Delay and Throughput Analysis of the Stack Algorithm in Mobile Radio Channels with Rayleigh Fading, Shadowing and Near-Far Effect

R.D. Vossenaar

Type: Graduation Thesis

Project carried out at: Telecommunications and Traffic-Control Systems Group
Faculty of Electrical Engineering
for: Physics Informatics Group
Faculty of Applied Physics
Delft University of Technology

Date: November 1991

Chairman of the Telecommunications and Traffic-Control Systems Group:
Prof. Dr. J.C. Arnbak
Mentor at the Telecommunications and Traffic-Control Systems Group:
Prof. Dr. R. Prasad
Mentor at the Physics Informatics Group:
Prof. Dr. I.T. Young

ABSTRACT

The performance of the stack algorithm in mobile radio channels with Rayleigh fading, shadowing and near-far effect, has been analyzed. Both coherent and incoherent Rayleigh fading were considered. Shadowing and near-far effect were modelled by assuming a log normal distribution for their local mean power and area mean power respectively. Three versions of the stack algorithm were taken into consideration, each with a different ability to distinguish between channel events in a previous time slot. The stack algorithm was described as a regenerative process. Throughput and delay characteristics have been determined up to a critical generation rate of traffic, where the mean packet delay and mean basic session length (regeneration cycle) grow to infinity. It was shown that capture models based on combined effects yield higher throughputs. Coherent Rayleigh fading offered higher throughputs than incoherent Rayleigh fading. Comparisons were made with the ALOHA algorithm. The results were slightly in favour of the ALOHA algorithm. However, near the critical generation rate of one version of the stack algorithm, higher throughputs were found than the ALOHA algorithm offers.

Acknowledgements

I would like to thank Prof. Dr. R. Prasad for his scientific guidance and Dr. N.D. Vvedenskaja of the Academy of Sciences, for giving me better insight into the algorithm, during her stay.

TABLE OF CONTENTS

Abstract ii

List of Symbols v

1 Introduction 1

2 The Stack Algorithm with Capture 3
2.1 Introduction 3
2.2 Notations and Definitions 3
2.3 Stack Algorithm Instructions 4
2.3.1 Stack algorithm with "idle/success/failure" feedback 5
2.3.2 Stack algorithm with "conflict/no conflict" feedback 5
2.3.3 Stack algorithm with "idle/transmission" feedback 6
2.4 Stack States and Sessions 6
2.4.1 Introduction 6
2.4.2 Stack states 7
2.4.3 Partitioning of sessions 10
2.5 Session Length, Delay, Throughput and Offered Traffic 13
2.6 Stability of the Stack Algorithm 21
2.7 Solving the Equations 22

3 Capture 25
3.1 Introduction 25
3.2 Defining the Capture Probability 25
3.3 Rayleigh Fading Only 30
3.3.1 Incoherent interference cumulation 30
3.3.2 Coherent interference cumulation 31

3.4 Shadowing Only 31
3.5 Near-Far Effect Only 33
3.6 Combined Rayleigh Fading and Shadowing 34
3.6.1 Incoherent interference cumulation 34
3.6.2 Coherent interference cumulation 36
3.7 Combined Rayleigh Fading and Near-Far Effect 38
3.7.1 Incoherent interference cumulation 38
3.7.2 Coherent interference cumulation 39
3.8 Combined Shadowing and Near-Far Effect 40
3.9 Combined Rayleigh Fading, Shadowing and Near-Far Effect 41
3.9.1 Incoherent interference cumulation 42
3.9.2 Coherent interference cumulation 42
3.10 Conclusions 43

4 Computational Results 45

5 Conclusions and Recommendations 53

References 55

Appendix A Spatial Distribution of Traffic 58

Appendix B Additional Results 61

LIST OF SYMBOLS

D	mean packet delay [slots per packet]
d	distance between transmitting mobile unit and base station
d_k	conditional mean of the total delay of all packets, transmitted during a session, that started with an initial conflict of k packets [slots]
d_{max}	radius of the circular area with centre at the base station, in which mobile units are expected to move around
$f(x)$	probability density function of a random variable x
G	mean number of packets transmitted in a time slot (newly generated and retransmitted) [packets per slot]
G_t	total traffic offered to the channel [packets per slot]
$G(\rho)$	spatial distribution of traffic at normalized distance ρ [packets per slot]
H	mean length of a basic session [slots]
H_k	conditional mean of the length of a session, that started with an initial conflict of k packets [slots]
k	conflict multiplicity (number of packets transmitted in a time slot) [packets]
L_t	virtual stack level of a packet, at time t (the beginning of slot [t,t+1))
m_n	logarithmic mean of the joint area mean interference power
P_a	power of the one packet to be analyzed (testpacket)
P_i	power of an interfering packet
P_{li}	probability that i out of l packets are moved to level one of the virtual stack and l-i to level zero (binomial distribution)
P_n	joint interference power
p_a	instantaneous signal power of one packet
p_n	joint instantaneous interference power

p_j	probability that j newly generated packets are transmitted in a time slot (Poisson distribution, also indexed k,m)
p_{oa}	local mean power of one packet
p_{on}	joint local mean interference power
p_s	probability that a packet is moved to virtual stack level one after a conflict
Q_k	probability that any of k packets, transmitted in a time slot, captures the receiver
R	remainder when using Hermite integration
r_a	(instantaneous) amplitude of a received signal
S	throughput [packets per slot]
t	start of a time slot [t,t+1)
t_k	probability that k packets (new packets and retransmissions) are transmitted in a time slot
t_w	section of a time slot in which a packet may capture the receiver (capture window)
v_i	i^{th} weight factor of Hermite integration
v_{km}	mean number of multiplicity k conflicts in a session that started with an initial conflict of m packets
w_i	i^{th} sample point of Hermite integration
w_t	number of packets transmitted in slot [t,t+1) [packets]
z_o	capture ratio
$\alpha_{i,t}$	number of old packets waiting for retransmission at level i of the virtual stack at time t (start of slot [t,t+1)) [packets]
β_t	number of newly generated packets ready for transmission at time t (start of slot [t,t+1)) [packets]
γ	path loss exponent

δ_{km}	Kronecker delta (equals one if k equals m, zero otherwise)
δ_t	total delay of all packets transmitted during a session that started at time t (start of slot [t,t+1)) [slots]
ζ_a	area mean power of one packet
ζ_n	joint area mean interference power
Θ_t	channel event in slot [t-1,t) (idle slot, single packet transmission, conflict with/without capture)
λ	mean generation rate of packets [packets per slot]
λ_{cr}	critical generation rate, up to which mean session lengths and delays are finite and the stack algorithm is stable [packets per slot]
μ_t	virtual stack marker at time t (start of slot [t,t+1))
ρ	distance between transmitting mobile unit and base station normalized at the cell radius (d_{max})
σ_{da}^2	logarithmic variance of the area mean power of one packet (σ_{da} is referred to as spatial spread)
σ_{dn}^2	logarithmic variance of the joint area mean interference power
σ_n^2	logarithmic variance of the joint instantaneous interference power
σ_{oa}^2	logarithmic variance of the local mean power of one packet (σ_{oa} is referred to as shadowing spread)
σ_{on}^2	logarithmic variance of the joint local mean interference power
σ_t	virtual stack state at time t (start of slot [t,t+1))
τ_t	length of a session that started at time t (start of slot [t,t+1) [slots]

(The (joint or single packet) power levels mentioned above, are the power levels at the base station during the capture time)

1 INTRODUCTION

Efficiently sharing a common transmission channel among a number of users is a key design issue for communication networks. Systems, in which users share a common resource, are called multiple access systems. When there is a potentially large amount of users with bursty traffic, random multiple access (RMA) algorithms, become more efficient than fixed channel schemes, in which, for instance, each user has exclusive possession of a frequency band or the exclusive right to access the channel during a certain time period.

A well known RMA algorithm is the ALOHA algorithm. This algorithm allows each user to transmit whenever it has data to be sent. If a collision occurs, each station, whose packet is destroyed, waits a random amount of time before retransmitting its packet. In order to avoid partial overlap of packets (which may result in a time loss of up to twice the packet length), packets are transmitted in predefined time slots. The packet length is taken as the duration of a time slot. The use of time slots has shown to increase the throughput (the average number of successfully transmitted packets per time slot, which is of course less than one) [8].

The stack algorithm is another RMA algorithm. The difference with ALOHA is that packets, destroyed in a collision, are divided among two levels of a virtual stack. Packets at one level will be retransmitted immediately. Packets at the other will have to wait until those packets (plus any packets generated intermediately), are successfully transmitted [1],[2].

In most cable networks a collision is assumed to destroy all packets involved.

In packet radio networks however, this will not necessarily be the case.

Due to the fading and pathloss characteristics of a real electromagnetic environment, packets arrive at the receiver with different power levels [11]. The strongest of the colliding packets may capture a discriminating receiver. This property is called the capture effect.

In this project the capture effect was introduced into the stack algorithm in order to analyze its performance in a mobile radio network. The performance of the stack algorithm was measured by analyzing its packet delay (the number of time slots elapsed between packet generation and successful transmission) and throughput characteristics. Throughputs were compared with those of the ALOHA algorithm. The project was carried out at the Telecommunications and Traffic-Control Systems Group of the Faculty of Electrical Engineering as graduate work for the Physics Informatics Group of the Faculty of Applied Physics at the Delft University of Technology.

2 THE STACK ALGORITHM WITH CAPTURE

2.1 Introduction

During the last decade investigations have been carried out to analyze the performance of the stack algorithm [1],[2]. In those investigations, it has been assumed that packets transmitted simultaneously are all destroyed and need to be retransmitted.

In a real electromagnetic environment of mobile and indoor radio channels however, the strongest packet may be able to capture a discriminating receiver. This property, called capture effect, is introduced into the stack algorithm in this chapter.

2.2 Notations and Definitions

The number of stations in the communication system is considered to be infinite. Each station has at most one packet ready for transmission. The packets are of fixed lengths.

The stack algorithm uses a slotted scheme, in which the packet length is taken as the time unit. The time interval [t,t+1) will denote a slot ($t \in \{0,1,2,...\}$).

Packet transmissions can only begin at the beginning of a time slot. If only one packet is transmitted in a slot, the packet is considered to be transmitted successfully. If a number of stations (k, $k \geq 2$) transmit their packet in the same time slot, the packets will interfere with one another. This will be called a conflict of multiplicity k.

Under some circumstances (discussed in chapter 3) one of these packets may capture the receiver (successful transmission).

All unsuccessfully transmitted packets have to be retransmitted later on.

Roughly there are four possible events e_t in a time slot [t-1,t):

- Idle slot (no transmissions)
- Single packet transmission
- Conflict with capture (one of k packets was transmitted successfully)
- Conflict without capture (none of k packets were transmitted successfully).

The kind of feedback determines to what extend a distinction is made between these events (see next paragraph). Feedback is considered to be errorless.
At time t all stations will know the status of slot [t-1,t).
The stack algorithm makes use of a virtual stack:
If a station has a packet to transmit, it assigns it an integer packet level L_t, valid during slot [t,t+1), as if the packet was residing on a stack at that level. Level assignment is done at the beginning of each time slot, from the moment of packet generation until its successful transmission, and is based on feedback information about the previous time slot.
A packet generated during the interval [t-1,t) (that is ready for transmission in slot [t,t+1)) is said to be new at time t, all previously generated packets are said to be old at time t (waiting for retransmission).

