of state i and q_{ij} is the transition probability from state i to state j. Therefore, MMPP can be used in a number of ways, for instance, as a single traffic source with a variable rate. It can be quantized into a finite number of rates and each rate would give rise to a state in a Markov modulating process.

In addition, the superposition of MMPPs is also an MMPP. As the number of component processes increases, the number of states of the superposed process increases exponentially. To reduce the complexity of solving queues with large number of arrival streams, the superposed process may be approximated by a simpler process that captures important characteristics of the original process as closely as possible. The simplest model that has the potential to approximate an MMPP accurately with large number of phases is the two phase MMPP. Then, the problem is reduced to choosing the parameters of the two-state MMPP.

First, consider the simplest Markov modulated Poisson process, the Interrupted Poisson process (IPP) or 2-state MMPP. Fig. A.3(b) illustrates the state transition diagram of a general IPP. In cooperating with the ON/OFF traffic model the IPP is alternatively turned on for an exponentially distributed period of time and turned off for another independent exponentially distributed period of time. During the active (ON) period, the interval times of traffic are exponentially distributed, while no traffic are generated during the silent (OFF) period. All processes are assumed to be mutually independent. Let $1/\sigma_A$, $1/\sigma_S$,

and λ be the average duration of the active and silent periods, and the packet-generation rate during the active period, respectively. Then, the probability distribution function of the active period, silent period, and the interarrival times of packets during the active period are exponentially distributed. The two-state Markovian ON/OFF process in active and silent periods, π_A and π_S, can be obtained by

$$\pi_A = \sigma_S/(\sigma_A + \sigma_S) \quad \pi_S = \sigma_S/(\sigma_A + \sigma_S) \tag{A.6}$$

While the IPP models concentrate on the continuous-time Markov process, the Interrupted Bernoulli Process (IBP) is its discretized version. In this case time is slotted, in each slot length the arrival occurring time is equal to the cell time in the medium. A slot is either in an active state or in a silent state. A slot in an active state contains a cell with probability α and no cell with probability $1-\alpha$, while no cells arrive in a silent state.

A.5.2 Transition-Modulated Process

A variation of MMPP on the state modulated idea, which is, instead of a state itself, a state transition, and is called *transition-modulated processes* [CPL+94]. Let $M = \{M(t), t \geq 0\}$ be a discrete-time Markov process on the positive integers. State transitions occur on slot boundaries, and are governed by an $m \times m$ Markov transition matrix $Q = [Q_{ij}]$. Let B_n denote the number of arrivals in slot n, and assume that the probabilities $P\{B_n = k | M_n = i, M_{n+1} = j\} = t_{ij}(k)$, are independent of any part state information. Notice that these probabilities are conditioned on transitions, (M_n, M_{n+1}), of M from state M_n to state M_{n+1} during slot n. Furthermore, the number of traffic arrivals during slot n is completely determined by the transition of the modulating chain. Markov-modulated traffic models can be obtained if $t_{ij}(n) = t_i(n)$ depends only on the state i of the modulating chain in slot n, but is independent of its state j in the next slot $n+1$.

A.6 Markov and Markov-Renewal Traffic Models

Compared with renewal traffic models, Markov and Markov-renewal traffic models can be used to reflect the characteristics of burstiness, because they depend on the interarrival sequence $\{A_n\}$ with nonzero autocorrelations. Markov and Markov-renewal processes are derived from a Markov chain defined as [Cin75, Ros84]: A stochastic process $\{T_n, n \in N + \{0\}\}$ provides that

$$P\{T_{n+1} = j | T_0, T_1, ..., T_n\} = p\{T_{n+1} = j | T_n\} \tag{A.7}$$

for all $j \in E$ and $n \in N$, where E is a countable state space. The process moves from one state to another with random sojourn time in between. If the sojourn times are all exponentially independent of the next state, the process becomes a Markov process. It can be mathematically represented as: a stochastic process $\{X(t), t \geq 0\}$ with state space E provided that for any $t, s \geq 0$ and $j \in E$,

$$P\{X(t+s) = j | X(u), u \leq t\} = P\{X(t+s) | X(t)\}. \tag{A.8}$$

For fixed $i, j \in E$ the function $P\{X(t+s) = j | X(t) = i\} = p_{ij}$ is called the *transition function*. p_{ij} is the probability that a system stays in state i and then jumps to state j with the parameter λ_i of an exponential

distribution which depends on the state i alone, and the family of p_{ij} forms a matrix called the *transition matrix*. The discrete-time version of Markov model can be defined for the process $\{A_n\}$ in terms of a Markov transition matrix $P = [p_{ij}]$ where i and j signify two successive arrivals separated by i and j idle time slots. Thus, p_{ij} is the probability of j idle slot occurring when given the previous one with i-slot separation.

Markov-renewal traffic models are the super class of the processes that have been examined. The stochastic process $(X,T) = \{(X_n,T_n), n \in N+\{0\}\}$ is said to be a *Markov-renewal process* with space space E provided that

$$P\{X_{n+1} = j, T_{n+1} - T_n = A_{n+1} \leq t | X_0, X_1, ..., X_n; T_0, T_1, T_n\} = P\{X_{n+1} = j, A_{n+1} \leq t | X_n\}, \quad \text{(A.9)}$$

for all $n \in N$, $j \in E$, and $t \geq 0$. It means a Markov chain $\{X_n\}$ with associated jump times $\{T_n\}$ changes its states from the state (X_n, T_n) to next state (X_{n+1}, T_{n+1}) depending on current state X_n, but not on previous states nor previous jump times.

A.6.1 Phase-Type Renewal Process

An important renewal process occurs when the interarrival times have the characteristic of phase type [Neu79, LMHN90, Res92, FM94, Sai94]. A continuous-time Markov process $C = \{C(t), t \geq 0\}$ has state space $(0, 1, 2, \cdots, m)$. This set of states is said to be *closed* when it cannot attain any outside state from any state in it. An *absorbing* state is a state forming a closed set by itself [Cin75]. Phase-type interarrival times can be modeled as the time with a finite value to be absorbed by state 0, and when all other states are transient. Consider a $(m+1)$-state Markov process $\{X(t), t \geq 0\}$ with m transient states and one absorbing state. Its generator is of this form

$$Q = \begin{vmatrix} T & \mathbf{T}^* \\ \mathbf{0} & 0 \end{vmatrix}, \quad \text{(A.10)}$$

where T is a $m \times m$ matrix, with $T_{ii} < 0, T_{ij} \geq 0$, for $i \neq j$ and such that T^{-1} exists. $\mathbf{0}$ is a row vector of zeros. Moreover, the states $1, \cdots, n$ are all assumed to be transient. Since the row sums of Q equal zero, the column vector $\mathbf{T}^*$ has non-negative entries and satisfies $T\mathbf{e} + \mathbf{T}^* = \mathbf{0}$, where $\mathbf{e} = (1, \cdots, 1)^t$. The initial distribution a of $\{X(t)\}$ is denoted by (α, α_{m+1}), where α is a $1 \times n$ sub-stochastic vector. The distribution of hitting time $\tau = \inf\{t > 0, X(t) = m+1\}$ is called a *continuous phase-type (PH) distribution.* A renewal process where the inter-renewal times have a PH-distribution is called a *PH-renewal process.* The superposition of a Poisson process and a PH-renewal process have been studied as queues with background input (the Poisson process) and burst input (the PH-renewal process with group arrivals). The family of PH-renewal processes are relatively tractable traffic models. However, it seems there are very few traffic sources with this property.

A.6.2 Markovian Arrival Processes

Similar to the PH-renewal process the Markovian arrival process (MAP) is a subset of the semi-Markov process and is well-suited for matrix-analytic and numerical investigations [LMHN90, CPL+94, Sai94].

The major difference between the PH-renewal process and MAP is that the new state of the PH-renewal process is chosen according to the probability vector α which is independent of the state from which absorption occurred.

A Markov process on the state space $E = \{1, \cdots, m+1\}$, where $\{1, \cdots, m\}$ are transient state and $\{m+1\}$ is absorbing. The arrival process is a Markov process until absorption. The epoch of absorption corresponds to an arrival in the arrival process. The Markov process is then instantaneously restarted in a transient state, where the selection of the new state is allowed to depend on the state from which absorption occurred. Therefore, the MAP is a semi-Markov arrival process whose transition probability matrix is of the form

$$P\{X_{n+1} = j, L_{n+1} = k, T_{n+1} \leq t | X_n = i\} = \int_0^t exp(D_0 u) du D_k, \tag{A.11}$$

for $k \geq 1, t \geq 0$, where L_n is the number of arrivals of each transition, and the metrics $D_k, k \geq 1$ are nonnegative and the matrix D_0 has nonnegative off-diagonal elements. MAP is defined as a Markov renewal process with a transition probability matrix of the stated particular form [Neu92]. Thus, a MAP could be a renewal or non-renewal process. Its formulation includes Poisson processes, phase-type renewal processes, and others as special cases. It also has the attractive property that the superposition of independent MAP traffic streams form a single MAP traffic stream.

The discrete-time batch Markovian arrival process (D-BMAP) has been proposed for representing a VBR source at the cell-level [Ran94]. In addition, D-BMAP has also been used in speech recognition referred to as a hidden Markov process. However, the process is governed by an unobservable process modeled by a Markov process. Any processing which relies on detailed statistical information about the arrival streams can be difficult. Another difficulty of this model is that the parameters of this model must estimate online instead of being obtained in advance.

A.6.3 Semi-Markov Processes

The semi-Markov process releases the restriction that the Markov process must take a transition from the current state to other state, possibly back to the same state. Therefore, the sojourn time of the Markov process in a state is geometrically distributed. The semi-Markov process permits an arbitrary distribution of time so the process may remain in a state [Kle75, GK89, CPL+94, Sai94].

A *semi-Markov process* (SMP) is defined as a step linear process $S = \{S(t), t \geq 0\}$. For a fixed realization of the Markov chain $S_n = i_n, n \geq 1$, of duration $t_1, t_2 - t_1, t_3 - t_2, \cdots$, where $S(t) = S_n$, if $t \in [t_{n-1}, t_n)$, the sojourn of $S(t)$ in the states $i_1, i_2, i_3, \cdots$ are independent. Moreover, each one of these variables depends only on the state in which the process is at present and on the next state. Its distribution functions

$$P\{t_n - t_{n-1} < x | S_n = i, S_{n+1} = j\} = F_{ij}(x), n \geq 1, \tag{A.12}$$

are defined to achieve generality where it is assumed that $t_0 = 0$. Let $[p_{ij}]$ be the transition matrix, then the transition probability

$$P_{ij}(x) = p_{ij} F_{ij}(x). \tag{A.13}$$

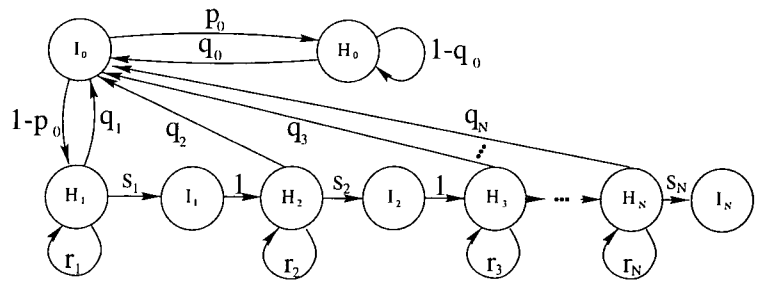


Figure A.4: State transition diagram of the semi-Markov process.

If at a given instant the process arrives at the state i, then with probability $P_{ij}(x)$, the next transition of the process will occur during a period of shorter than x and will proceed to the state j. The function $P_j(x) = \sum_j P_{ij}(x)$ is the distribution function of the time until the next transition of the process.

The semi-Markov model in Fig. A.4 which is a two phases process was proposed by Berry [Ber90]. The normal and active phase generates cells with interarrival distributions G_0 and G_1, respectively. The idle states $I_0, I_1, \cdots, I_N$ generate no cells. p_0 is the probability to transit from idle state I_0 to normal state H_0. The cell arrival time within a burst is drawn from distribution G_0 so that the interarrival times are generated with a probability $1 - q_0$. Similarly, there is a transition from idle state I_0 to active state H_1 with probability $1 - p_0$ and the interarrival time within a burst is drawn from the distribution G_1. $r_i, i = 1, \cdots, N$ control the mean returns to state H_i. For active state H_1 the state either re-enters to state I_0 with probability q_1 or passes through idle state I_1 to state H_2.

A.7 Fluid-Flow Traffic Models

The fluid traffic model views traffic as a stream of fluid, characterized by a flow rate [CPL+94, FM94]. These models are appropriate to cases where individual units are numerous relative to a chosen time scale. In the B-ISDN context of ATM, all packets are fixed-size cells of relatively short length; in addition, the high transmission speeds render the transmission impact of an individual cell negligible.

Although an important advantage of fluid models is their conceptual simplicity, important benefits also accrue from a simulation model of fluid traffic. For example, if one simulation distinguishes between cells, then every cell needs an event to identify them and there will consume a vast amount of CPU time and memory. In contrast, a fluid simulation would assume that the incoming fluid flow remains constant over a much longer time period. Traffic fluctuations are modeled by events signaling a change of flow rate.

Apparently, a fluid-flow model for non-bursty sources is useless, since in this case it is only the cell-level behavior that conveys information about the source activity and the information would be removed by the fluid-flow simplification.

