



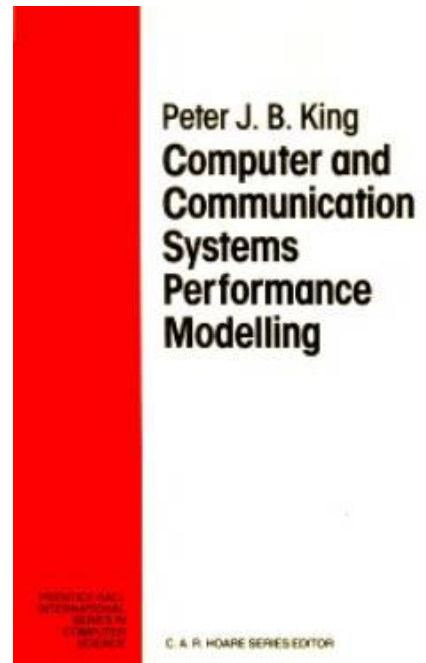
**COMPUTER AND
COMMUNICATION SYSTEMS
PERFORMANCE MODELLING**
PETER J.B.KING

SUMMARY

Computer and Communication Systems Performance Modelling provide an introduction to the field of quantitative analysis of computer and communication system performance. The book introduces some of the most powerful mathematical tools for analysing queueing systems and applies them to realistic examples.

Readers will find detailed considerations of Little's theorem and its consequences, analyses of M/M/1, M/G/1 and discrete time queues showing all steps of analysis. There is also extensive coverage of BLMP networks and algorithms for their analysis. A unique feature is the treatment of numerical methods for queueing system solution. Each chapter has a bibliography and most contain exercises.

This text is suitable for students of Computer Science as well as practising Communications Engineers.



CONTENTS

Preface		xii
1	Introduction	1
1.1	History	2
1.2	Performance measures	3
1.3	Notation	3
1.4	D/D/1 queue	5
1.5	Little 's theorem	6
	1.5.1 Proof of Little's theorem	9
1.6	Applications of Little 's theorem	11
	1.6.1 G/G/1 queue	11
	1.6.2 Time sharing system response time	13
1.7	Further reading	14
1.8	Bibliography	15
2	Probability theory	18
2.1	Axioms of probability	18
	2.1.1 Conditional probability	20
	2.1.2 Bayes theorem	21
2.2	Random variables	22
2.3	Discrete random variables	22
	2.3.1 Discrete distributions	24
2.4	Continuous random variables	26
	2.4.1 Continuous distributions	28
2.5	Generating functions and Laplace transforms	28
2.6	Exercises	31
2.7	Further reading	32
2.8	Bibliography	32
3	Stochastic processes	34
3.1	Poisson process	34
3.2	Random walk	36
3.3	Markov processes and chains	37
	3.3.1 Markov chains	37
	3.3.2 State classification	39
	3.3.3 Stationary distribution	39
	3.3.4 Markov processes	39
	3.3.5 Local balance and time reversibility	40
3.4	Renewal theory	41

3.4	Renewal theory	44
3.5	Proof of Little's theorem	43
3.6	Further reading	46
3.7	Bibliography	46
4	Simple queues	48
4.1	M/M/1 queue	48
4.2	Steady state diagrams	52
4.3	Birth-and-death processes	52
4.4	Limited waiting room	53
4.5	Finite customer population	55
4.6	Discrete time queues	57
4.7	Exercises	60
4.8	Further reading	61
4.9	Bibliography	61
5	M/G/1 queues	62
5.1	Mean queue length	62
5.2	Embedded Markov chain	65
5.2.1	Khintchine's argument	67
5.3	Tagged jobs	67
5.4	Random observations	69
5.5	M/G/1 queues with server vacations	71
5.5.1	Disk sector queueing	73
5.6	Exercises	75
5.7	Further reading	76
5.8	Bibliography	76
6	Queues with breakdowns	78
6.1	M/M/1 queue with breakdowns	78
6.2	M/G/1 queue with breakdowns	81
6.3	Exercises	83
6.4	Further reading	83
6.5	Bibliography	84
7	Priority queues	85
7.1	Non-preemptive priority queue	85
7.2	Preemptive priority queues	87
7.3	Kleinrock's conservation law	89
7.4	Service time dependent priorities	90
7.5	Exercises	91
7.6	Further reading	92
7.7	Bibliography	92
8	Waiting time distributions of queues	94
8.1	Waiting time distribution	94
8.2	FCFS waiting time	94
8.3	Busy period analysis	95
8.4	Waiting time distributions	98
8.5	Waiting time for LCFS discipline	101
8.6	Exercises	102
8.7	Further reading	103
8.8	Bibliography	103
9	Multiple server queues	104
9.1	M/M/c queues	104
9.2	M/M/∞ queueing system	106
9.3	Slow servers	107
9.4	System with delayed second server	111
9.4.1	Numerical results	115
9.5	Multiprocessor systems with priorities	117
9.6	Exercises	122
9.7	Further reading	122
9.8	Bibliography	123
10	Networks of queues	124
10.1	Tandem queues	124
10.2	Queues with feedback	128
10.3	Jackson networks	129
10.4	Closed queueing networks	130
10.5	More general networks	132
10.6	Coxian distributions	132
10.7	Service disciplines	133
10.7.1	Processor sharing discipline	135
10.7.2	Server per job discipline	136
10.8	Local balance	136
10.9	BCMP theorem	137
10.9.1	Extensions to BCMP theorem	140
10.10	Exercises	141
10.11	Further reading	141