2.3 Stack Algorithm Instructions

In this paragraph the instructions of three versions of the stack algorithm will be discussed, each with a different kind of feedback.
All algorithms are nonblocked; newly generated packets are transmitted at the beginning of the next slot, regardless of what happened in previous slots.
Each station transmitting a packet will know at the beginning of the next slot, whether it was successful or not.

2.3.1 Stack algorithm with "idle/success/failure" feedback

This algorithm (further referred to as A1) uses ternary feedback. A distinction is made between an idle slot, successful transmission (that is single packet transmission or conflict with capture) and failure (conflict without capture).

Instructions:

1) A packet is transmitted if and only if $L_t=0$.
2) Every newly generated packet is assigned the level $L_t=0$ and transmitted in slot $[t,t+1)$.
3) If $L_{t-1}=0$ (assigned at the beginning of the previous slot) and this packet was transmitted successfully it leaves the system (stack).
4) If $L_{t-1}=0$ and a successful transmission occurred in the previous time slot, but this packet was unsuccessfully transmitted, then $L_t=0$ (immediate retransmission).
5) If $L_{t-1}=0$ and failure occurred then with probability p_s the packet is moved to the stack at level $L_t=1$ and with probability $1-p_s$ it stays at level $L_t=0$ (immediate retransmission).
6) If $L_{t-1}>0$ and failure occurred then $L_t=L_{t-1}+1$.
7) If $L_{t-1}>0$ and a successful transmission occurred then $L_t=L_{t-1}$.
8) If $L_{t-1}>0$ and an idle slot occurred then $L_t=L_{t-1}-1$.

2.3.2 Stack algorithm with "conflict/no conflict" feedback

This stack algorithm (further referred to as A2) uses binary feedback. A distinction is made between conflict (with or without capture) and no conflict (idle slot or single packet transmission).

Instructions:

Instructions 1, 2 and 3 coincide with the corresponding instructions of algorithm A1.

4) If $L_{t-1}=0$ and conflict occurred, without this packet capturing the receiver, then with probability p_s it is moved to the stack at level $L_t=1$ and with probability $1-p_s$ it stays at level $L_t=0$ (immediate retransmission).

5) If $L_{t-1}>0$ and conflict occurred then $L_t=L_{t-1}+1$.

6) If $L_{t-1}>0$ and no conflict occurred then $L_t=L_{t-1}-1$.

2.3.3 Stack algorithm with "idle/transmission" feedback

This algorithm (further referred to as A3) uses binary feedback. A distinction is made between an idle slot and transmission of one or more packets.

Instructions:

The instructions 1, 2 and 3 coincide with the corresponding instructions of algorithm A1.

4) If $L_{t-1}=0$ (transmission), without this packet capturing the receiver, then with probability p_s it is moved to the stack at level $L_t=1$ and with probability $1-p_s$ it stays at level $L_t=0$ (immediate retransmission).

5) If $L_{t-1}>0$ and one or more packets were transmitted then $L_t=L_{t-1}+1$.

6) If $L_{t-1}>0$ and an idle slot occurred then $L_t=L_{t-1}-1$.

2.4 Stack States and Sessions

2.4.1 Introduction

In the previous paragraph several versions of the nonblocked stack algorithm were described in terms of how they work for each station. This section contains a description of how the virtual stack varies from slot to slot. It also describes an important feature of the stack algorithm: partitioning of sessions.

2.4.2 Stack states

Let β_t denote the number of packets new at time t and let $\alpha_{i,t}$ denote the total number of old packets in cell i (that is level i) of the virtual stack at time t. Then the conflict of multiplicity w_t, in slot [t,t+1) is given by:

$$w_t = \beta_t + \alpha_{0,t} \tag{2.1}$$

Further let μ_t-1 be the highest level of the virtual stack at time t, that is, there are no packets at level μ_t or higher. μ_t Is called the stack marker at time t. The stack is empty if and only if μ_t=0. Therefore a one level stack with zero packets at level zero is considered not to be empty!

In a blocked stack algorithm [1] each station actually keeps track of this marker because new packets are not allowed on the channel until the stack is empty. In the discussion of nonblocked versions, its just a notation, useful in describing the stack state σ_t. The following holds for the stack state at time t:

$$\sigma_t = \{ \alpha_{0,t}, \alpha_{1,t}, \ldots, \alpha_{\mu_t-1,t} \} \tag{2.2a}$$

or, for an empty stack:

$$\sigma_t = \varnothing \tag{2.2b}$$

The sequence of stack states is a homogeneous first order Markov chain [1]; the stack state at time t depends on the stack state at time t-1 and the transition probabilities from σ_{t-1} to σ_t. These transition probabilities are determined by the probability distribution of the newly generated packets β_{t-1} and by how the algorithm responds to the conflict of multiplicity w_{t-1} in slot [t-1,t) (see figure 2.1).

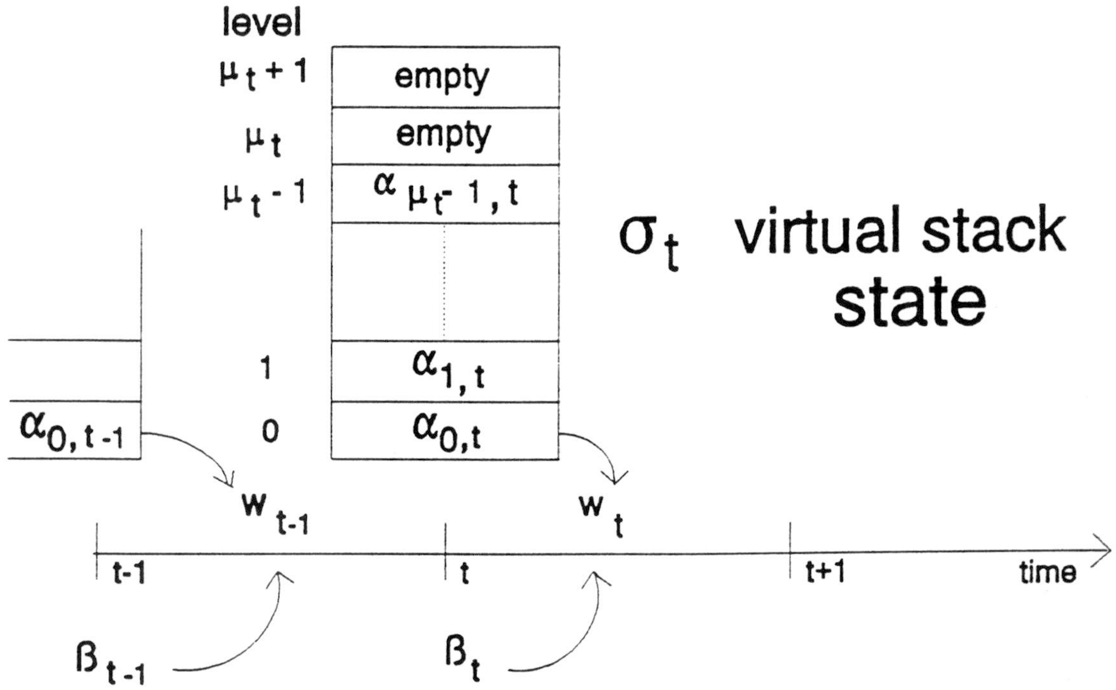

Figure 2.1 Virtual stack state at time t. $\alpha_{i,t}$ Denotes the number of old packets in cell i of the virtual stack, β_t the number of new packets, w_t the conflict multiplicity and μ_t the stack marker at time t (the beginning of time slot [t,t+1)).

The transition between stack states will shortly be described here for each algorithm, given an event in the previous time slot.

Algorithm A1:

- Idle slot ($w_{t-1}=0$):

  If the stack was empty it stays empty, if it was not, all packets move one level down ($\alpha_{i,t}=\alpha_{i+1,t-1}$, for all i: $0\leq i\leq \mu_{t-1}-2$) and the marker decreases by one.

- Successful transmission ($w_{t-1}\geq 1$ and one successful transmission):

  If the stack was empty the marker increases by one. If the stack was not empty all packets stay at the same level and the marker stays unchanged. The lowest level will contain $\alpha_{0,t}= w_{t-1}-1$ (even if $w_{t-1}=1$).

- Failure ($w_{t-1}\geq 2$ and no capture):

  All packets at level one and higher (if there) move one level up. Each of the w_{t-1} packets move to level one with probability p_s (summing up to $\alpha_{1,t}$) and stay at level zero with probability $1-p_s$ (summing up to $\alpha_{0,t}$ and $\alpha_{0,t}+\alpha_{1,t}=w_{t-1}$).

  If the stack was empty the marker increases by two, otherwise by one.

Algorithm A2:

- Conflict ($w_{t-1}\geq 2$):

  The same transition takes place as when failure occurs using A1, except for the fact that if one of the packets was transmitted successfully (capture), $w_{t-1}-1$ packets are distributed over levels zero and one (if $w_{t-1}-1=1$, the packet will be placed at level zero or one).

- no conflict ($w_{t-1}\leq 1$):

  The same transition takes place as when an idle slot occurs using A1.

Algorithm A3:

- Idle slot ($w_{t-1}=0$):

  An identical transition takes place as when an idle slot occurs using A1.

- Transmission ($w_{t-1}\geq 1$):

  The same applies as when conflict occurs using A2 (if $w_{t-1}=1$ the lowest two levels will both contain zero packets).

2.4.3 Partitioning of sessions

A session starting at t (the beginning of slot [t,t+1)), consists of those slots needed to process the packets at level zero of the virtual stack at time t, plus any new packets that may arrive during this session. The packets are considered to be processed when they are transmitted successfully and the packets at level one of the virtual stack at time t, are now ready to be processed (moved to level zero). (In case of a one level stack at time t the session ends when the stack becomes empty).
If the stack is empty at time t the session starts with a slot in which only new packets are transmitted and ends when the stack becomes empty again (all packets arriving intermediately are processed). Such a session is called a basic session.
The above can also be described in terms of the stack marker.
If μ_t is not equal to zero (stack not empty) then a session starting at time t consists of all slots from t to a time $t+\tau_t$, where $\mu_{t+\tau_t}$ equals μ_t-1 for the first time (see figure 2.2.a on the next page). If $\mu_t=0$ (empty stack) then the basic session starting at time t consists of all slots from t to a time $t+\tau_t$, where $\mu_{t+\tau_t}$ becomes zero for the first time since t (see figure 2.2.b on the next page). τ_t is the length of the session starting at time t ($\tau_t\geq 1$). It should now be clear that a session ends after an idle slot or a single packet transmission, depending on the algorithm.