A.7.1 Fluid-Flow Approximation of ON/OFF Traffic Models

Typical fluid-flow models assume that sources are bursty of the ON/OFF type [GSV95]. While in the OFF state traffic is switched off, whereas in the ON state traffic arrives deterministically at a constant

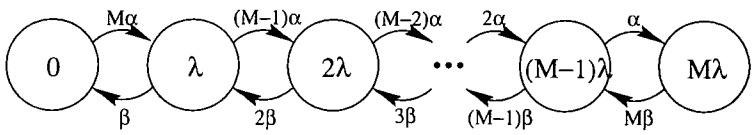


Figure A.5: State transition diagram of the multi-minisource process.

rate λ. For analytical tractability the duration of ON and OFF period are assumed to be exponentially distributed, as the simple assumption of the ON/OFF traffic model in Sec. A.4.

A.7.2 Multi-Minisource Traffic Models

The multi-minisource model represents the superposition of M identically independent ON/OFF traffic in a single fluid-flow model [CPL+94]. The pictorial representation of this model is illustrated in Fig. A.5. The model is based on a $(M+1)$-state continuous-time Markov chain describing the number of sources which are currently active. The arrival rate in state i is given by $i\lambda$. The cell interarrival time within a slot is deterministic and equals$1/(i\lambda)$.

A.7.3 Aggregate Multi-Minisource Traffic Models

The multi-minisource traffic model assumes that all traffic streams are identically independent traffic [CPL+94]. The superposition of two classes of ON/OFF traffic models can be modeled as a two-dimensional Markov chain depicted in Fig. A.6. Each class consists of $N_i, i = 1,2$, independent and identical ON/OFF traffic with parameters α_i, β_i, and λ_i.

A.8 Diffusion Traffic Models

The prototype for diffusion processes is the Brownian motion [Dyn65, IM65, Bre68, Fel71, Fre83, Pap91, Res92]. A normalized Brownian motion $B(t)$ is a stochastic process on a probability space with properties:

1. $B(0) = 0$,
2. $B(\cdot)$ is continuous, and
3. for $0 < t_1 < t_2 < \cdots < t_{n-1} < t_n$, the increments

$$B(t_1), B(t_2) - B(t_1), \cdots, B(t_n) - B(t_{n-1}), \tag{A.14}$$

are independent and normally distributed, with means 0 and variances $t_1, t_2 - t_1, \cdots, t_n - t_{n-1}$.

Imagine a continuous path process $\{X(t), t \geq 0\}$ which, given that it is a state x of its state space, adds an increment which is normal distribution with mean $\mu(x)$ and variance $\sigma^2(x)$. A method to think about such a process is through stochastic differentials:

$$dX(t) = \mu(X(t))dt + \sigma(X(t))dB(t), \tag{A.15}$$

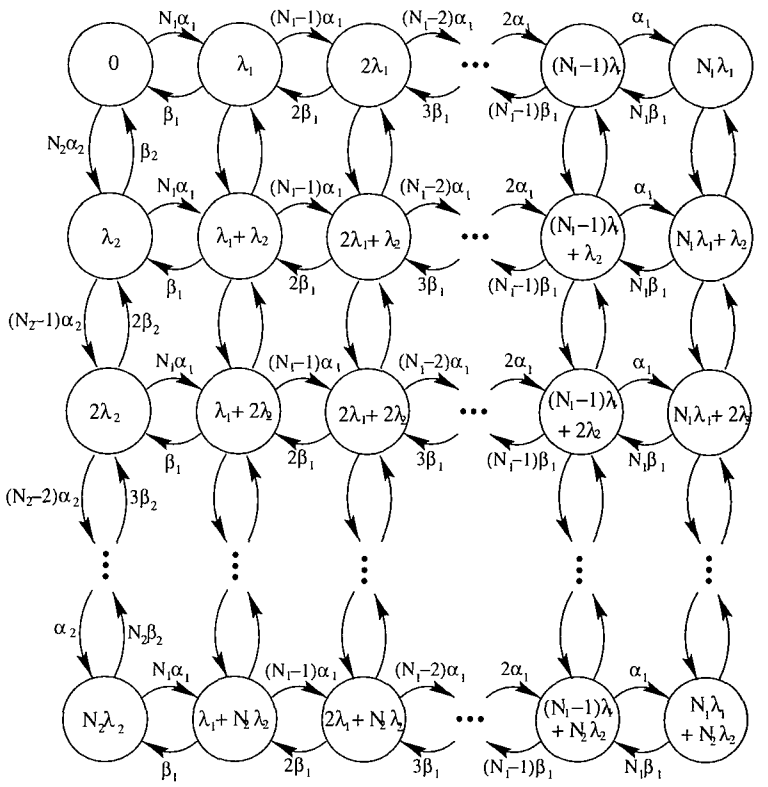


Figure A.6: State transition diagram of the aggregate multi-minisource process.

so that the change in the process at time t results from a drift of $\mu(X(t))$ and a Brownian increment with variance $\sigma^2(X(t))$. Then, $X(t)$ is a *one-dimensional diffusion process*. The extension of multi-dimensional diffusion process $\{\mathbf{X}(t)\}$, said M-dimensional, can be easily derived from above equation,

$$d\mathbf{X}(t) = \mu(\mathbf{X}) \cdot \mathbf{X}(t) + \sqrt{V(\mathbf{X})} \cdot d\mathbf{B}(t), \tag{A.16}$$

where $\mathbf{B}(t)$ is a M-dimensional Brownian motion with zero mean and the covariance matrix $I\delta(t)$, in which I is the $M \times M$ identity matrix and $\delta(t)$ is the impulse function. The corresponding mean and variance matrices, $\mu(\mathbf{X})$ and $V(\mathbf{X})$, are singular.

The discontinuous arrival and service processes in queueing models can be approximately derived as diffusion processes. The waiting time of jump processes has the Kolmogorov's backward equation, which can be observed in Brownian motion when the particles gain an elastic force. In addition, a more easily solved equation, Kolmogorov's forward equation, can be obtained formally by the Chapman-Kolmogorov equation (the continuous-time expression of the Markov property),

$$P(t+s) = P(t)P(s). \tag{A.17}$$

However, the forward equation exists only while the particles are not defective. This nature of the process is determined by additional boundary conditions.

Two boundary conditions have been proposed for modeling cell streams. The reflecting boundary is used to characterize the traffic from superposition of a homogeneous set of ON/OFF sources and the behavior of the superposition of the N Markov-modulated rate processes with M states [RK94]. Another

boundary condition, absorbing or jump, was proposed to obtain the lower and higher bounds of cell loss ration estimation [GMF96, GMO97]. In this application the ON/OFF sources are supposed as the input of the queues and the two bounds are obtained with finite and infinite length of queues. The diffusion models have long been used in queueing theory to model traffic and service. The computational effort are more attractive for more detailed traffic representations.

A.9 Autoregressive Traffic Models

Autoregressive models use the previous random variables within a time window stretching from the present into the past to obtain the next random variable through an explicit function [SMRA89, FM94, Sai94, Onv95]. The traffic of VBR-coded video is particularly suitable for the traffic models. Because the amount of data transmitted per frame varies due to interframe coding, the data rate varies in a continuous manner.

A.9.1 Linear Autoregressive Traffic Models

The simplest form of linear autoregression scheme is called AR(p), where p is the order of regression [BJ76, HTL92]. The autoregression function is

$$X_n = a_0 + \sum_{r=1}^{p} a_r X_{n-r} + \varepsilon_n, n > 0, \tag{A.18}$$

where $X_0, ..., X_{p-1}$ are prescribed random variables, the a_r are real constants, and the ε_n are zero mean, i.i.d. random variables, called *residuals*, are independent of the X_n. From Eq. (A.18) the recursive form obviously shows the next random variable in $\{X_n, n \in N + \{0\}\}$ is generated from the previous one via the equation. This simplicity makes AR schemes popular candidates for modeling autocorrelated traffic. For instance, a simple AR(2) model has been proposed to model VBR video traffic [HTL92]. More elaborate models can be constructed out of AR(p) schemes combined with other schemes. Although autoregressive models are typically used to fit the empirical autocorrelation function, they cannot generally fit the empirical marginal distribution. Therefore, they are mostly used in the simulation studies, but not are suitable for use in numerical or analytical studies of queueing models.

A.9.2 Autoregressive Moving Average Traffic Models

The Autoregressive Moving Average (ARMA) model is proposed to take into account recorrelation that the covariance in the bit streams of video sources is obtained to further recorrelate at lags. In addition to capturing two correlations, the autocovariance function dose not vanish for lags $k \neq 0$ and the utilization of queue will affect the correlation of the arrival process [GCMOO91, Sai94].

The duration of the interval is generally a fraction of a frame duration, in order to preserve frame and scene correlations. Line correlations are not preserved if the interval is longer than a line duration. The number of cells in the n^{th} interval is modeled by the recursive relation

$$X_i = g(\alpha Z_{i-m} + Y_i + v_i), \text{ with } |\alpha| > 1. \tag{A.19}$$

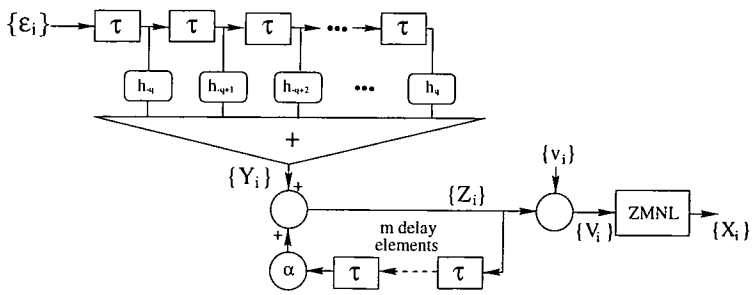


Figure A.7: Block diagram of the ARMA traffic models.

In this formula, Y_i is a sequence of correlated Gaussian random variables with zero mean and variance σ_Y^2, which models frame correlations. The scene and frame correlation are modeled by a sequence of correlated Gaussian variables, Z_i, which has zero mean and variance σ_Z^2. White noise is modeled by v_i, a sequence of uncorrelated Gaussian random variables with mean zero and variance σ_v^2. The function $g(\cdot)$ is a zero memory non-linear (ZMNL) operator which transforms the Gaussian distribution $f(V)$ to the required distribution of $f(X)$ seriously affecting the autocorrelation structure of the random variables, v_i. It is evident that the arrival rate in the n^{th} interval depends only on the arrival rate of the previous $(n-m)^{th}$ interval.

Characterizing the arrival statistics of ATM cells using the ARMA model is a three-step procedure: step measurement, parameter estimation, and transfer function. Fig. A.7 depicts the block diagram of the ARMA traffic model. Therefore, the ARMA arrival processes can be used in Monte Carlo simulations to estimate the probability-distribution function of the queueing delay and the mean and variance of the interdeparture time seen by an arrival rate. However, these models cannot be used in the numerical and analytical analysis of queues.

A.10 Transform-Expand-Sample Traffic Models

Other modeling approaches, called *transform-expand-sample* (TES) models, have been proposed to fit both marginals and autocorrelations of empirical records simultaneously [FM94]. The empirical TES models are based on the assumption that there exist some stationary time series. The model satisfies three fidelity requirements: the marginal distribution of the model should fit its empirical counterpart; the leading autocorrelations should approximate their empirical counterparts to a reasonable lag; and the same path realizations generated by simulation should be tractable by the empirical records. The first two are precise quantitative requirements, whereas the third is a heuristic qualitative one.

TES processes are divided into two sub-processes: TES^+ and TES^-, where the superscripts mean positive and negative lag-1 autocorrelations of TES processes, respectively. The two stochastic sub-processes are called background and foreground sequences. Background TES sequences are described as:

$$U_n^+ = \begin{cases} U_0, & n = 0, \\ < U_{n-1}^+ + V_n >, & n > 0, \end{cases} \qquad U_n^- = \begin{cases} U_n^+, & n \text{ even}, \\ 1 - U_n^+, & n \text{ odd}. \end{cases} \tag{A.20}$$

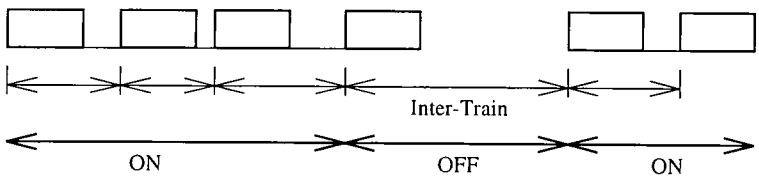


Figure A.8: Packet-train traffic model.

In these functions, U_0 is a uniform distribution in $[0,1)$; $\{V_n, n \in N+\{0\}\}$ is called the innovation sequence that is a sequence of i.i.d. random variables, independent of U_0, and the operator $< x >$ is module-1 function. Background sequences play an auxiliary role. The real target is the foreground sequences shown as:

$$X_n^+ = D(U_n^+), \qquad X_n^- = D(U_n^-), \tag{A.21}$$

where D is a transformation from $[0,1)$ to the reals, called a distortion. The background sequences are all Markovian stationary and uniform distribution between $[0,1)$, regardless of the probability law of the innovations $\{V_n\}$. However, the transition function $\{U^+\}$ is time invariant while $\{U^-\}$ is time dependent. The inversion method transforms any background uniform variants to foreground ones with an arbitrary marginal distribution.

The empirical TES modeling methodology takes advantage of this fact which effectively decouples the fitting requirements of the empirical distribution and the empirical autocorrelation function. Because the former is automatically guaranteed by TES, one can concentrate on fitting the latter.

A.11 Packet-Train Traffic Models

The packet-train model was proposed by Jain and Routhier [JR86] with the measurement of a token ring network at MIT. The model considers an individual connection between two end nodes, A and B. All packets on the individual connection are either from node A or node B. With a predefined time interval, the maximum allowed intercar gap (MAIG), any interpacket period is larger than MAIG that the previous packet is declared to be the end of the previous train. The time between the end of the previous train and the beginning of this train is defined as the intertrain time. Fig. A.8 illustrates the concept of this model.