10.12	Bibliography	141
11	Computational algorithms for product form queueing networks	143
11.1	Convolution algorithm	143
11.2	Mean value analysis	150
11.3	LBANC	152
11.4	Multiple class networks 154	
11.4.1	Convolution algorithm	154
11.4.2	MVA	155
11.4.3	LBANC	156
11.5	Dynamic scaling techniques	156
11.6	Tree structured convolution	158
11.7	Augmented MV A	160
11.8	RECAL	162
11.9	Mixed networks	166
11.10	Further reading	169
11.11	Bibliography	171
12	Approximations and bounds	173
12.1	Approximate MV A and Linearizer	173
12.2	Proportional approximation method	175
12.3	Priority approximations	176
12.3.1	Reduced availability approximations	177
12.3.2	Delay modification approximations	178
12.4	Flow equivalent aggregation	180
12.5	Asymptotic bound analysis	182
12.6	Balanced job bounds	183
12.7	Performance bound hierarchies	185
12.7.1	Optimistic hierarchy	186
12.7.2	Pessimistic hierarchy	186
12.8	Further reading	186
12.9	Bibliography	187
13	Numerical solution of queueing models	190
13.1	Homogenous equations	191
13.1.1	Wachter's algorithm	192
13.1.2	Plemmon's algorithm	193
13.1.3	Grassmann's algorithm	193
13.1.4	Iterative techniques	195
13.2	Eigenvector solutions	196
13.2.1	Power method	197
13.2.2	Simultaneous iteration	198
13.3	Decomposition methods	200
13.4	Matrix-geometric methods	204
13.5	Exercises	209
13.6	Further reading	209
13.7	Bibliography	210
14	Local area networks	212
14.1	Broadcast networks	213
14.1.1	ALOHA protocols	213
14.1.2	Carrier sense multiple access protocols	218
14.1.3	Carrier sense multiple access with collision detection	221
14.2	Ring networks	222
14.2.1	Token rings	224
14.2.2	Slotted rings	228
14.2.3	Register insertion rings	231
14.3	Exercises	234
14.4	Further reading	234
14.5	Bibliography	235
Index		238

Computer systems design is full of conundrums: Given a choice between a single machine with speed s , or n machines each with speed s/n , which should we choose? If both the arrival rate and service rate double, will the mean response time stay the same? Should systems really aim to balance load, or is this a convenient myth? If a scheduling policy favors one set of jobs, does it necessarily hurt some other jobs, or are these "conservation laws" being misinterpreted? Do greedy, shortest-delay, routing strategies make sense in a server farm, or is what's good for the individual dis

As complexity of computer and communication systems increases, it becomes hard to analyze the system via analytic models. Measurement based system evaluation may be too expensive. In this tutorial, discrete event simulation as a model based technique is introduced. This is widely used for the performance/availability assessment of complex stochastic systems. Importance of applying a systematic methodology for building correct, problem dependent, and credible simulation models is discussed. These will be made evident by relevant experiments for different real-life problems and interpreting their System models for distributed systems.

INF5040/9040 autumn 2011. lecturer: Frank Eliassen.

- Physical models: capture the hardware composition of a system in terms of computers and other devices and their interconnecting network;
- Architecture models: define the main components of the system, what their roles are and how they interact (software architecture), and how they are deployed in a underlying network of computers (system architecture)

Clock Performance Performance. Process Process Channel. Process's local clock exceeds the bounds on its rate of drift from real time Process exceeds the bounds on the interval between two processing steps A message's transmission takes longer than the stated bounds.

Keywords: Computer networks, performance, modeling, simulation, verification.

1. Introduction.

Computer networks are an inherent substrate in many daily tasks such as business, e-commerce

- Modeling a system involves the abstraction of its features and properties, focusing exclusively on those that are of interest to the study [Garzia et. al., 1990]. As a result, a model can be understood as the
- voice communication services on the market. Voice traffic models should consider components both at user and packet levels.

Analytical Modeling for Computer Systems But it does not have to be this way! These same systems designers could mathematically model the system, stochastically characterize the workloads and performance goals, and then analytically derive the performance of the system as a function of workload and input parameters. The fields of analytical modeling and stochastic processes have existed for close to a century, and they can be used to save systems designers huge numbers of hours in trial and error while improving performance. Analytical modeling can also be used in conjunction with simulation t