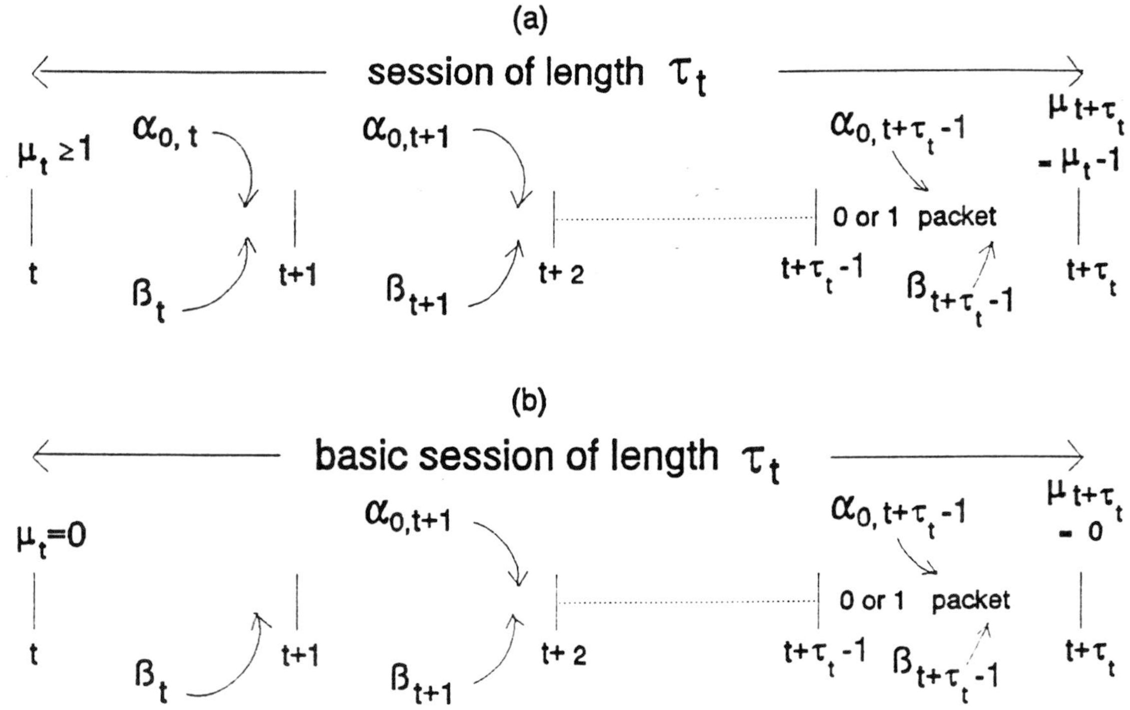

Figure 2.2 Sessions, (a) (non basic) session and (b) basic session. $\mu_{.}$ Is the stack marker , $\alpha_{0,.}$ the number of old packets at level zero of the virtual stack and $\beta_{.}$ the number of newly arriving packets for each time slot. τ_t is the session length. In the last slot of the session, 0 or 1 packets were transmitted (depending on the algorithm).

The first slot of a session is called the initial slot, its conflict multiplicity is called the session multiplicity. If a session consists of more than one slot, the slot right after the initial slot will be the initial slot of a new session, within this session. This is called partitioning of sessions [1],[2] (see figure 2.3 on the next page) and will be shortly described here for each algorithm, based on feedback information about the initial slot of a session. The reader is supposed to be familiar with the description of stack states given in paragraph 2.4.2.

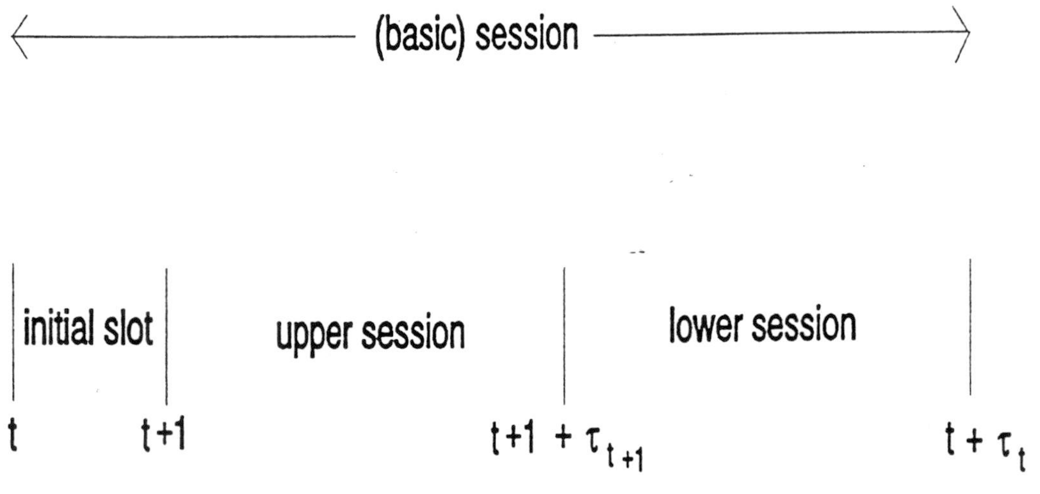

Figure 2.3 Partitioning of a session of length τ_t starting at t.

The upper session starts at t+1 and is of length τ_{t+1}, the lower session starts at $t+1+\tau_{t+1}$ and is of length $\tau_{t+1+\tau_{t+1}}$.

Algorithm A1:

- Idle slot: Session consists of only the initial slot.
- Successful transmission: Session consists of an initial slot and a new session that starts right after the initial slot.

  Since this new session is dependent of the initial slot it is called the upper session of the session and when it ends the whole session ends. Summarizing, the session consists of an initial slot and an upper session.
- Failure: The unsuccessful packets of the initial slot are divided over level zero and one of the virtual stack. The packets at level zero (plus any new packets) are transmitted in the next slot, thus starting a new session. When this session ends the packets placed at level one after the initial slot (plus any new packets), are transmitted. Again a new session is started. When this new session ends the whole

session ends. Both sessions are dependent of the initial slot and are therefore called the upper and lower session (respectively) of the session.
Summarizing, the session consists of an initial slot, an upper session and a lower session.

Algorithm A2:
- Conflict: Same partitioning as algorithm A1, failure.
- No conflict: Session consists of only the initial slot.

Algorithm A3:
- Idle slot: Session consists of only the initial slot.
- Transmission: Same partitioning as algorithm A1, failure.

The upper and lower sessions, as mentioned above, are sessions themselves and are partitioned in accordance to the same rules.
Basic sessions, although partitioned as any other session, are never part of another session; they do not overlap. If a basic session with length τ_t starts at time t the next basic session will start at time $t+\tau_t$. Basic sessions are independently and identically distributed random variables, as are their lengths [1],[2].

2.5 Session Length, Delay, Throughput and Offered Traffic

Let H_k denote the conditional mean of the length of a session, that started at t, given a session multiplicity $w_t=k$. Then H_k is given by:

$$H_k = E\{ \tau_t \mid w_t = k \} \tag{2.3}$$

H_k is independent of t.

In case of a basic session, w_t is only inflicted by newly generated packets.
Let p_k be the probability distribution of the number of newly generated packets (k), then the (unconditional) mean length of a basic session, H, is given by:

$$H = \sum_{k=0}^{\infty} p_k H_k \tag{2.4}$$

since basic sessions do not overlap and are identically and independently distributed variables. The number of newly generated packets is assumed to be Poisson distributed [6],[7], with mean generation rate λ packets per time slot:

$$p_k = \frac{\lambda^k}{k!} \exp(-\lambda) \tag{2.5}$$

Let δ_t denote the total delay of all packets transmitted during a session that started at t. (The delay of one packet is the number of elapsed slots between the moment of generation until successful transmission).
Let d_k denote the conditional mean of δ_t, given a session multiplicity w_t=k. Then d_k is given by:

$$d_k = E\{\delta_t \mid w_t = k\} \tag{2.6}$$

The mean total delay in a basic session can be calculated similarly to what was done for the length of a basic session (equation 2.4); by weighting it with p_k.

Dividing this by the mean number of packets arriving during a basic session (λH), yields the mean packet delay D [1]-[3],[5]:

$$D = \frac{\sum_{k=0}^{\infty} p_k d_k}{\lambda H} \tag{2.7}$$

Successful transmission can happen in one of two ways:

- Only one packet is transmitted in a time slot.
- More then one packet is transmitted and one of those packets captures the receiver.

Let Q_k be the probability that any one of k arriving packets captures the receiver and t_k be the probability that k packets are transmitted in a slot (new packets plus retransmissions), then the throughput (S,S≤1) [6],[7],[8], can be given by:

$$S = t_1 + \sum_{k=2}^{\infty} t_k Q_k \tag{2.8}$$

Q_k will be given in the next chapter, where capture is discussed.
The mean number of packets transmitted per time slot, G, is given by:

$$G = \sum_{k=1}^{\infty} k t_k \tag{2.9}$$

Let V_{km} denote the mean number of conflicts of multiplicity k in a session of multiplicity m (m packets in initial slot). Then the mean number of conflicts of multiplicity k in a basic session can be calculated similarly to what was done for the length of a basic session (equation 2.4); by weighting it with p_m. Dividing this by the mean length of a basic session [1],[5], yields t_k:

$$t_k = \frac{\sum_{m=0}^{\infty} p_m V_{km}}{H} \tag{2.10}$$

Finding expressions for H, D, S, and G is now reduced to finding expressions for H_k, d_k and V_{km} (for each algorithm).

For this purpose, an additional expression is needed.

Let there be l packets to be divided over virtual stack levels zero and one and let p_s be the probability that a packet is moved to level one. Then the probability that i packets are moved to level one (and l-i packets stay at level zero) is given by:

$$P_{li} = \binom{l}{i} p_s^i (1-p_s)^{l-i} \tag{2.11}$$

The desired expressions for H_k, d_k en V_{km} will be derived for each algorithm [1]-[4], by writing them in terms of an initial slot, an upper session and a lower session in accordance with the rules described in paragraph 2.4 (partitioning of sessions and stack states).

In what follows j will denote the number of newly generated packets (with Poisson distribution p_j, given by equation 2.5). In the expressions of V_{km}, δ_{km} is the Kronecker delta and is given by:

$$\delta_{km} = \begin{cases} 1, & k=m \\ 0, & k \neq m \end{cases} \tag{2.12}$$

Algorithm A1:

$$H_0 = 1 \tag{2.13a}$$

$$H_1 = 1 + \sum_{j=0}^{\infty} p_j \, H_j \tag{2.13b}$$

$$H_k = 1 + (1-Q_k) \sum_{i=0}^{k} \sum_{j=0}^{\infty} P_{ki} \, p_j \, (H_{k-i+j} + H_{i+j})$$

$$+ \, Q_k \sum_{j=0}^{\infty} p_j \, H_{k-1+j} \qquad , \; k \geq 2 \tag{2.13c}$$

$$d_0 = 0 \tag{2.14a}$$

$$d_1 = \sum_{j=0}^{\infty} p_j d_j \tag{2.14b}$$

$$d_k = k - Q_k + (1 - Q_k) \sum_{i=0}^{k} \sum_{j=0}^{\infty} P_{ki} p_j \, (d_{k-i+j} + d_{i+j} + iH_{k-i+j})$$

$$+ \, Q_k \sum_{j=0}^{\infty} p_j d_{k-1+j} \qquad , \; k \geq 2 \tag{2.14c}$$

$$V_{k0} = \delta_{k0} \tag{2.15a}$$

$$V_{k1} = \delta_{k1} + \sum_{j=0}^{\infty} p_j V_{kj} \tag{2.15b}$$

$$V_{km} = \delta_{km} + (1-Q_m)\sum_{i=0}^{m}\sum_{j=0}^{\infty} P_{mi} p_j (V_{k,m-i+j} + V_{k,i+j})$$

$$+ Q_m \sum_{j=0}^{\infty} p_j V_{k,m-1+j} \quad , m \geq 2 \tag{2.15c}$$

Equations 2.13a, 2.14a and 2.15a are "idle slot" situations; the session consists of only an initial slot.

Equations 2.13b, 2.14b and 2.15b are " successful transmission" situations; the session consists of an initial slot and an upper session. The upper session multiplicity is only inflicted by new packets.