This model is sometimes also referred to as the ON/OFF model with a more detailed breakdown. All the traffic on a network can be aggregated with each connection on the network. However, this model suffers a shortcoming because it is very hard to decide the MAIG value. In addition, it is closer to a concept than an analytical model. Therefore, it is more suitable for use in simulations.

A.12 Self-Similar Traffic Models

A number of researchers observing the practical network traffic under high-quality and high-resolution traffic measurements, have revealed that the packet traffic appears to be statistically self-similar [LW91, LWTW94]. The similar phenomenon is observed in an analysis of a few million of the observed frame

data generated by VBR video services [BSTW95]. A self-similar (or fractal) phenomenon exhibits structural similarities across all time scales.

Self-similarity manifests itself in a variety of different ways: a spectra density that diverges at the origin ($1/f^{\alpha}$ – noise ,$0 < \alpha < 1$); a non-summable autocorrelation function (indicating LRD); and a variance of the sample mean that decrease (as a function of the sample size n) more slowly than $1/n$.

The key parameter characterizing these phenomena is the so-called Hurst parameter, H, which is design to capture the degree of self-similarity in a given empirical record as follows [Hur51]. Let $\{Y_k\}_{k=1}^{n}$ be an empirical time series with sample mean $\overline{Y}(n)$ and sample variance $S^2(n)$. The rescaled adjusted range, or R/S statistic, is given by $R(n)/S(n)$ with:

$$R(N) = max\{\sum_{i=1}^{k}(Y_i - \overline{Y}(n)) : 1 \leq k \leq n\} - min\{\sum_{i=1}^{k}(Y_i - \overline{Y}(n)) : 1 \leq k \leq n\}. \quad \text{(A.22)}$$

It has been found empirically that many naturally occurring time series appear to obey the relation:

$$E[R(n)/S(n)] \cong n^H, \text{ n large}, \quad \text{(A.23)}$$

with an H typically about 0.73. On the other hand, for renewal and Markovian sequences, it can be shown that the previous equation holds with $H = 0.5$, for large n. This discrepancy, generally referred to as the Hurst phenomenon, is a measure of the degree of self-similarity in time series, and can be estimated from empirical data.

A.13 Traffic Burstiness

In broadband networks traffic burstiness is an important issue when the traffic of new services, such as compressed video, file transfer, etc., pour into the networks. *Burstiness* can be presented by the arrival points $\{T_n\}$ when they appear to form visual clusters; i.e., $\{A_n\}$ is shown as a relatively long interarrival time following several relatively short ones. The mathematical presentation of burstiness is very complicated. The shapes of the marginal distribution and autocorrelation function of $\{A_n\}$ are two major sources of the cause of burstiness. In particular, the strong positive autocorrelations are the most important effect of burstiness.

Several commonly-used definitions try to model burstiness [EW94,Sai94,FM94,Onv95]. The ITU-T definition is presented as 'the ratio of the peak-to-average traffic generation rate.' However, this definition is too crude to measure and has the backward of the dependence on the interval length utilized for rate measure. The coefficient of variance is a more precise measure of burstiness, which is defined as the ratio of standard deviation to mean, $c_A = Var[A_n]/E[A_n]$, of interarrival intervals. However, similar to the ITU-T definition it only take account of the first-order properties of traffic.

Another two traffic models for burstiness, the peakness measure and the index-of-dispersion measure, do consider the second-order properties, temporal dependence, in traffic. The index of dispersion for counts (IDC) is the function $I_c(\tau) = Var[N(\tau)]/E[N(\tau)]$ where τ is a given length of time interval. Since the number of arrivals is related to the sum of interarrival intervals, the numerator of the IDC includes the autocorrelations of $\{A_n\}$. Similar to the concept of IDC but more involved, the peakedness is defined

as the square coefficient of variance of the interarrival times of intervals. Assume that the traffic stream $\{A_n\}$ is offered to an infinite server group consisting of independent servers with common service time distribution F. Let S be the equilibrium number of busy servers. The peakedness is the functional $z_A[F] = Var[S]/(E[S])^2$, which maps a service time distribution to a real number.

In addition, according to the discovery of self-similar phenomenon the fractal correlation dimension has been proposed as a measure of distributions [EW94]. Fractal dimensions do not directly provide information about the size of a set, but about how non-uniform mass is distributed within the set. The correlation measure $C(t)$ of a point set is defined as the fraction of the numbers of pairs of points which are less than a distance t apart. If there are M arrivals, $C(t)$ can be expressed as

$$C(t) = \sum_{i}^{M} \sum_{j}^{M} \frac{1}{M^2} C_{ij} \tag{A.24}$$

where C_{ij} is 1 or 0 depending on whether the arrival points indexed i and j are within t of each other. Therefore, $C(t)$ can be stated as the probability of a pair of arrivals that occur within a time interval t. The correlation dimension D_c is then defined as

$$D_c = \lim_{t \to 0} \frac{\log C(t)}{\log t}. \tag{A.25}$$

Appendix B

Derivation of the dFGN+WM Algorithm

Let $B_H(t)$ be a FBM process. Two certain time epochs, 0 and T, of this process are known. Assume $B_H(0) = a$, $B_H(T) = b$. The variance of these two time epochs are known as $E[|B_H(T) - B_H(0)|^2] = \sigma^2 = (b-a)^2$. The process can be divided into two terms: the contant increasing part and the fluctuations. Thus, the FBM process can be expressed as

$$B_H(t) = \frac{t}{T}(b-a) + B_H(0) + B_{H,WM}(t), \text{ with } 0 \leq t < T. \tag{B.1}$$

Let $\tilde{B}_{H,WM}(f,T)$ be the spectrum of the FBM process,

$$\tilde{B}_{H,WM}(f,T) = \frac{1}{T}\int_0^T B_{H,WM}(t)\exp(-j2\pi ft)dt. \tag{B.2}$$

Therefore, the FBM process $B_H(t)$ can be expressed as

$$\begin{aligned} B_{H,WM}(t) &= \int_{-\infty}^{\infty} \tilde{B}_{H,WM}(f,T)\exp(j2\pi ft)df \\ &= \sum_{n=1}^{N} \tilde{B}_{H,WM}(n,T)\exp(j2\pi\frac{n}{N}t) \end{aligned} \tag{B.3}$$

The variance of the FBM process can be represented as

$$E[|B_H(t) - B_H(0)|^2] = E[|\frac{t}{T}(b-a) + B_{H,WM}(t)|^2] \tag{B.4}$$

$$\begin{aligned} &= E[(\frac{t}{T})^2(b-a)^2 + \frac{t}{T}(b-a)B_{H,WM}(t) \\ &\quad + \frac{t}{T}(b-a)B^{\star}_{H,WM}(t) + B_{H,WM}(t)B^{\star}_{H,WM}(t)] \end{aligned} \tag{B.5}$$

$$= (\frac{t}{T})^2(b-a)^2 + E[|B_{H,WM}(t)|^2] \tag{B.6}$$

For the discrete-time implement, $[0,T]$ is divided into N subintervals. Then, the discrete version of the FBM process can be expressed as

$$B_H(k) = \frac{k}{N}(b-a) + B_H(0) + B_{H,WM}(k), k = 0, \cdots, N. \tag{B.7}$$

Similarly, the variance of the discrete-time FBM process is

$$\begin{aligned} E[|B_H(k) - B_H(0)|^2] &= (\frac{k}{N})^{2H}\sigma^2 = (\frac{k}{N})^{2H}(b-a)^2 \\ &= (\frac{k}{N})^2(b-a)^2 + E[|B_{H,WM}(k)|^2]. \end{aligned} \tag{B.8}$$

We know

$$\tilde{B}_{H,WM}(n,T) = \frac{1}{N}\sum_{n=1}^{N} B_{H,WM}(k)\exp(-j2\pi n\frac{k}{N}) \tag{B.9}$$

Thus,

$$B_{H,WM}(k) = \sum_{n=1}^{N} \tilde{B}_{H,WM}(n,T)\exp(j2\pi n\frac{k}{N}) \tag{B.10}$$

$$E[|B_{H,WM}(k)|^2] = E[(\sum_{n=1}^{N} B_{H,WM}(n,T)\exp(j2\pi n\frac{k}{N}))(\sum_{l=1}^{N} B^{\star}_{H,WM}(l,T)\exp(-j2\pi l\frac{k}{N}))] \tag{B.11}$$

The above equation can be divided into two parts. If $n \neq l$, then

$$\begin{aligned} &E[B_{H,WM}(n,T)\exp(j2\pi n\tfrac{k}{N})B^{\star}_{H,WM}(l,T)\exp(-j2\pi l\tfrac{k}{N})] = \\ &E[B_{H,WM}(n,T)\exp(j2\pi n\tfrac{k}{N})]E[B^{\star}_{H,WM}(l,T)\exp(-j2\pi l\tfrac{k}{N})] = 0 \end{aligned} \tag{B.12}$$

If $n = l$, then

$$\begin{aligned} E[B_{H,WM}(n,T)\exp(j2\pi n\frac{n}{N})B^{\star}_{H,WM}\exp(-j2\pi n\frac{n}{N}) &= E[|B_{H,WM}(n,T)|^2] \\ &= c_k E[|B_{H,WM}(1,T)|^2]n^{-1-2H} \\ &= \sigma_f^2(k)n^{-1-2H} \end{aligned} \tag{B.13}$$

Thus,

$$\sigma_f^2(t) = [(\frac{t}{T})^{2H}\sigma^2 - (\frac{t}{T})^2(b-a)^2]/\int_0^T t^{-1-2H}dt. \tag{B.14}$$

Finally, the variance of the magnitude part of spectrum of the FBM process is

$$\begin{aligned} \sigma_f^2(k) &= [(\frac{k}{N})^{2H}\sigma^2 - (\frac{k}{N})^2(b-a)^2]/\sum_{n=1}^{N} n^{-1-2H} \\ &= \sigma^2[(\frac{k}{N})^{2H} - (\frac{k}{N})^2]/\sum_{n=1}^{N} n^{-1-2H} \end{aligned} \tag{B.15}$$

Appendix C

Measurements of Burstiness of Self-similar Traffic

As mentioned in Appendix A there is increasing evidence to show that the non-real-time traffic is strongly correlated and bursty. Although many definitions of burstiness try to describe the characteristics of bursty traffic, it is too complicated to be represented by simple mathematical formula. From the simulations in Chapter 6 the fractal traffic streams normally exhibit more fluctuating pictorial graphs with higher Hurst parameters. In this appendix three different definitions, the peak-to-mean ratio (PMR), the average burst length, and the squared coefficient of variation (CSQ) are used to examine the fractal traces generated by the marginal distribution model with exponential distribution. Meanwhile, from the results an appropriate definition can be identified to represent the burstiness of self-similar traffic.

C.1 Bursty Measurements

Various definitions try to characterize the burstiness behavior. Three widely used measures, the peak-to-mean ratio (PMR), the burst length, and the squared coefficient of variation (CSQ), will be discussed in this section [Onv95, EW94]. The ITU-T defines the term burstiness as corresponding to the ratio of peak-to-average traffic generating rate, i.e., the PMR.

C.1.1 Peak-to-Mean Ratio

The PMR is calculated by taking the ratio of peak rate to average rate over specified peak and average rate measurement intervals. The PMRs are widely used as measurement of burstiness. Higher PMR indicates increasing burstiness. The difficulty in using peak rates lies in the interpolation of a statistic which is not well behaved. The choice of measurement and reporting intervals will affect the peak levels. The peak level measurements can not be directly used by standard queueing models in which the average or percentile delays and throughput form the basis of service criteria and requirements. Furthermore, the measurements of peak levels are inefficient.

For packet trunks, the service time depending on the length of the packet, the PMR is computed by taking the peak byte count over the specified interval to the average byte count. To simplify the counts

of the packets instead of bytes can be used. This assumption does not lose its generality because for different applications the mean packet sizes are different. To obtain the PMR with byte counts is to multiply the mean packet size with the PMR with packet counts.

C.1.2 The average Burst Length

The definition of average burst length is related to the ON/OFF traffic model which is also known as the packet train model [JR86]. Jain and Routhier who proposed this model collected the packet data on a token ring network at MIT. The packet train model is based on the traffic which communicates between two distinguished sources. Depending on a pre-given time interval, if no packet is sent between this pair of sources, the previous train is declared to have ended and the next packet is declared to be the first packet of the next train. The model is depicted in Figure A.8.

Based on this model the intertrain time is the OFF period while the time between two OFF periods is the ON period. Although the ON/OFF model is based on a pair of sources, the same concept can be used to model the burst length for aggregated sources. Based on a pre-defined time gap if the interarrival time of two successive packets is large or equal to the pre-defined value, the interarrival (intertrain) time is the OFF period and the burst length is defined as the ON period, that is, the time interval between two OFF periods. However, the longer burst length does not indicate the higher burstiness. In contrast, the more fluctuations the traffic traces indicate the shorter the burstiness length.

The average burst length can be directly used as the input of standard queueing to predict average delays and throughput. In addition, the burst lengths depend on the choice of the reporting intervals.