Equations 2.13c, 2.14c and 2.15c are partly "successful transmission" situations and partly "failure" situations. In the first case the session consists of an initial slot and an upper session ($Q_{..}$ Σ...). The upper session multiplicity is inflicted by new packets (j) and the old packets from the initial slot minus one (the successful packet).

In the latter case (failure) the session consists of an initial slot, an upper session and a lower session (both upper and lower session represented in ($1-Q_{..}$) ΣΣ...).

The upper and lower session multiplicities are inflicted by new (j) and old packets. The old packets moved to level one of the virtual stack (i), are retransmitted in the initial slot of the lower session, the rest of the old packets (level zero) are retransmitted in the initial slot of the upper session.

In equation 2.14c it should be noted that those packets moved to level one of the stack, all have to wait H_{k-i+j} slots before they are retransmitted (H_{k-i+j} is the mean length of the upper session).

Further, it should be noted that the contribution of the initial slot to the mean session length equals one. Its contribution to the mean total delay equals zero if $k \leq 1$ and $k-Q_k$ if $k \geq 2$ (in case of capture k-1 packets have to wait one slot).

Finally, in the expression for V_{km}, the initial slot is only counted if its multiplicity equals the multiplicity which is meant to be counted (m=k).

Algorithm A2:

$$H_0 = H_1 = 1 \qquad \textbf{(2.16a)}$$

$$H_k = 1 + (1-Q_k)\sum_{i=0}^{k}\sum_{j=0}^{\infty} P_{ki}p_j(H_{k-i+j} + H_{i+j})$$
$$+ Q_k\sum_{i=0}^{k-1}\sum_{j=0}^{\infty} P_{k-1,i}p_j(H_{k-1-i+j} + H_{i+j}) \qquad , k \geq 2 \qquad \textbf{(2.16b)}$$

$$d_0 = d_1 = 0 \qquad \textbf{(2.17a)}$$

$$d_k = k - Q_k + (1-Q_k)\sum_{i=0}^{k}\sum_{j=0}^{\infty} P_{ki}p_j(d_{k-i+j} + d_{i+j} + iH_{k-i+j})$$
$$+ Q_k\sum_{i=0}^{k-1}\sum_{j=0}^{\infty} P_{k-1,i}p_j(d_{k-1-i+j} + d_{i+j} + iH_{k-1-i+j}) \qquad , k \geq 2 \qquad \textbf{(2.17b)}$$

$$V_{ko} = \delta_{ko} \qquad \textbf{(2.18a)}$$

$$V_{k1} = \delta_{K1} \tag{2.18b}$$

$$V_{km} = \delta_{km} + (1-Q_m)\sum_{i=0}^{m}\sum_{j=0}^{\infty} P_{mi}p_j(V_{k,m-i+j} + V_{k,i+j})$$

$$+ Q_m\sum_{i=0}^{m-1}\sum_{j=0}^{\infty} P_{m-1,i}p_j(V_{k,m-1-i+j} + V_{k,i+j}) \quad , m \geq 2 \tag{2.18c}$$

Equations 2.16a, 2.17a, 2.18a and 2.18b are "no conflict" situations; the session consists of only an initial slot.

Equations 2.16b, 2.17b and 2.18c are "conflict" situations; the session consists of an initial slot, an upper session and a lower session. The only difference with equations 2.13c, 2.14c and 2.15c is that in case of capture, the successful packets are divided over levels zero and one of the virtual stack as well. The contribution of the initial slot to the equations is the same as described for algorithm A1.

Algorithm A3:

Equations 2.13a, 2.14a and 2.15a also hold for algorithm A3, since the "idle slot" situation is equivalent for both algorithms.

The "transmission" situation of algorithm A3 is equivalent to the "conflict" situation of algorithm A2, in terms of stack state transitions and partitioning. Therefore, equations 2.16b, 2.17b and 2.18c also hold for algorithm A3.

In case of a single packet transmission these equations simplify to:

$$H_1 = 1 + 2\sum_{j=0}^{\infty} p_j H_j \tag{2.19}$$

$$d_1 = 2\sum_{j=0}^{\infty} p_j d_j \tag{2.20}$$

$$V_{k1} = \delta_{k1} + 2\sum_{j=0}^{\infty} p_j V_{kj} \tag{2.21}$$

since the initial slot of both the upper and lower session have zero old packets to process.

2.6 Stability of the Stack Algorithm

As discussed in paragraph 2.4, the sequence of stack states is a Markov chain. It was also shown that basic sessions are independently and identically distributed random variables.

If the mean basic session length is finite, the sequence of stack states is renewed at the beginning of each basic session (empty stack). Such a process is called a renewal process, and the start of a basic session is called a point of regeneration [5].

In [1] it is shown that a necessary condition for the stack algorithm to be stable is that the mean basic session length is finite. Then the distribution of stack states (σ_t) goes to a stationary one when t goes to infinity. For given p_s, algorithm and capture model (Q_k, see chapter 3), the mean basic session length and mean packet delay will only be finite up to a certain packet generation rate, called critical genaration rate (λ_{cr}).

2.7 Solving the Equations

The equations 2.13 to 2.21 in paragraph 2.5 are all of the form:

$$x_k = b_k + \sum_{l=m}^{\infty} a_{kl} x_l \quad , k \geq m \tag{2.22}$$

; for which a solution $x = (x_0, x_1, x_2, ...)$ is sought.
a_{kl} Is determined by probabilities P, Q and p, and b_k is determined by all known values in the equations. The values of $a_{kl}x_l$ are summed for all unknown values of x_l.
For algorithm A2, x_0 and x_1 are already known (m=2), for algorithms A1 and A3, x_0 is known (m=1).
In [5] it is shown that a finite solution to equation 2.22 is guaranteed, if a sequence $x^0 = (x_0^{\,0}, x_1^{\,0}, x_2^{\,0}, ...)$ is found which satisfies the following inequalities:

$$x_k^0 \geq b_k + \sum_{l=m}^{\infty} a_{kl} x_l^0 \geq 0 \tag{2.23}$$

The sequence x^0 is an upper bound to the solution x, which is sought. One way of solving the equations is to find a lower bound, y^0, in addition to the upper bound and make the following iteration steps for all k≥m, until upper and lower bound are sufficiently close (first iteration step n=0):

$$x_k^{n+1} = b_k + \sum_{l=m}^{\infty} a_{kl} x_l^n \tag{2.24a}$$

$$y_k^{n+1} = b_k + \sum_{l=m}^{\infty} a_{kl} y_l^n \tag{2.24b}$$

A problem is, that uniqueness is not always guaranteed [5].
Another way of solving the equations is to truncate the system to N values (N should be greater or equal to the number of values which are sought).
Equation 2.22 can be rewritten as:

$$c = (A - I)x \tag{2.25}$$

where $c= (-b_m,-b_{m+1},...,-b_N)^T$, $x= (x_m,x_{m+1},...,x_N)^T$.
Matrix A is determined by a_{kl} (k^{th} row, l^{th} column) and I equals the identity matrix.
Then x can be found by:

$$x = (A-I)^{-1}c \tag{2.26}$$

By examining the shape of matrix A, one may see why truncation is justified. Generally speaking, the elements a_{kl} of matrix A, are largely determined by $P_{k,k-i}$, $P_{k,i}$ and p_j. Since in a_{kl} l equals i+j, i is smaller or equal to k and p_j decreases rapidly with increasing j, A has a nearly triangular shape. In addition, for large k and p_s unequal to one or zero, $P_{k,k-i}$ and $P_{k,i}$ (equation 2.11) will be very small, especially for values of i near k. To satisfy this criterium, the closer p_s gets to zero or one the larger K and thus N should be taken. Hence for large enough N, the N^{th} column of A will contain elements, approximately equal to zero, ensuring accuracy (no values of x are left out, that should be included).

Having calculated x, the next step is to recalculate x by increasing the number of values to which the system is truncated. This is repeated until those values of x ,that are needed, sufficiently stay the same.

{It turned out that for p_s equal to zero or one this method could also be used, although the description above implies that for those values of p_s N should be infinitely large. Another method, to truncate the system to N values and subsequently adjust c in equation 2.25 with extrapolated values of $(x_{N+1},x_{N+2},\ldots)$, showed no significant advantage}.

3 CAPTURE

3.1 Introduction

When packets are transmitted simultaneously in a time slot, the strongest packet may be able to capture a discriminating receiver. This capture effect was used in the equations of paragraph 2.5 in terms of Q_k; the probability that one of k packets captures the receiver.
In this chapter Q_k will be exemplified.
The effects of receiver noise and choice of modulation are not taken into consideration.

3.2 Defining the Capture Probability

One of k packets is considered to be transmitted successfully, only if its power (P_a) exceeds the joint interference power of all other packets (P_n) by at least z_0, during a certain section (t_w) of a time slot. (The one packet to be analyzed is often referred to as the testpacket in literature and t_w is called the capture window). Thus, the probability of capture, given k arriving packets, [9],[10] may be expressed as:

$$Q_k = k \, Prob\{\frac{P_a}{P_n} > z_0\} \tag{3.1}$$

, where z_0 is called the capture ratio and

$$P_n = \sum_{i=1}^{k-1} P_i \qquad (3.2)$$

and

$$Prob\{\frac{P_a}{P_n} > z_0\} = \int_0^{\infty} f_n(P_n)\left[\int_{P_n z_0}^{\infty} f_a(P_a)\, dP_a\right] dP_n \qquad (3.3)$$

P_i Is the power of the i^{th} interferer and $f_a(P_a)$ and $f_n(P_n)$ are the probability density functions (p.d.f.'s) of P_a and P_n respectively.
The real electromagnetic environment of mobile radio communications introduces the capture effect through its fading and path-loss characteristics [11].

(Fast) multipath fading (a.k.a. short term fading) is mainly caused by multipath reflections of a transmitted wave by houses, buildings and other man-made structures. These reflected waves arrive at the base station with different phases and at different angles, causing the received signal envelope to fluctuate rapidly around its local mean [11].
Short term fading is modelled by assuming the received signal amplitude (r_a), conditional to its local mean power (p_{oa}), to be Rayleigh distributed [9]-[12]:

$$f(r_a|p_{oa}) = \frac{r_a}{p_{oa}} \exp(\frac{-r_a^2}{2p_{oa}}) \qquad (3.4)$$

Using the following expression for the instantaneous signal power (p_a):

$$p_a = \frac{1}{2} r_a^2 \tag{3.5}$$

yields

$$f(p_a | p_{oa}) = \frac{1}{p_{oa}} \exp(-\frac{p_a}{p_{oa}}) \tag{3.6}$$

, since

$$f(r_a | p_{oa}) \left| \frac{dr_a}{dp_a} \right| = f(p_a | p_{oa}) \tag{3.7}$$

(Slow) shadowing (a.k.a. long term fading) is mainly caused by terrain configuration (such as hills and mountains) and the man-made environment (such as buildings), between base station and the mobile unit.

Therefore, there is a relatively slow statistical variation in local mean power with respect to the area mean power, which is determined by the distance between base station and the mobile unit.

Shadowing is modelled by assuming a log-normal density function for the local mean power, conditional to its area mean power (ζ_a) [10]-[13]:

$$f(p_{oa} | \zeta_a) = \frac{1}{\sqrt{2\pi}\, \sigma_{oa} p_{oa}} \exp(-\frac{(\ln(p_{oa}) - \ln(\zeta_a))^2}{2\sigma_{oa}^2}) \tag{3.8}$$

, where σ_{oa}^2 is the logarithmic variance of p_{oa} (σ_{oa} is also referred to as shadowing spread).