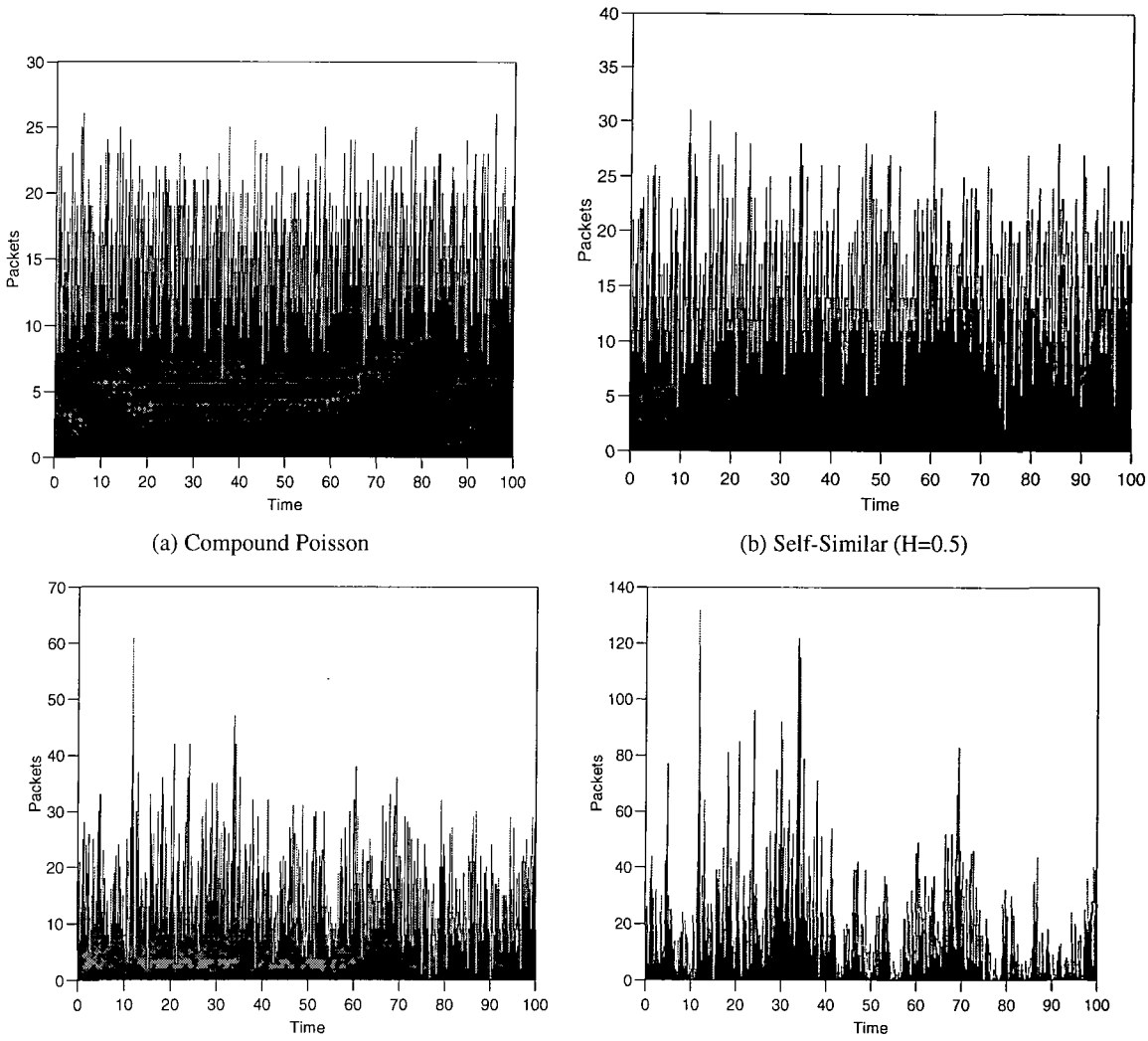
C.1.3 Squared Coefficient of Variation

The squared coefficient of variation (CSQ) of the interarrival time process is another broadly used measure of traffic burstiness. If X is the packet interarrival time random variable with mean $E[X]$ and variance σ_X^2, then the squared coefficient of variation is defined as the ratio of the variance and the square of the mean:

$$CSQ = \frac{\sigma_X^2}{(E[X])^2} \tag{C.1}$$

The CSQ can be directly used as the input of two moment queueing approximations to estimate the mean delays and queue lengths. Nevertheless, the CSQ can not calculate on-line network measurements because it is derived by the interarrival time. The larger CSQ corresponds to higher burstiness. For a standard Poisson process its mean and variance are both equal to the arrivals rate multiplying the observed time interval, so its CSQ is equal to 1.

However, there is a potential impact due to the basic property of self-similar processes. Self-similar processes have infinite variance, which means that the measure of the CSQ will increase unboundly as the observation interval increases, rather than converge to a constant value. Thus, CSQ can be an arbitrary value because it depends on its observation interval.



(a) Compound Poisson

(b) Self-Similar (H=0.5)

(c) Self-Similar (H=0.7)

(d) Self-Similar (H=0.9)

Figure C.1: Traffic traces of compound Poisson process and self-similar process.

C.2 Simulation Traffic Analysis

In the previous section, three metric benchmarks have been introduced and will now be used to examine the traffic traces generated by the compound Poisson process and the self-similar processes with different Hurst parameters. The traffic traces used as inputs for simulation in this study are generated by the method described in Chapter 6.

Figure C.1 depicts the generated traffic streams. Figure C.1.(a) is generated by a compound Poisson process in which there are 10 sources and the mean arrival rate of every source is 15 packets per unit of time. The other three graphics are generated by the above algorithm with the aggregated mean arrival rate of 150 packets per unit time but different Hurst parameters. All the traffic traces have 100 unit of times. For the self-similar traffic traces the time interval τ is chosen as 0.01 unit of time. From the diagrams the higher the Hurst value is, the more fluctuation the traffic stream exhibits.

The CSQ of the traffic trace is a burst measurement related to the first two statistical moments, mean and variance. Figure C.2 illustrates the results of the CSQ analysis where the inputs are the previously generated traffic traces. The compound Poisson process has the same result as the theoretical property

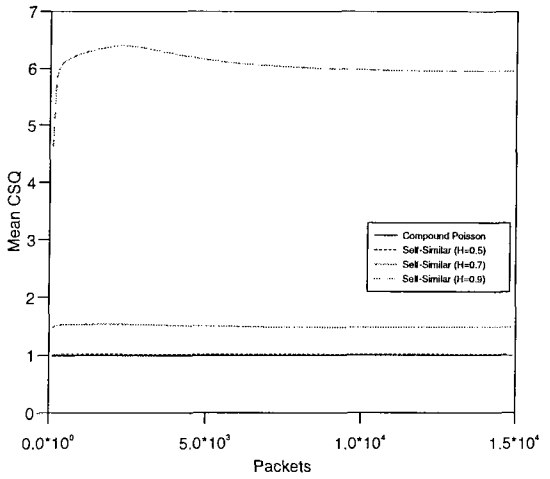


Figure C.2: Squared coefficient of variation.

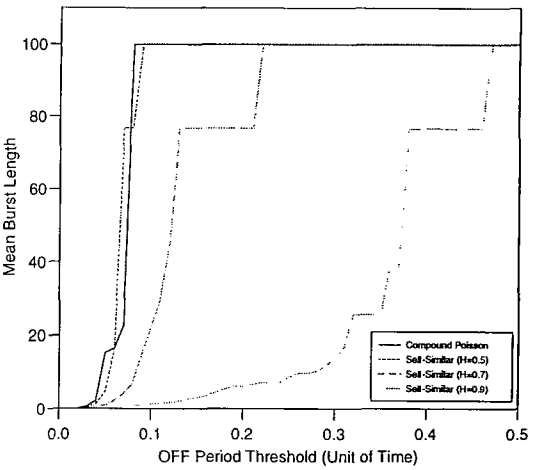


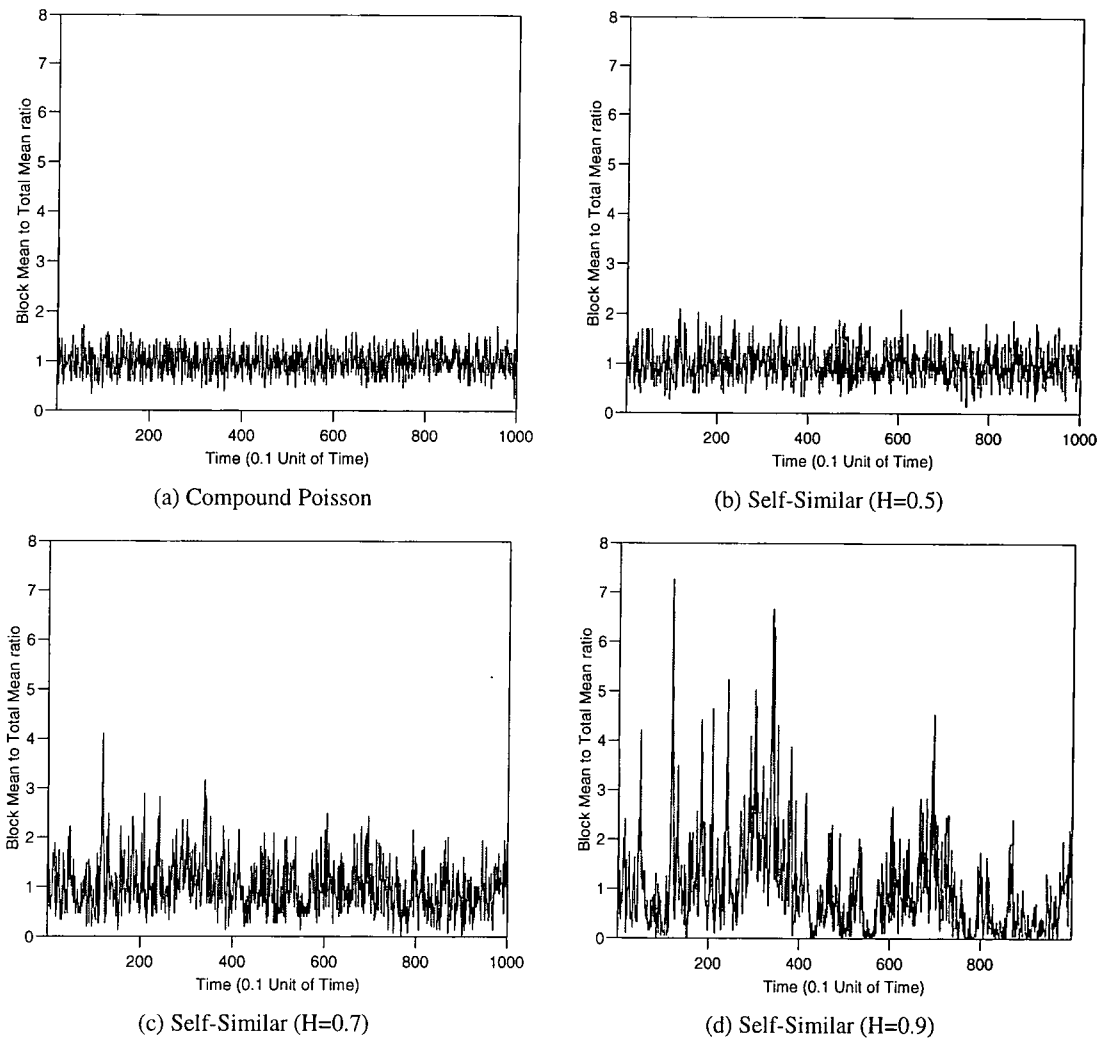
Figure C.3: The average burst length.

in which the CSQ value should be equal to 1. In addition, the self-similar traffic trace with $H = 0.5$ has almost the same result as the compound Poisson process. Although the observations of the diagrams of the compound Poisson process and the self-similar process with $H = 0.5$ depicted in Figure C.1 seem very different, their statistical characteristics are very similar. In particular, the self-similar process with the Hurst parameter equal to 0.5 is the only case where the self-similar process has finite variance. The self-similar process with $H = 0.5$ is the same as the ordinary Brownian motion and the random process of its increments is called ordinary Gaussian noise, whose statistical characteristics can be represented by two commonly used moments, mean and variance. In this case its increments are independent. This traffic trace is generated from the FGN with zero mean and unit variance and regulated by a negative exponential function. Thus, its CSQ should be the same as the regular Poisson process.

In theory the self-similar traffic traces which have higher Hurst values, $0.5 < H < 1.0$, then their variances are infinity. That means the CSQ values will not converge. In this figure the two self-similar traffic traces with higher Hurst parameter converge to a constant value when the observed interval increases. It is caused by the algorithm which maps FGN into the traffic trace. In this algorithm the traffic traces are generated from FGN by regulating with the negative exponential function. This procedure of mapping the self-similar process into a counting process will shape the infinite variance into a fixed constant. However, the higher Hurst value still indicates the larger CSQ.

The expected burst length versus the OFF period threshold is depicted in Figure C.3. The compound Poisson trace and the self-similar trace with $H = 0.5$ have almost the same results. The smoother the trace is, the shorter the mean interarrival time is. In this case the appropriate OFF period threshold can be chosen as 0.1. However, the threshold value varies between different arrival rates. Different applications have different arrival rates. Thus, how to chose an appropriate threshold is very difficult in practical terms. In addition, the average burst length only relates to the first order moment of statistical characteristic while the second order moment, variance or correlation, plays a more important role in real traffic.

Using the traffic traces as inputs the PMR is computed by taking the peak arrival rate over 0.1 to 10 unit of time. Figure C.4 illustrates four subfigures of the four different traffic traces with 0.1 observed



(a) Compound Poisson

(b) Self-Similar (H=0.5)

(c) Self-Similar (H=0.7)

(d) Self-Similar (H=0.9)

Figure C.4: Block mean to whole trace mean ratio with 0.1 unit time interval.

interval. Figure C.4.(a) is the result of the compound Poisson traffic trace. Figure C.4.(b) - (d) are the consequences of the self-similar traffic traces with the Hurst parameters, $0.5, 0.7$, and 0.9, respectively. The maximum value of each subfigure is their peak-to-mean ratio with 0.1 unite of time interval.

From these figures the block mean to whole trace mean ratios of the compound Poisson traffic trace and the self-similar traffic trace with $H = 0.5$ slightly fluctuate around 1, no matter what time interval is taken. When the Hurst parameter increases, the dynamic range of the ratio increases. However, the larger the time interval is, the smother the ratio is. The PMR of the four traffic traces with different time intervals is listed in Table C.2. In this case the appropriate time interval can be chosen as 0.1 unit time. Thus, a carefully chosen time interval for the PMR can give information on queueing behavior. If the rate measurement interval is too small, the PMR is too peaky and does not capture the low frequency burst. If the interval is too long, even the low frequency variations are smoothed out to a misleadingly low value.