Assuming similar conditions (such as antenna height and transmitting power [9]) for each mobile unit, propagation path loss is mainly determined by the distance between base station and mobile unit. It causes packets, coming in from different distances to arrive at different power levels. This is called the near-far effect and may be described [9],[11],[14], by expressing the area mean power as:

$$\zeta_a = \rho^{-\gamma} \tag{3.9}$$

, where γ is the path loss exponent, which is considered to be equal to four for UHF propagation in cellular radio, and ρ is the normalized distance between mobile unit and base station:

$$\rho = \frac{d}{d_{max}} \tag{3.10}$$

In equation 3.10 d is the actual distance and d_{max} is the radius of the circular area with centre at the base station, in which the mobile units are expected to move around.

The near-far effect is modelled by a log-normal density function for the area mean power, with zero mean and logarithmic variance σ_{da}^2 (see Appendix A):

$$f(\zeta_a) = \frac{1}{\sqrt{2\pi}\,\zeta_a \sigma_{da}} \exp\left(\frac{-(\ln\zeta_a)^2}{2\sigma_{da}^2}\right) \tag{3.11}$$

The parameter σ_{da} is called the spatial spread.

In fact, the models (p.d.f.'s) described previously, not only hold for the packet which is assumed to be the testpacket, but also hold for each of the interfering packets. Therefore, the variables indexed a, that came about, will be used for any arriving packet.

The models (p.d.f.'s), described for each of the characteristics of a typical mobile environment, may be joined together to determine the combined effects of these characteristics on the capture probability. In the following paragraphs the capture probability will be described for:

- Multipath (or Rayleigh) fading only
- Shadowing only
- Near-far effect only
- Combined Rayleigh fading and shadowing
- Combined Rayleigh fading and near-far effect
- Combined shadowing and near-far effect
- Combined Rayleigh fading, shadowing and near-far effect

The (combined) power p.d.f. for each individual packet may be used to determine the p.d.f. of the joint interference power ($f_n(P_n)$).

In case of a log-normal power p.d.f. for each of the packets, the interference power p.d.f. is approximated by another log-normal p.d.f., with mean and variance determined according to the method of Schwartz and Yeh [18],[19].

When describing the joint interference power in terms of the instantaneous signal power (that is when Rayleigh fading is included), the following two situations may occur [9]:

1. The random phase terms of each of the interfering signals vary sufficiently fast during the capture time t_w, which may be caused by inaccuracies in carrier frequencies and Doppler shifts. In that case the interfering packets cumulate incoherently.

2. There is negligible variation in the random phase terms of the interfering signals during the capture time t_w. In that case the interfering packets cumulate coherently.

The mathematical approaches used in these two cases will be discussed in those following paragraphs, where Rayleigh fading is included.

3.3 Rayleigh Fading Only

In this paragraph, all packets are assumed to have equal local mean power (p_{oa}) but different instantaneous power (p_a). Further, p_n will denote the joint instantaneous power of all interfering packets.

3.3.1 Incoherent interference cumulation

The p.d.f. of the joint interference power (of the (k-1) packets) is obtained by taking the (k-1)-fold convolution of equation 3.6 [9]:

$$f(p_n) = \frac{1}{p_{oa}} \frac{(p_n/p_{oa})^{k-2}}{(k-2)!} \exp(-\frac{p_n}{p_{oa}}) \qquad (3.12)$$

(which is a gamma distribution).

Using equation 3.3 with the following substitutions:

- $f_n(P_n) \rightarrow f(p_n)$
- $f_a(P_a) \rightarrow f(p_a)$ ($= f(p_a | p_{oa})$, equation 3.6)
- $P_a \rightarrow p_a$
- $P_n \rightarrow p_n$

and subsequently using equation 3.1 yields:

$$Q_k = k\left(\frac{1}{z_0+1}\right)^{k-1} \tag{3.13}$$

3.3.2 Coherent interference cumulation

The p.d.f. of the joint interference power is approximated by [9]:

$$f(p_n) = \frac{1}{(k-1)p_{oa}} \exp\left(-\frac{p_n}{(k-1)p_{oa}}\right) \tag{3.14}$$

Using equation 3.3 with substitutions equal to those made in paragraph 3.3.1 and subsequently using equation 3.1 yields:

$$Q_k = \frac{k}{(k-1)z_0+1} \tag{3.15}$$

Coherent cumulation suggests a smaller likelihood of harmful interference than incoherent cumulation, since for the former $f(p_n)$ approaches a maximum for $p_n \to 0$, whereas the latter approaches a minimum.

3.4 Shadowing Only

In this paragraph all packets are assumed to have equal area mean power (ζ_a) but different local mean power (p_{oa}) with equal logarithmic variance (σ_{oa}^2). Further, p_{on}

will denote the joint local mean power of all interfering packets and ζ_n the joint area mean power of all interfering packets.

Each individual (independent) packet may be modelled according to the log-normal p.d.f., given by equation 3.8.

The joint interference power (p_{on}) may be modelled by another log-normal p.d.f., with logarithmic variance σ_{on}^2 and mean $\ln\zeta_n$:

$$f(p_{on}) = \frac{1}{\sqrt{2\pi}\,p_{on}\sigma_{on}} \exp\left(-\frac{(\ln(p_{on})-\ln(\zeta_n))^2}{2\sigma_{on}^2}\right) \tag{3.16}$$

Using the method of Schwartz and Yeh [18],[19], yields the values for $\ln(\zeta_n)$ and σ_{on}, for given σ_{oa} and $\ln(\zeta_a)$.

Using equation 3.3 with the following substitutions:

- $f_n(P_n) \rightarrow f(p_{on})$
- $f_a(P_a) \rightarrow f(p_{oa})$ $(=f(p_{oa}|\zeta_a)$, equation 3.8)
- $P_n \rightarrow p_{on}$
- $P_a \rightarrow p_{oa}$

(of which the result is given in [13]) and subsequently using equation 3.1 yields:

$$Q_k = \frac{k}{\sqrt{2\pi}} \int_{-\infty}^{x} \exp\left(-\frac{u^2}{2}\right)du \tag{3.16a}$$

, where

$$\frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} \exp\left(-\frac{u^2}{2}\right)du \triangleq P(x) \tag{3.16b}$$

, and

$$x = \frac{\ln(\zeta_a)-\ln(\zeta_n)-\ln(z_0)}{\sqrt{\sigma_{oa}^2+\sigma_{on}^2}} \tag{3.16c}$$

Q_k may be approximated by making use of an approximation for P(x), given by [20, paragraph 26.2.19].

3.5 Near-Far Effect Only

In this paragraph all packets are assumed to have different area mean power (ζ_a) with equal spatial spread (σ_{da}). Further, ζ_n will denote the joint area mean power of all interfering packets.
Each individual (independent) packet may be modelled according to the log-normal p.d.f. given by equation 3.11.
The joint interference power (ζ_n) may be modelled by another log-normal p.d.f., with logarithmic variance σ_{dn}^2 and logarithmic mean m_n.

$$f(\zeta_n) = \frac{1}{\sqrt{2\pi}\,\zeta_n\sigma_{dn}}\exp\left(\frac{(\ln(\zeta_n)-m_n)^2}{2\sigma_{dn}^2}\right) \tag{3.17}$$

The values for σ_{dn} and m_n, can be determined using the method of Schwartz and Yeh [18],[19], for given σ_{da} and zero mean.

Using equation 3.3 with the following substitutions:

- $f_n(P_n) \rightarrow f(\zeta_n)$
- $f_a(P_a) \rightarrow f(\zeta_a)$ (equation 3.11)

- $P_n \rightarrow \zeta_n$
- $P_a \rightarrow \zeta_a$

and subsequently using equation 3.1 yields similar equations to those given in 3.16a&b, only with a different integration interval:

$$x = \frac{-m_n - \ln(z_0)}{\sqrt{\sigma_{da}^2 + \sigma_{dn}^2}} \tag{3.18}$$

Q_k is approximated using the same method as used in paragraph 3.4.

3.6 Combined Rayleigh Fading and Shadowing

In this paragraph all packets are assumed to have equal area mean power (ζ_a), but different instantaneous power (p_a and different local mean power (p_{oa}), with equal logarithmic variance (σ_{oa}^2). Each individual packet may be modelled by combining equations 3.6 and 3.8:

$$f(p_a) = \frac{1}{\sqrt{2\pi}\,\sigma_{oa}} \int_0^\infty \frac{1}{p_{oa}^2} \exp\left(-\frac{p_a}{p_{oa}} - \frac{(\ln(p_{oa}) - \ln(\zeta_a))^2}{2\sigma_{oa}^2}\right) dp_{oa} \tag{3.19}$$

3.6.1 Incoherent interference cumulation

When using the p.d.f. given by equation 3.19 for each interfering packet, according to [10] the p.d.f. may be approximated by a log-normal p.d.f. with logarithmic variance:

$$\sigma^2 = \sigma_{oa}^2 + \ln(2) \tag{3.20}$$

and area mean power:

$$\zeta = \frac{\zeta_a}{\sqrt{2}} \tag{3.21}$$

The p.d.f. of the joint instantaneous interference power (p_n) may then be given by again a log-normal p.d.f., with logarithmic variance σ_n^2 and logarithmic area mean $\ln(\zeta_n)$, obtained using the method of Schwartz and Yeh [18],[19], for given σ and $\ln(\zeta)$.

Hence:

$$f(p_n) = \frac{1}{\sqrt{2\pi}\, p_n \sigma_n} \exp\left(-\frac{(\ln(p_n) - \ln(\zeta_n))^2}{2\sigma_n^2}\right) \tag{3.22}$$

Using equation 3.3 with the following substitutions

- $f_n(P_n) \rightarrow f(p_n)$ (equation 3.22)
- $f_a(P_a) \rightarrow f(p_a)$ (equation 3.19)
- $P_n \rightarrow p_n$
- $P_a \rightarrow p_a$

yields:

$$Prob\{\frac{p_a}{p_n} > z_o\} =$$

$$\int_0^{\infty}\int_0^{\infty}\int_{p_n z_o}^{\infty} \frac{1}{2\pi\sigma_{oa}\sigma_n p_n p_{oa}^2} \exp(-\frac{p_a}{p_{oa}} - \frac{(\ln p_{oa} - \ln\zeta_a)^2}{2\sigma_{oa}^2} - \frac{(\ln p_n - \ln\zeta_n)^2}{2\sigma_n^2}) dp_a dp_{oa} dp_n$$

(3.23)

After repeated integration of equation 3.23 and using equation 3.1, Q_k becomes:

$$Q_k = \frac{k}{\sqrt{\pi}} \int_{-\infty}^{\infty} \exp(-w^2) f(w) dw \qquad (3.24)$$

, where [10],[12]:

$$f(w) = \exp\{-\exp\{\ln(\zeta_n) - \ln(\zeta_a) + \ln(z_o) - w\sqrt{2(\sigma_{oa}^2 + \sigma_n^2)}\}\} \qquad (3.25)$$

3.6.2 Coherent interference cumulation

Using equation 3.14, with substitution of p_{on} (the joint local mean power) for $(k-1)p_{oa}$ and equation 3.16, the joint instantaneous interference power p.d.f. may be given by:

$$f(p_n) = \frac{1}{\sqrt{2\pi}\sigma_{on}} \int_0^{\infty} \frac{1}{p_{on}^2} \exp(-\frac{p_n}{p_{on}} - \frac{(\ln(p_{on}) - \ln(\zeta_n))^2}{2\sigma_{on}^2}) dp_{on} \qquad (3.26)$$

ζ_n and σ_{on} are determined as discussed in paragraph 3.4.