Self-similar traffic models are proposed to model packet data. Long-range dependence can be characterized by a parameter that describes the magnitude of fluctuations and the Hurst parameter H which describes the decay in the autocorrelation. For conventional traffic models which exhibit short-range dependence the autocorrelation in the time series of counts are nonexistent or decay exponentially with

	0.1 Unit Time	1.0 Unit Time	10.0 Unit Time
Compound Poisson	1.72233	1.21888	1.04665
Self-Similar (H=0.5)	2.10005	1.36165	1.04325
Self-Similar (H=0.7)	4.64818	1.89969	1.21324
Self-Similar (H=0.9)	7.26819	3.1368	1.95367

Table C.1: The PMR of the traffic traces with different time intervals.

the lag. These processes have been shown to have $H = 0.5$. For self-similar processes with long-range dependence the autocorrelation is

$$\gamma(k) \sim ck^{-(2-2H)} \tag{C.2}$$

for asymptotic lag k and $0.5 < H < 1$. Using this measure, the larger the Hurst parameter indicates the burstier the traffic.

C.3 Summary

Although there are several approaches to capture the characteristics of burstiness, none of them can be fully accepted. Conventional traffic models have tried to capture it, but, they suffer the problem of inaccuracy or too many parameters. Self-similar traffic models can use fewer parameters to model the packet traffic. The Hurst parameter is the key indicator of long-range dependence processes with $0.5 < H < 1$. Conventional traffic models use $H = 0.5$.

The PMR, average burst length, and CSQ are the three commonly used benchmarks. The PMR makes use of the process of interarrivals. The burst length metric is devoted to the ON/OFF model. The CSQ is a measure with the mean and variance of the interarrival process. Analyzing the traffic traces generated by compound Poisson process and the self-similar processes with different Hurst parameters, the larger the Hurst parameter indicates the higher the degree of burstiness.

Although all three metric benchmarks can be used to indicate the degree of burstiness, in practical usage they still suffer problems. It is necessary to chose an appropriate time interval for the PMR. If the interval is too long, even the low frequency burst will be smoothed. If the interval is too small, the PMR is too peaky and the effect of the low frequency burst may be lost. The average burst length is affected by the choice of OFF period threshold. If the threshold chosen IS too small or too large, the average burst length cannot be distinguished. Although the CSQ can distinguish the degree of burstiness very well and quickly, it cannot operate on-line measurement.

Although the larger Hurst parameter indicates a burstier model, the relationship between them is still unclear. In addition, there are other benchmarks for measuring the burstiness which have not been examined, such as the index of dispersion for counts (IDC), peakedness, correlation dimension, etc. These will be the subject of further study.

Appendix D

Tables of Computational Time, Output Arrival Rates, and the Hurst Estimation Results

This appendix complies three computational time tables of the three novel hybrid algorithms in Chapter 5, three output arrival rates of the transforming models, and seven tables of the estimated results of the hybrid algorithms in Chapter 5 and the transforming models for self-similar traffic in Chapter 6.

D.1 Table of Computational Time

This section tabulates the computational times of the hybrid algorithms. For each table the upper and lower rows are the results of the two original algorithms. Between them, the results of different combination schemes are tabulated. All the results are obtained from at least twenty simulations of each algorithm with each Hurst parameter setting. The measured scale is in second. Table D.1 shows the results of the dFGN, RMD, and dFGN+RMD algorithms. Table D.2 shows the computational time of the dFGN, SRA and dFGN+SRA algorithms. The results of the dFGN algorithm, the WM function, and their hybrid approach are tabulated in Table D.3.

D.2 Tables of Output Arrival Rates

This section compiles the output arrival rates of the transforming models in Chapter 6. Table D.4 tabulates the mean arrival rates and their 95% confidence intervals of the marginal distribution models with the exponential distribution. Table D.5 records the output mean arrival rates of the marginal distribution models with the gamma distribution. Table D.6 lists the output arrival rates of the self-similar traffic streams generated by the storage models.

Algorithm	Hurst Parameter		
	0.5	0.7	0.9
dFGN	395.973 ± 17.80941	339.947 ± 8.918893	338.3165 ± 5.028307
Hybrid-1	191.4736 ± 4.802936	166.8595 ± 5.147197	166.9199 ± 3.998776
Hybrid-7	49.38212 ± 1.414637	42.7595 ± 1.356706	42.535 ± 1.06458
Hybrid-31	12.81488 ± 0.3708887	11.3045 ± 0.3390075	11.27275 ± 0.2258736
Hybrid-511	2.3075 ± 0.1192629	2.219875 ± 0.1257719	2.227625 ± 0.1451815
Hybrid-1023	1.277125 ± 0.1229052	1.341375 ± 0.1065235	1.33475 ± 0.1154287
RMD	0.7553333 ± 0.1556106	0.734625 ± 0.1112659	0.7478333 ± 0.1261012

Table D.1: Average computational time in seconds of different combination schemes of the dFGN+RMD algorithm.

Algorithm	Hurst Parameter		
	0.5	0.7	0.9
dFGN	395.973 ± 17.80941	339.947 ± 8.918893	338.3165 ± 5.028307
Hybrid-1	197.1191 ± 3.25762	171.6371 ± 2.623185	171.7443 ± 2.89168
Hybrid-7	51.1254 ± 1.396938	44.9493 ± 1.083579	44.8978 ± 1.464038
Hybrid-31	13.8776 ± 0.4788871	12.4733 ± 0.4638044	12.4511 ± 0.4027214
Hybrid-255	3.1691 ± 0.1843511	3.1014 ± 0.1428945	3.0898 ± 0.09597877
Hybrid-1023	2.063 ± 0.09874946	2.1704 ± 0.09868735	2.1668 ± 0.08481355
SRA	1.126417 ± 0.4843563	1.112214 ± 0.485692	1.1492 ± 0.4579411

Table D.2: Average computational time in seconds of different combination schemes of the dFGN+SRA algorithm.

Algorithm	Hurst Parameter		
	0.5	0.7	0.9
dFGN	395.973 ± 17.80941	339.947 ± 8.918893	338.3165 ± 5.028307
Hybrid-1	197.825 ±0.09071445	173.054 ± 0.06275867	173.0545 ± 0.0889078
Hybrid-7	51.615 ± 0.04946203	45.863 ± 0.07084026	45.886 ± 0.1026159
Hybrid-31	14.743 ± 0.04501043	13.7635 ± 0.0477037	13.755 ± 0.04946203
Hybrid-255	4.1025 ± 0.05502533	4.5045 ± 0.06921618	4.497 ± 0.05796886
Hybrid-1023	3.23 ± 0.05507125	3.7925 ± 0.002440789	3.8155 ± 0.08683695
WM	3.05 ± 0.7301767	2.8905 ± 0.622944	2.900656 ± 0.2752822

Table D.3: Average computational time in seconds of different combination schemes of the dFGN+WM algorithm.

Observed Time	Hurst Parameter		
	0.5	0.7	0.9
1.0	999.9034 ± 12.28178	1000.499 ± 44.02559	1200.653 ± 185.1612
0.1	999.8930 ± 12.27422	1000.490 ± 44.02604	1200.625 ± 185.1691
0.01	999.5118 ± 12.18654	1000.066 ± 44.15394	1199.793 ± 185.3497
0.001	959.2439 ± 13.72048	960.1988 ± 47.67838	1160.938 ± 190.7987
0.0001	67.57234 ± 8.464469	66.46729 ± 13.58390	283.4779 ± 104.0720

Table D.4: Output rate of the marginal distribution model with the exponential distribution.

Shape Parameter	Observed Time	Hurst Parameter		
		0.5	0.7	0.9
2.0	1.0	999.9290 ± 8.4541	1000.24 ± 32.3588	1107.156 ± 133.11
	0.1	999.9271 ± 8.4471	1000.236 ± 32.352	1107.153 ± 133.11
	0.01	999.8785 ± 8.3980	1000.257 ± 32.312	1107.019 ± 133.18
	0.001	983.5525 ± 9.7153	983.6565 ± 34.693	1077.676 ± 140.62
	0.0001	4.394532 ± 2.2161	4.913330 ± 2.2118	55.80139 ± 27.59
3.0	1.0	999.9378 ± 6.7957	1000.142 ± 26.787	1073.081 ± 109.70
	0.1	999.9361 ± 6.7919	1000.143 ± 26.784	1073.083 ± 109.69
	0.01	999.9420 ± 6.8885	1000.128 ± 26.794	1073.12 ± 109.776
	0.001	1004.487 ± 7.2810	1004.365 ± 28.842	1057.524 ± 115.50
	0.0001	0.3204321 ± 0.5306	0.3051759 ± 0.5489	11.26099 ± 7.6524
4.0	1.0	999.9420 ± 5.8253	1000.092 ± 23.359	1055.444 ± 95.568
	0.1	999.9338 ± 5.8236	1000.087 ± 23.352	1055.443 ± 95.572
	0.01	999.9558 ± 5.9164	1000.195 ± 23.306	1055.405 ± 95.594
	0.001	1019.440 ± 6.4407	1018.897 ± 24.935	1052.171 ± 100.19
	0.0001	0.02906438 ± 0.1799	0.0152588 ± 0.13374	2.258301 ± 2.6281

Table D.5: Output rate of the marginal distribution mode with the gamma distribution.

Variance Coef.	Hurst Parameter		
	0.5	0.7	0.9
200	1001.444 ± 4.756994	1001.387 ± 21.02250	1009.838 ± 84.65158
400	1014.853 ± 6.591091	1014.884 ± 28.38523	1044.513 ± 110.2697
600	1035.395 ± 7.960251	1035.602 ± 33.19263	1084.954 ± 127.2003
800	1058.613 ± 9.096699	1059.003 ± 36.86211	1125.510 ± 140.5151
1000	1082.704 ± 10.06144	1083.229 ± 39.88767	1164.804 ± 151.8021

Table D.6: Output rate of the storage mode.

D.3 Tables of the Hurst Estimation Results

This section lists seven tables whose data has been illustrated as the comparisons of the estimated results of the algorithms and models described in Chapter 5 and Chapter 6. All the tables contain three categories of values which obtain from the estimation using the R/S statistic, the VT plot, and Whittle's MLE. For every estimation, twenty different values of random seeds are applied to generate the sample traces with three different Hurst parameter settings, $H = 0.5$, 0.7, and 0.9, respectively. For each Hurst parameter setting and estimator, the mean, maximum, and minimum estimated results are listed.

The first three tables are related to the hybrid algorithms developed in Chapter 5. Table D.7 records the estimated results of the sample traces generated by the dFGN+RMD algorithm. Table D.8 shows the estimated results of the sample traces generated by the dFGN+SRA algorithm. Table D.9 tabulates the estimated results of the sample traces generated by the dFGN+WM algorithm.

The last four tables demonstrated the estimated results of self-similar traffic generated by the marginal distribution models or the storage models. The corresponding diagrams of the tables to present their comparisons are depicted in Chapter 6. Table D.10 shows the estimated results of the self-similar traffic traces generated by the marginal distribution model with the exponential distribution. Table D.11 records the estimated results of the self-similar traffic traces generated by the marginal distribution model with the gamma distribution. Table D.12 shows the estimated results of the self-similar traffic traces generated by the marginal distribution model with the Pareto distribution. The final table, Table D.13, records the estimated results of the self-similar traffic traces generated by the storage model.

Algorithm		Estimator								
		The R/S Statistic			The VT Plot			Whittle's MLE		
		Mean	Max.	Min.	Mean	Max.	Min.	Mean	Max.	Min.
Hybrid-1	0.5	0.5330582	0.63694	0.422039	0.4928803	0.604282	0.278580	0.5013138	0.5076785	0.4929571
	0.7	0.6993364	0.830322	0.619462	0.6503404	0.741686	0.462209	0.6732155	0.6798833	0.6640221
	0.9	0.7957637	0.906473	0.697599	0.7557175	0.833208	0.608908	0.917747	0.9237791	0.9077473
Hybrid-7	0.5	0.5408759	0.612844	0.41094	0.4502089	0.608289	0.315392	0.5002402	0.5050810	0.4939562
	0.7	0.7183227	0.841685	0.531305	0.6382837	0.802383	0.510442	0.6606467	0.6658607	0.6543213
	0.9	0.8175831	0.953205	0.659768	0.7995814	0.924986	0.696131	0.8956654	0.900548	0.8907477
Hybrid-31	0.5	0.5441724	0.640434	0.469029	0.4697073	0.579718	0.348149	0.4995443	0.5053032	0.4927166
	0.7	0.7204534	0.871656	0.6309010	0.648917	0.763593	0.51536	0.6606467	0.6658607	0.6543213
	0.9	0.8325940	0.996488	0.7400770	0.801971	0.897553	0.669625	0.8630934	0.8700969	0.8551826
Hybrid-255	0.5	0.5263000	0.615715	0.471559	0.4819442	0.579839	0.337841	0.4995863	0.5045281	0.4949122
	0.7	0.7007921	0.839071	0.642423	0.6707939	0.786234	0.491584	0.6553463	0.6605065	0.6511898
	0.9	0.8882616	0.982444	0.801800	0.8310895	0.946462	0.663424	0.835720	0.8414433	0.8301325
Hybrid-1023	0.5	0.5330018	0.621992	0.486049	0.4873751	0.565824	0.385994	0.5002689	0.5064949	0.4958549
	0.7	0.7222829	0.806372	0.663423	0.6796917	0.762139	0.575928	0.6569051	0.6642832	0.6502037
	0.9	0.9114342	0.986638	0.767364	0.8565019	0.939165	0.786752	0.8323695	0.8415256	0.8220247

Table D.7: Lists of the statistical results of the dFGN+RMD algorithm with different combination schemes.