Using equation 3.3 with the following substitutions

- $f_n(P_n) \rightarrow f(p_n)$
- $f_a(P_a) \rightarrow f(p_a)$
- $P_n \rightarrow p_n$
- $P_a \rightarrow p_a$

yields:

$$Prob\{\frac{p_a}{p_n} > z_0\} = \frac{1}{2\pi\sigma_{oa}\sigma_{on}} \times$$

$$\int_0^\infty\int_0^\infty\int_0^\infty\int_{z_o p_n}^\infty \frac{1}{p_{on}^2 p_{oa}^2}\exp(-\frac{p_a}{p_{oa}} - \frac{p_n}{p_{on}} - \frac{(\ln p_{oa} - \ln\zeta_a)^2}{2\sigma_{oa}^2} - \frac{(\ln p_{on} - \ln\zeta_n)^2}{2\sigma_{on}^2})\,dp_a\,dp_{oa}\,dp_{on}\,dp_n$$

(3.27)

After repeated integration of equation 3.23, [10],[12] and using equation 3.1, Q_k is given by equation 3.24, with:

$$f(w) = 1 - \frac{1}{1 + \exp(\ln(\zeta_a) - \ln(\zeta_n) - \ln(z_o) - w\sqrt{2(\sigma_{oa}^2 + \sigma_{on}^2)}\,)} \qquad (3.28)$$

Equations of the form of equation 3.24 may be numerically approximated using Hermite Integration [20],[21]. The integral is written as:

$$\int_{-\infty}^{\infty} f(w)\exp(-w^2)\,dw = \sum_{i=1}^{n} v_i f(w_i) + R \qquad (3.29)$$

The sample points w_i and weight factors v_i can be found in [20, table 25.10]. For a large enough number of samples the remainder R vanishes. The number of samples should be increased until the numerical result becomes independent of the number of samples.

3.7 Combined Rayleigh Fading and Near-Far Effect

In this paragraph the local mean power (p_{oa}) equals the area mean power (ζ_a) for each packet. All packets are assumed to have different instantaneous power (p_a) and different area mean power (ζ_a), with equal spatial spread (σ_{da}). By combining equations 3.6 and 3.11 the p.d.f. of p_a is obtained:

$$f(p_a) = \frac{1}{\sqrt{2\pi}\,\sigma_{da}} \int_0^\infty \frac{1}{\zeta_a^2} \exp\left(-\frac{p_a}{\zeta_a} - \frac{(\ln\zeta_a)^2}{2\sigma_{da}^2}\right) d\zeta_a \qquad (3.30)$$

Q_k is derived using the method described in paragraph 3.6 for both incoherent and coherent interference cumulation.

3.7.1 Incoherent interference cumulation

For each interfering packet the p.d.f. given by equation 3.30 may be approximated by a log-normal p.d.f. with logarithmic variance:

$$\sigma^2 = \sigma_{da}^2 + \ln(2) \qquad (3.31)$$

and logarithmic mean:

$$m = -\frac{1}{2}\ln(2) \tag{3.32}$$

The p.d.f. of the joint instantaneous interference power (p_n) is again a log-normal p.d.f., with logarithmic variance σ_n^2 and logarithmic mean m_n, obtained using Schwartz and Yeh's method [18],[19], for given σ and m:

$$f(p_n) = \frac{1}{\sqrt{2\pi}p_n\sigma_n}\exp(-\frac{(\ln(p_n)-m_n)^2}{2\sigma_n^2}) \tag{3.33}$$

In accordance with the derivation of paragraph 3.6.1, Q_k is given by equation 3.24, with:

$$f(w) = \exp\{-\exp\{m_n + \ln(z_o) - w\sqrt{2(\sigma_{da}^2+\sigma_n^2)}\ \}\} \tag{3.34}$$

3.7.2 Coherent interference cumulation

Using equation 3.14, with substitution of ζ_n (the joint area mean power) for $(k-1)p_{oa}$ and using equation 3.17, the joint instantaneous interference power p.d.f. is given by:

$$f(p_n) = \frac{1}{\sqrt{2\pi}\sigma_{dn}}\int_0^\infty \frac{1}{\zeta_n^2}\exp(-\frac{p_n}{\zeta_n}-\frac{(\ln(\zeta_n)-m_n)^2}{2\sigma_{dn}^2})d\zeta_n \tag{3.35}$$

where m_n and σ_{dn} are determined as discussed in paragraph 3.5.
In accordance with the derivation of paragraph 3.6.2, Q_k is given by equation 3.24, with:

$$f(w) = 1 - \frac{1}{1 + \exp(-m_n - \ln(z_o) - w\sqrt{2(\sigma_{da}^2 + \sigma_{dn}^2)}\ )} \tag{3.36}$$

Again Hermite integration is used to find approximated values for Q_k.

3.8 Combined Shadowing and Near-Far Effect

In this paragraph, all packets are assumed to have different local mean power (p_{oa}), with equal logarithmic variance (σ_{oa}^2) and different area mean power (ζ_a), with equal spatial spread (σ_{da}).
By combining equations 3.8 and 3.11 the p.d.f. of p_{oa} is obtained:

$$f(p_{oa}) = \frac{1}{2\pi\sigma_{oa}\sigma_{da}p_{oa}} \int_0^{\infty} \frac{1}{\zeta_a} \exp\left(-\frac{(\ln(p_{oa}) - \ln(\zeta_a))^2}{2\sigma_{oa}^2} - \frac{(\ln\zeta_a)^2}{2\sigma_{da}^2}\right) d\zeta_a \tag{3.37}$$

, which becomes:

$$f(p_{oa}) = \frac{1}{\sqrt{2\pi}\sqrt{\sigma_{oa}^2 + \sigma_{da}^2}\ p_{oa}} \exp\left(-\frac{(\ln p_{oa})^2}{2(\sigma_{oa}^2 + \sigma_{da}^2)}\right) \tag{3.38}$$

Hence, p_{oa} is described by a log-normal p.d.f. with zero mean and logarithmic variance $\sigma_{oa}^2+\sigma_{da}^2$.

Q_k Is derived using the same method as described in paragraph 3.5. Hence, Q_k is given by equations 3.16a&b and:

$$x = \frac{-m_n - \ln z_o}{\sqrt{\sigma_{oa}^2+\sigma_{da}^2+\sigma_{on}^2}} \qquad (3.39)$$

Where m_n and σ_{on}^2 are the logarithmic mean and logarithmic variance respectively, of the joint interference power p.d.f. ($f(p_{on})$), obtained using Schwartz and Yeh's method [18],[19], for given spread ($\surd(\sigma_{oa}^2+\sigma_{da}^2)$) and zero mean.

3.9 Combined Rayleigh Fading, Shadowing and Near-Far Effect

In this paragraph, all packets are assumed to have different instantaneous power (p_a), different local mean power (p_{oa}), with equal logarithmic variance (σ_{oa}^2) and different area mean power (ζ_a), with equal spatial spread (σ_{da}).

By combining equations 3.6 and 3.38 the p.d.f. of p_a is obtained:

$$f(p_a) = \frac{1}{\sqrt{2\pi}\sqrt{\sigma_{oa}^2+\sigma_{da}^2}} \int_0^\infty \frac{1}{p_{oa}^2} \exp\left(-\frac{p_a}{p_{oa}} - \frac{(\ln p_{oa})^2}{2(\sigma_{oa}^2+\sigma_{da}^2)}\right) dp_{oa} \qquad (3.40)$$

Q_k is derived using the method described in paragraph 3.6 for both incoherent and coherent interference cumulation.

3.9.1 Incoherent interference cumulation

For each interfering packet the p.d.f. given by equation 3.40 may be approximated by a log-normal p.d.f. with logarithmic variance

$$\sigma^2 = \sigma_{oa}^2 + \sigma_{da}^2 + \ln(2) \tag{3.41}$$

and logarithmic mean

$$m = -\frac{1}{2}\ln(2) \tag{3.42}$$

The p.d.f. of the joint instantaneous interference power (p_n) is again a log-normal p.d.f., with logarithmic variance σ_n^2 and logarithmic mean m_n, obtained using Schwartz and Yeh's method [18],[19], for given σ and m.

This p.d.f. is given by equation 3.33.

In accordance with the derivation of paragraph 3.6.1, Q_k is given by equation 3.24, with:

$$f(w) = \exp\{-\exp\{m_n + \ln(z_o) - w\sqrt{2(\sigma_{da}^2+\sigma_{oa}^2+\sigma_n^2)}\ \}\} \tag{3.43}$$

3.9.2 Coherent interference cumulation

The joint instantaneous interference power p.d.f. is given by:

$$f(p_n) = \frac{1}{\sqrt{2\pi}\,\sigma_{on}} \int_0^{\infty} \frac{1}{p_{on}^2} \exp\left(-\frac{p_n}{p_{on}} - \frac{(\ln(p_{on}) - m_n)^2}{2\sigma_{on}^2}\right) dp_{on} \tag{3.44}$$

where m_n and σ_{on} are determined as discussed in paragraph 3.8 and p_{on} is the joint local mean power.

In accordance with the derivation of paragraph 3.6.2, Q_k is given by equation 3.24, with:

$$f(w) = 1 - \frac{1}{1 + \exp(-m_n - \ln(z_o) - w\sqrt{2(\sigma_{oa}^2 + \sigma_{da}^2 + \sigma_{on}^2)}\,)} \tag{3.45}$$

Again Hermite integration is used to find approximated values for Q_k.

3.10 Conclusion

The method of Schwartz and Yeh, as frequently used in the previous paragraphs, determines the logarithmic mean and logarithmic variance of n log-normal variables, given their individual logarithmic mean and logarithmic variance as input. An increase in the logarithmic mean (input) gives an equal increase in the joint logarithmic mean (output).

Logarithmic mean and joint logarithmic mean are subtracted in those capture models where shadowing is included. In those cases the choice of logarithmic mean (as input) will have no effect on the capture probability. By assuming the area logarithmic mean ($\ln\zeta_a$) to be zero in case of shadowing, it may be seen that the influence of shadowing and the near-far effect is fully determined by the equations given for their combined effect (with or without Rayleigh fading).

Expressing the total logarithmic variance as:

$$\sigma^2 = \sigma_{oa}^2 + \sigma_{da}^2 \tag{3.46}$$

, one of them may be excluded by setting the corresponding logarithmic variance to zero ($\sigma_{oa}^2=0$ to exclude shadowing and $\sigma_{da}^2=0$ to exclude the near-far effect). When they are both excluded the above does not hold and the equations given for Rayleigh fading only are used.

Equation 3.46 also shows that by combining the near-far effect and shadowing the total logarithmic variance is increased.

4 COMPUTATIONAL RESULTS

The results are given for a capture ratio of z_o=4.0 (6.0 dB). Shadowing will be modelled using a shadowing spread of σ_{oa}=6 dB. The near-far effect will be modelled using a spatial spread of σ_{da}=8 dB. (If both shadowing and near-far effect are included in a capture model, this results in a total spread of 10 dB).