Algorithm		Estimator								
		The R/S Statistic			The VT Plot			Whittle's MLE		
		Mean	Max.	Min.	Mean	Max.	Min.	Mean	Max.	Min.
Hybrid-1	0.5	0.5252614	0.609354	0.474016	0.477295	0.603344	0.293336	0.411317	0.416652	0.4075375
	0.7	0.693628	0.815644	0.594092	0.6332426	0.757555	0.443539	0.5385575	0.5459143	0.5327926
	0.9	0.8284624	0.953412	0.729966	0.7396106	0.858972	0.56883	0.6800033	0.6893448	0.6720444
Hybrid-7	0.5	0.4800244	0.552997	0.427222	0.4719111	0.60219	0.322208	0.3687004	0.3748608	0.3611006
	0.7	0.6600287	0.75515	0.576407	0.667022	0.808954	0.519056	0.4908085	0.4987745	0.4814003
	0.9	0.8278233	0.930045	0.739814	0.8225071	0.938942	0.690095	0.6291243	0.6419005	0.6162114
Hybrid-31	0.5	0.4273806	0.512969	0.360828	0.4382614	0.595865	0.281393	0.4253792	0.4360025	0.4173359
	0.7	0.6027112	0.682849	0.550304	0.6251828	0.789944	0.433223	0.5555359	0.5680006	0.5478726
	0.9	0.8064134	0.92568	0.72023	0.7897763	0.919115	0.606616	0.6961847	0.7097433	0.6895025
Hybrid-255	0.5	0.4272272	0.505371	0.360262	0.3992211	0.504774	0.288237	0.4802265	0.4868832	0.4739293
	0.7	0.5545192	0.622141	0.492467	0.585303	0.696714	0.469004	0.627455	0.634747	0.6206156
	0.9	0.7131069	0.835137	0.62468	0.7454433	0.850806	0.656991	0.7854442	0.7940317	0.7779401
Hybrid-1023	0.5	0.4751138	0.538992	0.394393	0.340039	0.438115	0.197896	0.4934448	0.5001705	0.4854259
	0.7	0.6067338	0.660098	0.514691	0.5027343	0.630709	0.331632	0.6457853	0.6528977	0.6363027
	0.9	0.7398409	0.824551	0.669486	0.6574458	0.797015	0.481771	0.8096867	0.8191196	0.7984878

Table D.8: Lists of the statistical results of the dFGN+SRA algorithm with different combination schemes.

Algorithm		Estimator								
		The R/S Statistic			The VT Plot			Whittle's MLE		
		Mean	Max.	Min.	Mean	Max.	Min.	Mean	Max.	Min.
Hybrid-1	0.5	0.5491411	0.598057	0.488724	0.4771256	0.604038	0.297126	0.5647237	0.5697552	0.5590367
	0.7	0.6984056	0.798682	0.627618	0.6280169	0.749473	0.414685	0.6872091	0.692639	0.6813637
	0.9	0.8229474	0.920732	0.722084	0.7387202	0.856772	0.56221	0.8260789	0.8319451	0.8201563
Hybrid-7	0.5	0.5628769	0.63771	0.485924	0.4766814	0.606478	0.337996	0.5239337	0.528933	0.5197466
	0.7	0.7083316	0.795884	0.617742	0.6688695	0.810614	0.518941	0.7080123	0.7129781	0.7025556
	0.9	0.7928071	0.894482	0.707318	0.8234072	0.939608	0.68983	0.882666	0.9003548	0.8653886
Hybrid-31	0.5	0.6134715	0.687927	0.511035	0.4860157	0.62954	0.322459	0.5452769	0.5504179	0.5284956
	0.7	0.687449	0.772618	0.610367	0.6506647	0.810489	0.451718	0.7513361	0.785287	0.7258121
	0.9	0.652564	0.808877	0.489221	0.801181	0.924889	0.617803	0.825466	0.9939223	0.7235813
Hybrid-255	0.5	0.7854026	0.894751	0.664238	0.585456	0.723497	0.456106	0.4623332	0.5261678	0.3979955
	0.7	0.7096165	0.929361	0.484106	0.705432	0.836202	0.582195	0.5623844	0.6865851	0.5038646
	0.9	0.2348948	0.520806	0.0118962	0.7873066	0.917605	0.641657	0.5676478	0.7572898	0.4950507
Hybrid-1023	0.5	0.8114819	0.886948	0.70856	0.683839	0.800167	0.573919	0.3652663	0.4552043	0.2544552
	0.7	0.7636195	0.853329	0.667847	0.7350849	0.871916	0.613795	0.4806178	0.5049363	0.4320374
	0.9	0.3262574	0.481523	0.110886	0.7339639	0.888347	0.576125	0.4978869	0.5739231	0.4513232

Table D.9: Lists of the statistical results of the dFGN+WM algorithm with different combination schemes.

Observed Time	Hurst Parameter	Estimator								
		The R/S Statistic			The VT Plot			Whittle's MLE		
		Mean	Max.	Min.	Mean	Max.	Min.	Mean	Max.	Min.
1.0	0.5	0.6107380	0.764366	0.400481	0.4662240	0.561242	0.351290	0.4989633	0.5037548	0.4950955
	0.7	0.6381003	0.843513	0.391815	0.5927740	0.728462	0.474160	0.6735684	0.6782339	0.6673955
	0.9	0.6742532	0.861899	0.442443	0.6797024	0.823091	0.546684	0.8672875	0.8742837	0.8592340
0.1	0.5	0.6106341	0.763334	0.400645	0.4662084	0.561304	0.351529	0.4989640	0.5037745	0.4951218
	0.7	0.6386530	0.849748	0.391693	0.5927865	0.728481	0.47414	0.67356880	0.6782125	0.6673838
	0.9	0.6744052	0.861370	0.442550	0.6797067	0.82309	0.546723	0.8672074	0.8741447	0.8592291
0.01	0.5	0.6110686	0.761258	0.393516	0.4663004	0.562986	0.357373	0.4989992	0.5037529	0.4951249
	0.7	0.6346905	0.829220	0.394249	0.5925208	0.728787	0.474152	0.6735300	0.6780985	0.6673546
	0.9	0.6755176	0.884347	0.441645	0.6797633	0.823164	0.546703	0.8672298	0.8742373	0.8590570
0.001	0.5	0.5773460	0.718295	0.372757	0.4706651	0.576698	0.358066	0.4992359	0.5043729	0.4954734
	0.7	0.6294545	0.792620	0.368192	0.5938170	0.728609	0.473284	0.6682505	0.6732770	0.6619373
	0.9	0.6512643	0.825295	0.421298	0.6814336	0.82384	0.551026	0.8550104	0.8624613	0.8448621
0.0001	0.5	N/A (11)	0.211895	0.060690	0.4822081	0.614187	0.370354	0.4822081	0.6141870	0.3703540
	0.7	N/A (15)	0.109821	0.0609203	0.5767227	0.711778	0.446135	0.6040443	0.6116647	0.5960951
	0.9	N/A (9)	0.329487	0.148200	0.6748547	0.827571	0.537637	0.7569481	0.7653617	0.7476773

Table D.10: Estimated Hurst values of the marginal distribution model with exponential distribution.

Observed Time (t)	Hurst Parameter (H)	Estimator								
		The R/S Statistic			The VT Plot			Whittle's MLE		
		Mean	Max.	Min.	Mean.	Max.	Min.	Mean	Max.	Min.
1.0	0.5	0.7177867	0.821467	0.527906	0.4622295	0.585761	0.337599	0.4991367	0.5040598	0.4947501
	0.7	0.7345818	0.856812	0.521362	0.5964018	0.738343	0.479201	0.6876511	0.6922073	0.6812362
	0.9	0.7730580	0.943982	0.548755	0.6842329	0.829486	0.561034	0.8844608	0.8897960	0.8762841
0.1	0.5	0.7178467	0.823262	0.526595	0.4621827	0.585801	0.338488	0.4991375	0.5040884	0.4947260
	0.7	0.7349108	0.854698	0.521558	0.5964269	0.728481	0.479285	0.6876451	0.6921930	0.6811964
	0.9	0.7724774	0.939972	0.548107	0.6854518	0.829503	0.561056	0.8841928	0.8898339	0.8763223
0.01	0.5	0.7164117	0.851492	0.523017	0.4612206	0.585907	0.331248	0.4991101	0.5044946	0.4947022
	0.7	0.7372988	0.883545	0.516876	0.5962802	0.737311	0.476409	0.6872309	0.6921169	0.6812218
	0.9	0.7695158	0.952415	0.545564	0.6854696	0.829310	0.561023	0.8833520	0.8887089	0.8756568
0.001	0.5	0.6628613	0.816297	0.448120	0.4688107	0.587490	0.363940	0.4992824	0.5036948	0.4949762
	0.7	0.6875597	0.811138	0.465378	0.5938244	0.742107	0.468510	0.6599142	0.6657296	0.6550630
	0.9	0.7498452	0.880862	0.516868	0.6856013	0.827016	0.564598	0.8335332	0.8401832	0.8281866
0.0001	0.5	N/A (20)	N/A	N/A	N/A (5)	0.522554	0.486942	N/A (5)	0.5081506	0.4998740
	0.7	N/A (19)	0.00128649	0.00128649	N/A (7)	0.507776	$-\infty$	N/A (7)	0.5000803	0.4998890
	0.9	N/A (20)	N/A	N/A	0.5581031	0.641594	0.442130	0.5915278	0.7015376	0.5358070

Table D.11: Estimated Hurst values of the marginal distribution model with gamma distribution with $s = 3$.

Shape Parameter (α)	Hurst Parameter (H)	Estimator								
		The R/S Statistic			The VT Plot			Whittle's MLE		
		Mean	Max.	Min.	Mean	Max.	Min.	Mean	Max.	Min.
0.5	0.5	0.05752117	0.134744	-0.00442782	0.4861691	0.591333	0.421116	0.5000395	0.5068442	0.499388
	0.7	0.04679622	0.136791	0.0145646	0.4979644	0.587814	0.444229	0.5014637	0.5186941	0.4997899
	0.9	0.09685982	0.253614	0.0339827	0.5760635	0.685223	0.44582	0.6363057	0.7150241	0.5708171
1.5	0.5	0.3639563	0.55384	0.188796	0.4898432	0.624628	0.381592	0.4998083	0.5063545	0.4943725
	0.7	0.3504521	0.500241	0.118798	0.5421314	0.636012	0.343208	0.5381108	0.5799088	0.5047121
	0.9	0.1248177	0.309225	0.0152414	0.5226587	0.640375	-0.12124	0.5815697	0.8462587	0.5046908
2.5	0.5	0.6422989	0.750552	0.457055	0.4763194	0.583078	0.347142	0.4987997	0.5059892	0.4936714
	0.7	0.6743605	0.861782	0.406644	0.5753622	0.69321	0.441572	0.606213	0.6305955	0.5642947
	0.9	0.3924275	0.652172	0.09164	0.619329	0.714574	0.418948	0.6746312	0.7950448	0.5568876

Table D.12: Estimated Hurst values of the marginal distribution model with Pareto distribution with $k = 10$.

Variance Coefficient (a)	Hurst Parameter (H)	Estimator								
		The R/S Statistic			The VT Plot			Whittle's MLE		
		Mean	Max.	Min.	Mean	Max.	Min.	Mean	Max.	Min.
200	0.5	0.7630548	0.857818	0.621881	0.4594993	0.610879	0.334853	0.4995216	0.5040745	0.494526
	0.7	0.791418	0.885841	0.615235	0.5988877	0.747895	0.48088	0.6959435	0.7007076	0.6898334
	0.9	0.8191873	0.936771	0.615662	0.6899063	0.83195	0.584351	0.8950935	0.9005004	0.8891128
400	0.5	0.7116669	0.821673	0.543057	0.4576488	0.610425	0.327111	0.499375	0.5041967	0.4944795
	0.7	0.753276	0.869643	0.546458	0.5980851	0.746571	0.479918	0.6939219	0.6983851	0.6878879
	0.9	0.7834781	0.902776	0.6893107	0.549167 0.832925	0.578046	0.8902937	0.8958914	0.885983	
600	0.5	0.6896741	0.795674	0.494555	0.457074	0.606604	0.323941	0.4993575	0.5043906	0.4942769
	0.7	0.7246042	0.834431	0.510964	0.5971982	0.744773	0.478205	0.6917575	0.696243	0.6858566
	0.9	0.7691949	0.904514	0.533887	0.6886233	0.833018	0.573659	0.886851	0.8926934	0.8830039
800	0.5	0.6742702	0.769513	0.469846	0.4573754	0.60511	0.319638	0.49934	0.5044525	0.4943486
	0.7	0.7082325	0.818984	0.487033	0.5965881	0.74311	0.47719	0.6898791	0.6944863	0.6841192
	0.9	0.7595101	0.905017	0.522278	0.6880806	0.832813	0.570892	0.8843081	0.8902571	0.8800214
1000	0.5	0.6635603	0.754264	0.455463	0.4582828	0.603942	0.323892	0.4993404	0.5044001	0.494462
	0.7	0.6962375	0.79693	0.471924	0.5960634	0.741805	0.475842	0.6882887	0.6931015	0.6825434
	0.9	0.7490424	0.89046	0.516395	0.687645	0.832696	0.568982	0.8823729	0.8885116	0.8778288

Table D.13: Estimated Hurst values of storage model with different variance coefficients.

REFERENCES

[Ali96] T. Alieva. Fractional Fourier transform as a tool for investigation of fractal objects. *J. Optical Soc. of America A - Optics Science and Vision*, 13(6):1189–1192, June 1996.