Figure 4.1 depicts λ_{cr} (the critical generation rate) as a function of p_s (the probability that a packet is moved to stack level one after a conflict), for all three algorithms (A1, A2 and A3) and for combined coherent Rayleigh fading, shadowing and near far-effect. Figure 4.2 depicts the incoherent counterpart of figure 4.1.

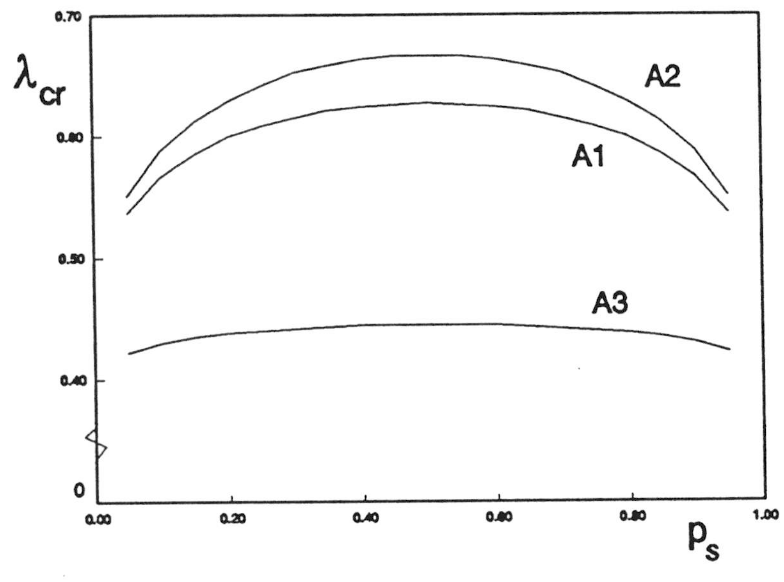

Figure 4.1 Critical generation rate (λ_{cr}, in packets/slot) as a function of the probability that a packet is moved to stack level one after a conflict (p_s), for all three algorithms (A1, A2 and A3) and for combined coherent Rayleigh fading, shadowing (σ_{oa}=6 dB) and near-far effect (σ_{da}=8 dB).

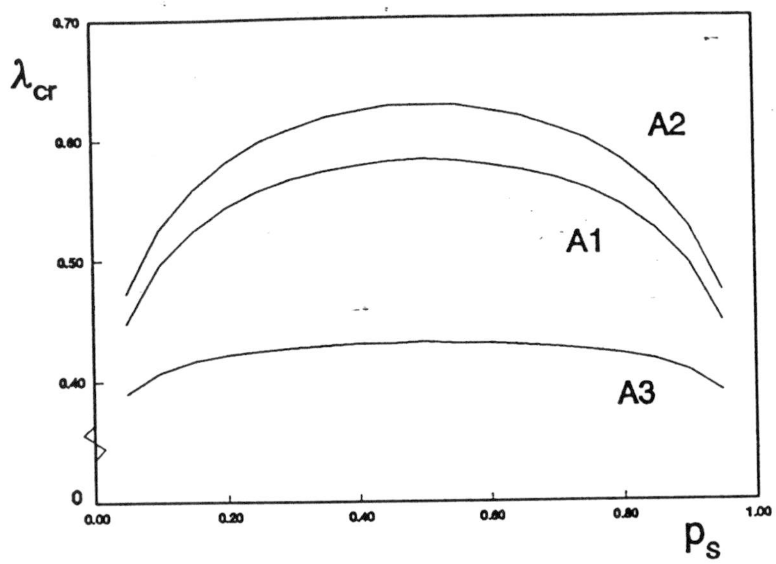

Figure 4.2 Critical generation rate (λ_{cr}, in packets/slot) as a function of the probability that a packet is moved to stack level one after a conflict (p_s), for all three algorithms (A1, A2 and A3) and for combined incoherent Rayleigh fading, shadowing (σ_{oa}=6 dB) and near-far effect (σ_{da}=8 dB).

Figure 4.1 and 4.2 show that the critical generation rate has a maximum for p_s=0.5 By comparing these two figures it is seen that the coherent case has higher critical values than the incoherent case for all p_s and for each algorithm.

Figure 4.3 shows D (the mean packet delay) as a function of p_s, for combined coherent Rayleigh fading, shadowing and near-far effect. For algorithms A1 and A2 a mean generation rate (λ) of 0.5 packets per slot is used, for algorithm A3 a mean generation rate of 0.4 packets per slot. The figure shows that for algorithm A1 and A2, the delay has its minimum at approx. p_s=0.45, but hardly increases around that value of p_s. For A3 the delay has a minimum around p_s=0.25, slowly increasing for increasing p_s. Figures 4.1-4.3 show that the algorithms have best overall performance for p_s=0.5. Therefore throughput and delay curves will be given for this value of p_s.

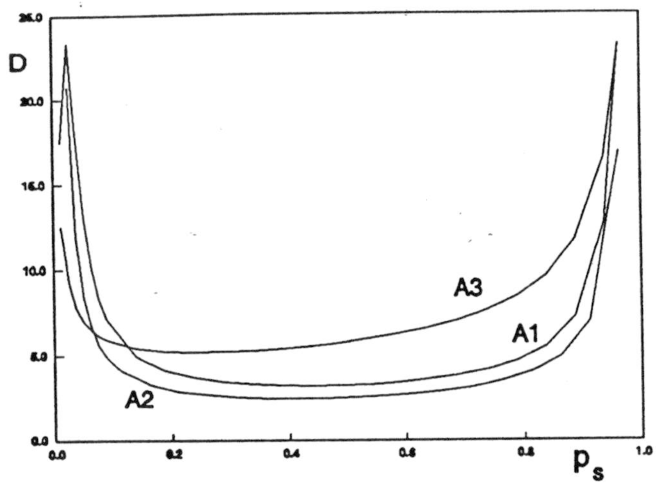

Figure 4.3 Mean packet delay (D, in slots/packet) as a function of the probability that a packet is moved to stack level one after a conflict (p_s), for all three algorithms (A1, A2 and A3) and for combined coherent Rayleigh fading, shadowing (σ_{oa}=6 dB) and near-far effect (σ_{da}=8 dB). For A1 and A2 a mean generation rate (λ) of 0.5 packets per slot is used, for A3 a mean generation rate of 0.4 packets per slot.

Figure 4.4 depicts the throughput (S) as a function of the mean total transmission rate (G), for each algorithm (A1, A2 and A3, with p_s=0.5) and for the ALOHA algorithm. The capture model includes coherent Rayleigh fading, shadowing and the near-far effect. Figure 4.5 depicts D as a function of λ, for each algorithm, using the same capture model. The incoherent counterparts of figure 4.4 and 4.5 are shown in figure 4.6 and figure 4.7. In appendix B, throughput curves are given for all other capture models discussed in chapter 3 (the delay curves are all of the form of figure 4.5 and 4.7; towards the critical generation rate the delay grows to infinity).
The calculated throughput equalled the mean generation rate, for all values up to the critical generation rate. Hence, the critical generation rate is the maximum throughput given for each algorithm in each of those figures.

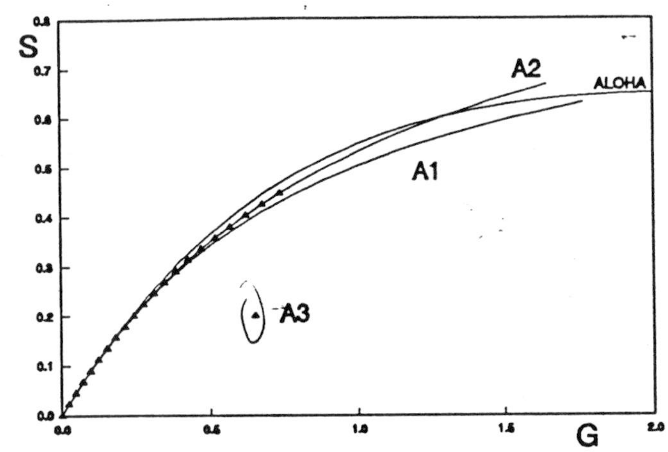

Figure 4.4 Throughput (S, in packets/slot) as a function of the mean total transmission rate (G, in packets/slot), for each algorithm (A1, A2 and A3, with p_s=0.5) and for the ALOHA algorithm. The capture model includes coherent Rayleigh fading, shadowing (σ_{oa}=6 dB) and near-far effect (σ_{da}=8 dB).

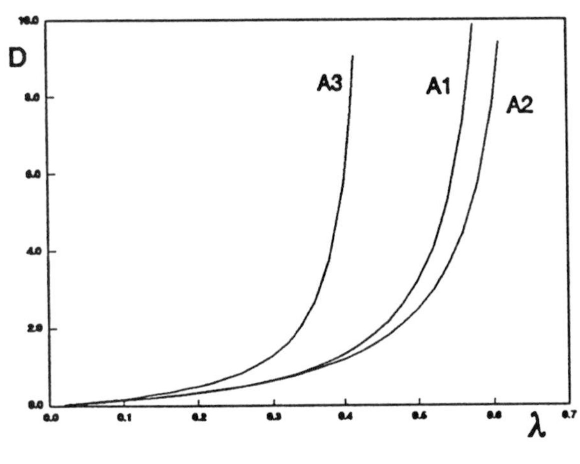

Figure 4.5 Mean delay (D, in slots/packet) as a function of the mean generation rate (λ, in packets/slot), for each algorithm (A1, A2 and A3, with p_s=0.5). The capture model includes coherent Rayleigh fading, shadowing (σ_{oa}=6 dB) and near-far effect (σ_{da}=8 dB).

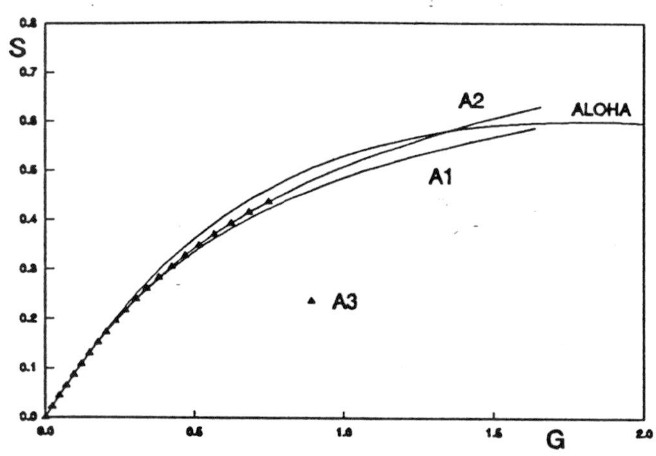

Figure 4.6 Throughput (S, in packets/slot) as a function of the mean total transmission rate (G, in packets/slot), for each algorithm (A1, A2 and A3, with p_s=0.5) and for the ALOHA algorithm. The capture model includes incoherent Rayleigh fading, shadowing (σ_{oa}=6 dB) and near-far effect (σ_{da}=8 dB).