[Ana95] V. Anantharam. On the sojourn time of sessions at an ATM buffer with long-range dependence input traffic. In *Proc. 34th IEEE Conference in Decision and Control*, December 1995.

[Arn83] B. Arnold. *Pareto Distribution*, volume 5 of *Statistical Distributions in Scientific Work*, chapter 3–5. International Cooperative Publishing House, Maryland, 1983.

[AZN95] R. G. Addie, M. Zukerman, and T. Neame. Performance of a single server queue with self-similar input. In *Proc. ICC*, pages 461–465, Seattle, U.S., June 1995.

[Bar93] M. Barnsley. *Fractals Everywhere*, chapter 4, 5. Academic Press, Boston, 2 edition, 1993.

[BD91] P. J. Brokwell and R. A. Davis. *Time Series: Theory and Methods*, chapter 1, 3, 8, 12. Springer Series in Statistics. Springer-Verlag, New York, 1991.

[Ber90] L. Berry. ATM traffic modelled by a semi-Markov point process source. In *Proc. of Austrialian Video Communications Workshop*, pages 27–35, Melbourne, July 1990.

[Ber94] J. Beran. *Statistics for Long-Momory Processes*, volume 61 of *Monographs on Statistics and Applied Probability*, chapter 1–7. Chapman & Hall, New York, 1994.

[BJ76] G. E. P. Box and G. M. Jenkins. *Time Series Analysis: Forecasting and Control*, chapter 2–4. Prentice-Hall, Englewood Cliffs, NJ, 1976.

[BL80] M. V. Berry and Z. V. Lewis. On the Weierstrass-Mandelbrot fractal function. *Proc. Royal Society London A*, 370:459–484, 1980.

[BL97] M. Bromirski and W. Lobejko. Fractals and chaos in description of multimedia ATM traffic. In *Proc. of 5th IFIP Workshop on Performance Modelling and Evaluation of ATM Network*, pages T06/1 – T06/20, Bradford, Uk, July 1997.

[BR95] B. Bobee and P. F. Rasmussen. Recent advances in flood frequency analysis. *Reviews of Geophysics*, 33(2):1111–1116, July 1995.

[Bre68] L. Breiman. *Probability*, chapter 7, 10, 12, 15, 16. Addison-Wesley Series in Statistics. Addison-Wesley, Menlo Park, CA, 1968.

[BRSV96] F. Brichet, J. Roberts, A. Simonian, and D. Veitch. Heavy traffic analysis of a fluid queue fed by on/off sources with long-range dependence. *Queueing Systems*, 23:197–215, 1996.

[BSTW95] J. Beran, R. Sherman, M. S. Taqqu, and W. Willinger. Long-range dependence in variable-bit-rate video traffic. *IEEE Trans. on Commun.*, 43(2/3/4):1566–1579, Feb./Mar./Apr. 1995.

[CB96] M. E. Crovella and A. Bestavros. Self-similarity in world wide web traffic evidence and possible causes. In *Proc. SIGMETRICS '96*, pages 160–169, PA, U.S.A., 1996.

[Cin75] E. Cinlar. *Introduction to Stochastic Processes*, chapter 4–10. Printice-Halll, Englewood Cliffs, NJ, 1975.

[Cox84] D. R. Cox. Long-range dependence: A review. In H. A. David and H. T. David, editors, *Statistics: An Appraisal*, pages 55–74. Iowa State Statistical Laboratory, The Iowa State University Press, 1984.

[CPL+94] J. P. Cosmas, G. H. Petit, R. Lehnert, C. Blondia, K. Kontovassilis, O. Casals, and T. Theimer. A review of voice, data, and video traffic models for ATM. *Euro. Trans. Telecommun.*, 5(2):139–154, Mar.-Apr. 1994.

[CTB97] M. Crovella, M. S. Taqqu, and A. Bestravos. Heavy-tailed probability distributions in the world wide web. In R. Adler, R. Feldman, and M. S. Taqqu, editors, *A Practical Guide to Heavy Tails: Statistical Techniques for Analysing Heavy Tailed Distributions.* Birkhäuser, Boston, 1997.

[De 95] M. De Prycker. *Asynchronous Transfer Mode: Solution for Broadband ISDN*, chapter 2, 3, 4, 7. Printice, London, 3 edition, 1995.

[Dev87] R. L. Devaney. *Introduction ot Chaotic Dynamic Systems*, chapter 1. Addison-Wesley, Redwood City, CA, 1987.

[DLRT95] N. G. Duffield, J. T. Lewis, R. Russell, and F. Toomy. Predicting quality of service for traffic with long-range fluctuation. In *Proc. ICC*, pages 473–477, Seattle, U.S., June 1995.

[DMRW94] D. E. Duffy, A. A. McIntosh, M. Rosenstein, and W. Willinger. Statistical analysis of CCSN/SS7 traffic data from working subnetworks. *IEEE J. Select. Areas Commun.*, 12(3):544–551, April 1994.

[DO95] N. G. Duffield and N. O'Connel. Large deviation and overflow probabilities for the general single-server queue, with applications. *Mathematical Proc. Cambridge Philosophical Society*, 118:363–374, 1995.

[DS96] S. Dastangoo and V. K. Samalam. Preliminary simulation results of self-similar traffic over ATM ABR service. Tech. Report atm96-0026, ATM Forum, 1996.

[Dyn65] E. B. Dynkin. *Markov processes*, volume 1, chapter 3, 5. Springer-Verlag, Berlin, 1965.

[EGW94] A. Erramilli, J. Gordon, and W. Willinger. Applications of fractals in engineering for realistic traffic processes. In *Proc. 14th ITC*, pages 35–44, Antibes Juan-les-Pins, Fracnce, June 1994.

[ENW96] A. Erramilli, O. Narayan, and W. Willinger. Experimental queueing analysis with long-range dependent packet traffic. *IEEE/ACM Trans. Networking*, 3(3):209–223, 1996.

[EPW95] A. Erramilli, P. Pruthi, and W. Willinger. Recent developments in fractal traffic modeling. In *Proc. International Teletraffic Seminar*, St. Peterburg, June 1995.

[ESP94] A. Erramilli, R. P. Singh, and P. Pruthi. Chaotic maps as models of packet traffic. In *Proc. ITC 14*, pages 329–338, Antibes Juan-les-Pins, Fracnce, June 1994.

[EW94] A. Erramilli and J. L. Wang. Monitoring packet traffic levels. In *Proc. IEEE GLOBECOM*, pages 274–280, San Francisco, CA, December 1994.

[EWP94] A. Erramilli, W. Willinger, and P. Pruthi. Fractal traffic flows in high-speed communications networks. *Fractals*, 2(3):409–412, September 1994.

[Fed88] J. Feder. *Fractals*, chapter 1, 8, 9, 10, 13. Plenum Press, New York, NY, 1988.

[Fel71] W. Feller. *An Introduction to Probability Theory and Its Applications*, volume 2. John Wiley & Sons, New York, 1971.

[FFC82] A. Fournier, D. Fussell, and L. Carpenter. Computer rendering of stochastic models. *Commun. of ACM*, 25(6):371–384, June 1982.

[FL91] H. J. Fowler and W. E. Leland. Local area network traffic characteristics, with implications for broadband network congestion management. *IEEE J. Select. Areas Commun.*, 9(7):1139–1149, September 1991.

[Fla92] P. Flandrin. Wavelete analysis and synthesis of fractional Brownian motion. *IEEE Trans. on Info. Theory*, 38(2):910–917, March 1992.

[FM94] V. S. Frost and B. Melamed. Traffic modelling for telecommunications networks. *IEEE Commun. Mag.*, 32(3):70–81, March 1994.

[For96] The ATM Forum. Traffic management 4.0. Specification, The ATM Forum, April 1996.

[Fre83] D. Freeman. *Brownian Motion and Diffusion*, chapter 1, 2. Springer-Verlag, New York, 1983.

[Gar93] W. M. Garrett. *Contributions Toward Real-Time Services on Packet-Switched Networks*. PhD thesis, Columbis University, New York, May 1993.

[GCMOO91] R. Grunefelder, J. Cosmas, S. Manthorpe, and A. Odinma-Okafor. Characterisation of video codecs as autoregressive moving avergae processes and related queueing system performance. *IEEE J. Select. Areas Commun.*, 8(3):284–293, 1991.

[GH97] J. J. Gordon and F. Huebner-Szabo de Bucs. A model of self-similar data traffic applied to Ethernet traces. In *Proc. of ATM'97*, pages 14/1 – 14/10, Ilkley, UK, July 1997.

[Gin96] D. Ginsburg. *ATM: Solutions fro Enterprise Internetworking*, chapter 1–3. Data Communications and Networks Series. Addison-Wesley, Harlow, England, 1996.

[GJ80] C. W. J. Granger and R. Joyeux. An introduction to long-memory time series models and fractionall differencing. *J. of Time Series Analysis*, 1(1):15–29, 1980.

[GK89] B. V. Gnedenko and I. N. Kovalenko. *Introduction to Queueing Theory*, chapter 1, 3, 4. Birkhauser, Boston, 1989.

[GMF96] E. Gelenbe, X. Mang, and Y. Feng. A diffusion cell loss estimate for ATM with multiclass bursty traffic. In D. Kouvatsos, editor, *ATM Networks: Performance Modelling and Evaluation*, volume 2, pages 233–248. Chapman & Hall, London, 1996.

[GMO97] E. Gelenbe, X. Mang, and R. Onv. Bandwidth allocation and call admission control in high-speed network. *IEEE Commun. Mag.*, 35(5):122–129, May 1997.

[Gor95] J. J. Gordon. Pareto process as a model of self-similar packet traffic. In *Proc. GLOBECOM'95*, pages 2232–2236, Singapore, 1995.

[Gru82] J. G. Gruber. A comparison of measured and calculated speech temporal parameteres revelant to speech activity detection. *IEEE Trans. on Commun.*, 30(4):728–738, April 1982.

[GSV95] X. Gu, K. Sohraby, and D. R. Vaman. *Control and Performance in Packet, Circuit, and ATM Networks*. Kiuwer Academic, BOston, 1995.

[Gus90] R. Gusella. A measurement study of diskless workstation traffic on an Ethernet. *IEEE Trans. on Commun.*, 38(9):1557–1568, September 1990.

[GW94] M. W. Garrett and W. Willlinger. Analysis, modeling and generation of self-similar VBR video traffic. In *Proc. ACM SIGCOMM*, pages 269–280, London, UK, August 1994.

[HDLK95a] C. Huang, M. Devetsikiotis, I. Lambadaris, and A. R. Kaye. Fast simulation for self-similar traffic in ATM networks. In *Proc. ICC*, pages 438–444, Seattle, U.S., June 1995.

[HDLK95b] C. Huang, M. Devetsikiotis, I. Lambadaris, and A. R. Kaye. Modeling and simulation of self-similar variable bit rate compressed video: A unified approach. *Computer Communication Review*, 25(4):114–125, October 1995.

[HL86] H. Heffes and D. M. Lucantoni. A Markov modulated characterization of packetized voice and data traffic and related statistical multiplexer performance. *IEEE J. Select. Areas Commun.*, SAC-4(6):856–868, September 1986.

[Hos81] J. R. M. Hosking. Fractional differencing. *Biometrika*, 68(1):165–176, April 1981.

[HRS96] D. Heath, S. Resnick, and G. Samorodnitsky. Heavy tails and long range dependence in ON/OFF processes and assocaited fluid models. Technical Report 1144, School of Operations Research and Industrial Engineering, Cornell University, January 1996.

[HS93] H. M. Hastings and G. Sugihara. *Fractals: A User's Guide for the Natural Sciences*, chapter 2, 5, 12. Oxford Univ. Press, Oxford, 1993.

[HTL92] D. Heyman, A. Tabatabai, and T. V. Lakshman. Statistical analysis and simulation study of video teletraffic in ATM networks. *IEEE Trans. Circuit and Systems for Video Technology*, 2(1):49–59, March 1992.

[Hur51] H. E. Hurst. Long-term storage capacity of reservoirs. *Trans. Amer. Soc. Civil Engineers*, 116:770–799, 1951.

[IM65] K. Itô and H. P. McKean, Jr. *Diffusion Processes and Their Sample Paths*, chapter 1–3. Springer-Verlag, Berlin, 1965.

[IT93a] ITU-T. B-ISDN ATM layer specification. Recommendation I.361, International Telecomm. Union, March 1993.

[IT93b] ITU-T. BISDN ATM adaption layer (AAL) funcational description. Recommendation I.362, International Telecomm. Union, March 1993.

[IT93c] ITU-T. Traffic control and congestion control in B-ISDN. Recommendation I.371, International Telecomm. Union, March 1993.

[JK70] N. L. Johson and S. Kotz. *Distributions in Statistics: Continuous Univariate Distributions - I*, volume 2, chapter 17–20. John Wiley & Sons, New York, 1970.

[JK92] N. L. Johson and S. Kotz. *Univariate Discrete Distributions*, chapter 10. John Wiley & Sons, New York, 2 edition, 1992.

[JR86] R. Jain and S. A. Routhier. Packet trains: Measurements and a new model for computer network traffic. *IEEE J. Select. Areas Commun.*, SAC-4(6):986–995, September 1986.

[KJ94] H. J. Korsch and H.-J. Jodl. *Chaos: A Program Collection for the PC*. Springer-Verlag, Berlin, 1994.

[Kle75] L. Kleinrock. *Queueing Systems*, volume 1, chapter 1–4. John Wiley & Sons, Inc., New York, 1975.

[Kor92] G. Korvin. *Fractal Models in the Earth Science*, chapter 4. Elsevier, Amsterdam, 1992.

[LEWW95] W.-C. Lau, A. Erramilli, J. L. Wang, and W. Willinger. Self-similar traffic generation: The random midpoint displacement algorithm and its properties. In *Proc. ICC*, pages 466–472, Seattle, U.S., June 1995.

[LL92] J. W. Lee and B. G. Lee. Performance analysis of ATM cell multiplexer with MMPP input. *IEICE Trans. Commun.*, E75-B(8):709–714, August 1992.

[LMHN90] D. M. Lucantoni, K. S. Meier-Hellstern, and M. F. Neuts. A single server queue with server vacations and a class of non-renewal arrival processes. *Adv. Appl. Prob.*, 22(3):676–705, 1990.

[LO86] W. Leland and T. Ott. Load-balancing heuristics and process behavior. In *PERFORMANCE '86 and ACM SIGMET-RISC 1986 Joint Conference on Computer Performance Modelling, Measurement and Evaluation*, pages 54–69, North Carolina State University, May 1986.

[LTG95] N. Likhanov, B. Tsybakov, and N. D. Georganas. Analysis of an ATM buffer with self-similar("fractal") input traffic. In *Proc. IEEE INFOCOM'95*, pages 985–992, Los Alamitos, CA, April 1995. IEEE Computer Society Press.

[LTWW93] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson. On self-similar nature of Ethernet traffic. In *Proc. ACM SIGCOMM*, pages 1–11, San Francisco, CA, USA, September 1993.

[LTWW94] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson. On self-similar nature of Ethernet traffic (extended version). *IEEE/ACM Trans. Networking*, 2(1):1–13, February 1994.

[LW91] W. E. Leland and D. V. Wilson. High time-resolution measurement and analysis of LAN traffic:implications for LAN interconnection. In *Proc. IEEE INFOCOM*, pages 1360–1366, Bal Harbour, April 1991.

[LWTW94] W. E. Leland, W. Willinger, M. S. Taqqu, and D. V. Wilson. Statistical analysis and stochastic modeling of self-similar datatraffic. In *Proc. ITC 14*, pages 319–328, Antibes Juan-les-Pins, Fracnce, June 1994.

[MA97] R. Macfadyen and H. Azmoodeh. Statistical characterisation of WAN traffic. In *Proc. of ATM'97*, pages 12/1 – 12/10, Ilkley, UK, July 1997.

[Man63] B. B. Manelbrot. New methods in statistical economics. *J. Political Economy*, 71(5):421–440, October 1963.

[Man69] B. B. Mandelbrot. Long-run linearity, locally gaussian processes, H-spectra and infinite variances. *Intern. Econom. Rev.*, 10(1):82–113, February 1969.

[Man71] B. B. Mandelbort. A fast fractional gaussian noise generator. *Water Resources Research*, 7(3):543–553, 1971.

[Man82] B. B. Mandelbrot. Comment on computer rendering of fractal stochastic models. *Commun. of ACM*, 25(8):581–583, August 1982.

[Man83] B. B. Mandelbrot. *The Fractal Geometry of Nature*, chapter 1, 2, 4, 7-9, 11. W. H. Freeman and Company, San Francisco, CA, 1983.

[MH89] K. S. Meier-Hellstern. The analysis of a queue arising in overflow models. *IEEE Trans. on Commun.*, 37(4):367–372, 1989.

[MHWYH91] K. Meier-Hellstern, P. E. Wirth, Y.-L. Yan, and D. A. Hoeflin. Traffic models for ISDN data users: Office automation applications. In *Proc. 13th ITC*, pages 167–172, Copenhagen, Denmark, 1991.

[MM85] W. T. Marshall and S. P. Morgan. Statistics of mixed data traffic on local area network. *Computer Network and ISDN Systems*, 10:185–194, 1985.

[MN68] B. B. Mandelbort and J. W. Van Ness. Fractional Brownian motions, fractional noises and applications. *SIAM Review*, 10(4):422–437, October 1968.

[MRR97a] H. Madsen, P. F. Rasmussen, and D. Rosberg. Comparison of annual maximum series and partial duration series methods for modeling extreme hydrologic events: 1. at-site modeling. *Water Resources Research*, 33(4):747–757, April 1997.

[MRR97b] H. Madsen, P. F. Rasmussen, and D. Rosberg. Comparison of annual maximum series and partial duration series methods for modeling extreme hydrologic events: 2. regional modeling. *Water Resources Research*, 33(4):759–769, April 1997.

[MS86] J. R. Michael and W. R. Schucany. Analysis of data from censored samples. In R. B. D'Agostino and M. A. Stephens, editors, *Goodness-of-Fit Techniques*, volume 68 of *Statistics*, pages 461–496. Marcel Dekker Inc., New York, 1986.

[MS97a] G. Mayor and J. Silvester. Performance modeling network management for self-similar traffic. In *Proc. of ATM'97 (Tutorial)*, pages T03/1 – T03/23, Ilkley, UK, July 1997.

[MS97b] G. Mayor and J. Silvester. Time scale analysis of an ATM queueing system with long-range dependent traffic. In *Proc. of IEEE INFOCOM '97*, Kobe, Japan, April 1997.

[MW69] B. B. Mandelbrot and J. R. Wallis. Computer experiments with fractional Gaussian noises. *Water Resources Research*, 5:228–267, 1969.

[Neu79] M. F. Neuts. A versatile Markov point process. *J. Applied Probability*, 16(4):764–779, 1979.

[Neu92] M. F. Neuts. Models based on the Markovian arrival process. *IEICE Trans. Commun.*, E75-B(12):1255–1265, December 1992.

[Nor94] I. Norros. A storage model with self-similar input. *Queueing Systems*, 16:387–392, 1994.

[Nor95a] I. Norros. The management of large flows of connectionless traffic on the basis of self-similar modeling. In *Proc. ICC*, pages 451–455, Seattle, U.S., June 1995.

[Nor95b] I. Norros. On the use of fractional brownian motion in the theory of connectionless networks. *IEEE J. Select. Areas Commun.*, 13(6):953–962, August 1995.

[Onv95] R. O. Onvural. *Asynchronous Transfer Mode Networks: Performance Issues*, chapter 1–4. Artech House, Norwood, MA, 2 edition, 1995.

[Pap91] A. Papoulis. *Probability, Random Variables, and Stochastic Processes*, chapter 9, 12. McGraw-Hill, New York, NY, 3 edition, 1991.

[Pax94] V. Paxson. Emperically derived analytic models of wide-area TCP connections. *IEEE/ACM Trans. Networking*, 2(4):316–336, August 1994.

[Pax95] V. Paxson. Fast approximation of self-similar network traffic. Technical report, Lawrence Berkeley Lab., April 1995.

[PF95] V. Paxson and S. Floyd. Wide-area traffic: The failure of Poisson modeling. *IEEE/ACM Trans. Networking*, 3(3):226–244, June 1995.

[PFTV92] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge Univ. Press, Cambridge, 2 edition, 1992.

[PJS92] H. Peitgen, H. Jürgens, and D. Saupe. *Chaos and Fractals: New Frontiers of Science*, chapter 2–4, 9–12. Springer-Verlaag, 1992.

[PM96] M. Parulekar and A. Makowski. Tail probabilities for az multiplexer with self-similar traffic. In *Proc. of IEEE INFOCOM'96*, pages 1452–1459, San Francisco, March 1996.

[PP97] P. Pruthi and A. Popescu. Effect of controls on self-similar traffic. In *Proc. of ATM'97*, pages 25/1 – 25/10, Ilkley, UK, July 1997.

[Pru95] P. Pruthi. Heavy-tailed ON/OFF source behavior and self-similar traffic. In *Proc. ICC*, pages 445–450, Seattle, U.S., June 1995.

[Ran94] N. Rananand. Approximating a variable bit rate source by Markov processes. In *Proc. of IEEE GLOBECOM*, pages 1107–1112, San Francisco, December 1994.

[RBL97] G. Rozanski, M. Bromirski, and W. Lobejko. Chaotic modelling of VBR video traffic. In *Proc. of 5th IFIP Workshop on Performance Modelling and Evaluation of ATM Network*, pages 18/1 – 18/10, Bradford, Uk, July 1997.

[RE96] B. Ryu and A. Elwalid. The importance of long-range dependence of VBR video traffic in ATM traffic engineering: Myths and realities. In *Proc. of ACM SIGCOMM '96*, pages 3–14, August 1996.

[Res92] S. I. Resnick. *Adventures in Stochastic Processes*, chapter 2–6. Birkhäuser, Boston, 1992.

[Res97] S. I. Resnick. Heavy tail modelling and teletraffic data. *Annals Statistics*, 25(5):1805–1869, October 1997.

[Ric61] L. F. Richardson. The problem of contiguity: an appendix of statistics of delay quarrels. In *General System Yearbook*, volume 6, pages 139–187. General System Yearbook, 1961.

[RK94] Q. Ren and H. Kobayashi. Diffusion process approximations of a statistical multiplexer with Markov modulated bursty traffic sources. In *Proc. of IEEE GLOBECOM*, pages 1100–1106, San Francisco, December 1994.

[Ros84] M. Rosenblatt. Stochastic processes with short-range and long-range dependence. In H. A. David and H. T. David, editors, *Statistics: An Appraisal*, pages 509–520. Iowa State Statistical Laboratory, The Iowa State University Press, 1984.

[RP94] S. Rambaldi and O. Pinazza. An accurate fractional Brownian motion generator. *Physica A*, 208(1):21–30, July 1994.

[RS95] J. Rosinski and G. Samorodnitsky. Classes of mixing stable processes. Technical Report 1143, School of Operations Research and Industrial Engineering, Cornell University, June 1995.

[Sai94] H. Saito. *Teletraffic Technologies in ATM Networks*, chapter 2–4. Artech House, Boston, 1994.

[Sau88] D. Saupe. Algorithms for random fractals. In H.-O. Peitgen and D. Saupe, editors, *The Science of Fractal Images*, chapter 2, pages 71–136. Spring-Verlag, New York, 1988.

[SKY91] H. Saito, M. Kawaraski, and H. Yamada. An analysis of statistical multiplexing in an ATM transport network. *IEEE J. Select. Areas Commun.*, 9(3):359–367, April 1991.

[SMRA89] P. Sen, B. Maglaris, N. Rikli, and D. Anastassiou. Models for package switching of variable-bit-rate video sources. *IEEE J. Select. Areas Commun.*, 7(5):865–869, June 1989.

[ST94] G. Samorodnitsky and M. S. Taqqu. *Statble Non-Gaussian Random Processes: Stochastic Modeling with Infinite Variance*, chapter 7. Stochastic Modeling. Chapman & Hall, New York, 1994.

[Sta92] W. Stallings. *ISDN and Broadband ISDN*, chapter 5, 13. MacMillan, New York, 2 edition, 1992.

[SW86] K. Sriram and W. Whitt. Characterizing superposition arrival processes in packet multiplexers for voice and data. *IEEE J. Select. Areas Commun.*, 4(6):833–846, September 1986.

[TG97] B. Tsybakov and N. D. Georganas. On self-similar traffic in ATM queues: Definitions, overflow probability bound, and cell delay distribution. *IEEE/ACM Trans. Networking*, 5(3):397–409, June 1997.

[TTW95] M. S. Taqqu, V. Teverovsky, and W. Willinger. Estimators for long-range dependence: An emperical study. *Fractals*, 3(4):758–798, 1995.

[TWS97] M. S. Taqqu, W. Willinger, and R. Sherman. Proof of a fundamental result in self-similar traffic modeling. *Computer Communication Review*, 27:5–23, 1997.

[VKH96] B. Venkateshware, K. R. Krishnan, and D. P. Heyman. Performance of finite-buffer queues under traffic with long-range dependence. In *Proc. GLOBECOM 96*, volume 1, pages 601–605, London, 1996.

[Vos85a] R. F. Voss. Random fractal forgeries. In R. A. Earnshaw, editor, *Fundamental Algorithms in Computer Graphics*, pages 805–835. Springer-Verlag, Berlin, 1985.

[Vos85b] R. F. Voss. Random fractals: Characterization and measurement. In R. Pynn and A. Skjeltorp, editors, *Scaling Phenomena in Disordered Systems*, pages 1–11. Plenum Press, New York, NY, 1985.

[Vos88] R. F. Voss. Fractals in nature: From characterization to simulation. In H.-O. Peitgen and D. Saupe, editors, *The Science of Fractal Images*, chapter 1, pages 21–70. Spring-Verlag, New York, 1988.

[Wil95] W. Willinger. Traffic modeling for high-speed networks: Theory versus practice. In F. P. Kelly and R. J. Williams, editors, *Stochastic Networks*, pages 395–409. Springer-Verlag, Berlin, 1995.

[WTE96] W. Willinger, M. S. Taqqu, and A. Erramilli. A bibliograsphical guide to self-similar traffic and performance modeling for morden high-speed network. In F. P. kelly, S. Zachary, and I. Ziedins, editors, *Stochastic networks: Theory and Applications*, pages 339–366. Clarendon Press, Oxford, 1996.

[WTLW95] W. Willinger, M. S. Taqqu, W. E. Leland, and D. V. Wilson. Self-similarity in high-speed packet traffic: Analysis and modeling for Ethernet traffic measurements. *Statistical Science*, 10(1):67–85, 1995.

[WTSW95] W. Willinger, M. S. Taqqu, R. Sherman, and D. V. Wilson. Self-similar through high-variability: Statistical analysis of Ethernet LAN traffic at the source level. In *Proc. of ACM/SIGCOMM'95*, pages 100–113, 1995.

[WTSW97] W. Willinger, M. S. Taqqu, R. Sherman, and D. V. Wilson. Self-similar through high-variability: Statistical analysis of Ethernet LAN traffic at the source level. *IEEE/ACM Trans. Networking*, 5(1):71–86, February 1997.

[Yul45] G. U. Yule. On a mthod of studying time series based on their internal correlation. *J. Royal Stat. Soc.*, 108:208–225, 1945.