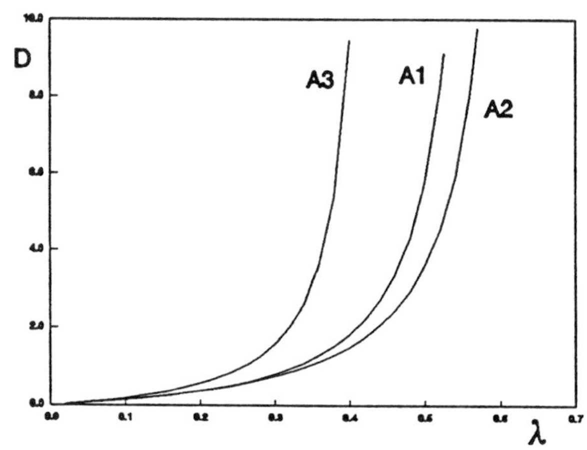

Figure 4.7 Mean delay (D, in slots/packet) as a function of the mean generation rate (λ, in packets/slot), for each algorithm (A1, A2 and A3, with p_s=0.5). The capture model includes incoherent Rayleigh fading, shadowing (σ_{oa}=6 dB) and near-far effect (σ_{da}=8 dB).

The figures in this chapter and in appendix B show that algorithm A2 performs best of all three versions of the stack algorithm (highest throughputs, highest critical generation rates and lowest delays). Algorithm A3 has highest delays and lowest critical generation rates. Up to those rates the throughput curves of A3 follow the corresponding curves of algorithm A2.

Comparing the throughput curves of the stack algorithm with those of the ALOHA algorithm, assuming there are no discrepancies between the two algorithms in modelling the total number of transmissions, the ALOHA algorithm performs slightly better than the stack algorithm. However, near the critical generation rates of A1 and A2, where the throughput of the ALOHA algorithm is about to drop or has started to drop, stack algorithms A1 and A2 maintain high throughputs. The highest achievable throughputs of A2 are higher than those of the ALOHA algorithm.

In figures 4.8 and 4.9 the throughput curves of A2 are compared for different capture models. Figure 4.8 shows that a higher value of the logarithmic variance leads to higher throughputs. (Coherent Rayleigh fading was added to figure 4.8 to form a reference point for figure 4.9). Figure 4.9 shows that coherent Rayleigh fading offers higher throughputs than incoherent Rayleigh fading. From figure 4.8 and 4.9 it may be concluded that combined effects lead to higher throughputs.

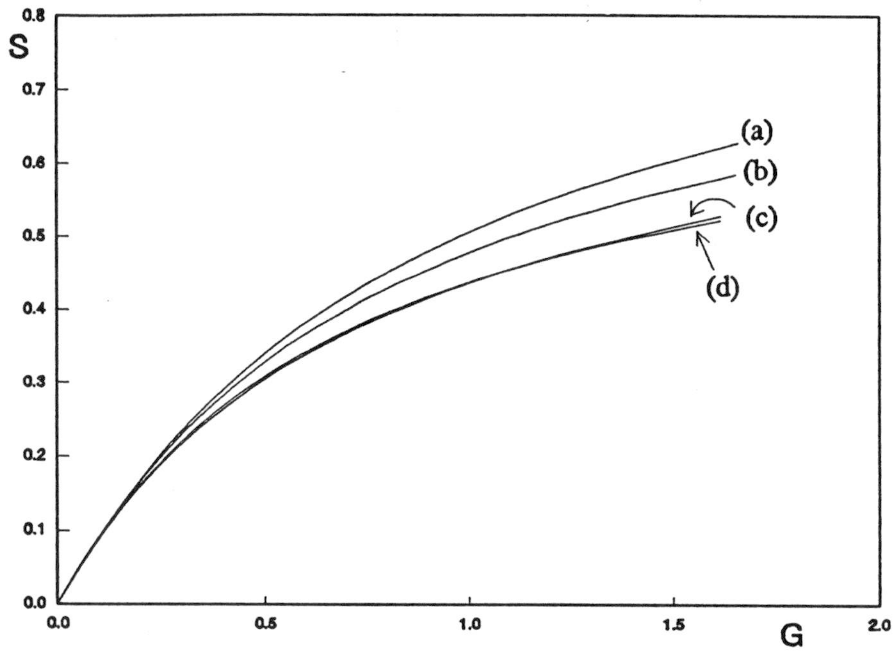

Figure 4.8 Throughput (S, in packets/slot) as a function of the mean total transmission rate (G, in packets/slot), for algorithm A2 with p_s=0.5.

a. Combined shadowing and near far effect (σ_{oa}=6 dB, σ_{da}=8 dB, resulting in a total spread of 10 dB)

b. Near far effect only (σ_{da}=8 dB)

c. Coherent Rayleigh fading only

d. Shadowing only (σ_{oa}=6 dB).

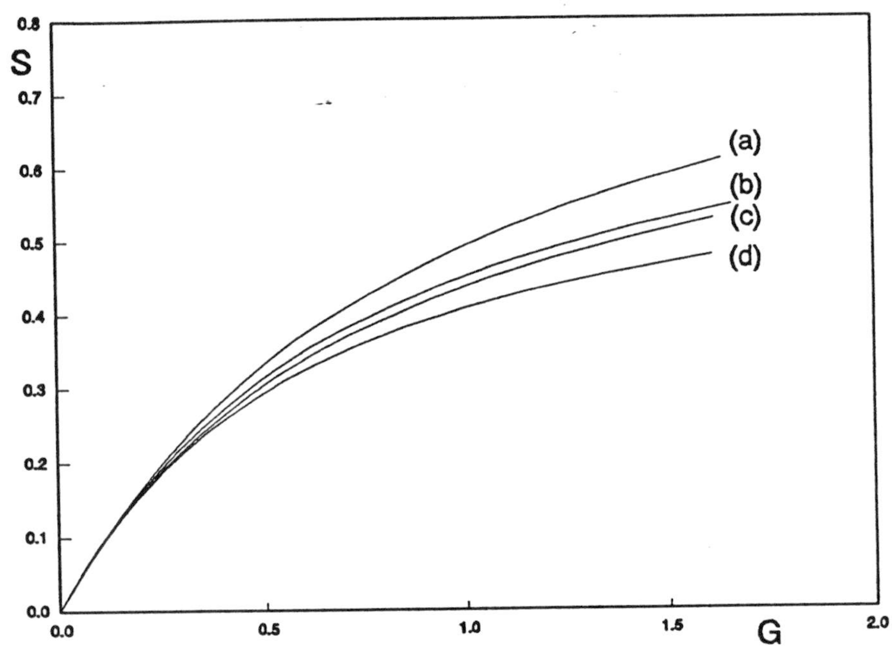

Figure 4.9 Throughput (S, in packets/slot) as a function of the mean total transmission rate (G, in packets/slot), for algorithm A2 with p_s=0.5.

a. Combined coherent Rayleigh fading and shadowing (σ_{oa}=6 dB)

b. Combined incoherent Rayleigh fading and shadowing (σ_{oa}=6 dB)

c. Coherent Rayleigh fading only

d. Incoherent Rayleigh fading only.

5 CONCLUSIONS AND RECOMMENDATIONS

The performance of the stack algorithm in mobile radio channels with Rayleigh fading, shadowing and near-far effect, has been analyzed.

Three versions of the stack algorithm were taken into consideration, each with a different ability to distinguish between channel events in a previous time slot.

The stack algorithm was described as a regenerative process. Throughput and delay characteristics have been determined up to a critical generation rate of traffic, where the mean packet delay and mean basic session length grow to infinity.

It was shown that capture models based on combined effects offer higher throughputs. For (combined) shadowing and near-far effect it can be said that a higher value of the logarithmic variance leads to higher throughputs.

By comparing coherent Rayleigh fading with incoherent Rayleigh fading it was shown that the former offers higher throughputs than the latter.

Algorithm A2 has shown to perform better than the other versions.

By comparing the throughput curves of the stack algorithm with those of the ALOHA algorithm, under the assumption that there are no discrepancies between the two algorithms in modelling the total number of transmissions, it was shown that the ALOHA algorithm performs slightly better than the stack algorithm. However, near the critical generation rates of A1 and A2, where the throughput of the ALOHA algorithm is about to drop or has started to drop, stack algorithms A1 and A2 maintain high throughputs. The highest achievable throughputs of A2 are higher than those of the ALOHA algorithm. [?]

Therefore, it is worth developing another model to describe the stack algorithm beyond the critical generation rate. Beyond the critical generation rate the stack algorithm can no longer be described as a regenerative process, since the regeneration cycle (basic session length) is no longer finite.

The stack algorithm may be modelled by determining t_k (the probability that k packets are transmitted in a time slot) based on the channel event in the previous time slot. The only difficulty in that respect, is to predict what happens if a stack level reduction occurs (when zero or one packet is transmitted). The number of packets returning from the stack is not simply a function of t_k since not all upper sessions that may correspond to k, are equally likely to result in a finite upper session.

Choosing an appropriate value of p_s should also be reconsidered.

The key factor in maintaining high throughputs is to keep the probability of high loads low, since the capture probability decreases with higher conflict multiplicities. Up to the critical point this is done by taking $p_s=0.5$, since in that case traffic is evenly distributed over time. Beyond the critical point it may be useful to select higher values for p_s, either to reduce the probability of infinite upper sessions or to at least keep the upper session load low.

Another aspect that should be taken into consideration is that a transmitter will not allow its packet to get lost on the stack. It is more likely that a transmitter will use some (random) time out interval, so that when that interval expires it will transmit its packet regardless of at what level it was. In that case the stack algorithm shows similarities to the ALOHA algorithm.

A final remark concerns the near-far effect. The near-far effect is notorious for the fact that transmitters near the base station are more likely to capture the receiver than transmitters far away and that therefore most retransmissions will come from further away. One may expect that this will be less so with the stack algorithm. When a conflict occurs part of the colliding packets will be sent to stack level one, and the other part will be retransmitted immediately. In this, no distinction is made between strong or weak packets. All packets sent to level one will have to wait until the packets transmitted immediately (weak and strong), are successfully transmitted.

Fairness ?

REFERENCES

[1] B.S. Tsybakov, "Survey of USSR Contributions to Random Multiple Access Communications", *IEEE Transactions on Information Theory*, Vol. IT-31, No. 2, pp. 143-165, March 1985.

[2] N.D. Vvedenskaya, B.S. Tsybakov, "Packet Delay for Multi-access Stack Algorithm", *Probl. Peredachi Inform.*, Vol. 20, pp. 89-101, April-June 1984.

[3] N.D. Vvedenskaya, J.C. Arnbak, B.S. Tsybakov, "Packet Delay for Stack Algorithm in a Channel with Capture", *Fifth Joint Soviet-Swedish International Workshop on Information Theory*, pp. 167-170, January 1991.

[4] M. Sidi, I. Cidon, "Splitting Protocols in Presence of Capture", *IEEE Transactions on Information Theory*, Vol. IT-31, No. 2, pp. 295-301, March 1985.

[5] L. Georgiadis, L.F. Merakos, P. Papantoni-Kazakos, "A Method for the Delay Analysis of Random Multiple-Access Algorithms Whose Delay Process is Regenerative", *IEEE Journal on Selected Areas in Communications*, Vol. SAC-5, No. 6, July 1987.

[6] J.A. White, J.W. Smidt, G.K. Bennett, *Analysis of Queueing Systems*, Academic Press, New York, 1975.

[7] L. Kleinrock, *Queueing Systems*, Vol. 1&2, Wiley, New York, 1975,1976.

[8] Andrew S. Tanenbaum, *Computer Networks*, 2nd edition, Prentice-Hall International Inc., 1989.

[9] J.C. Arnbak, W. van Blitterswijk, "Capacity of Slotted ALOHA in Rayleigh-Fading Channels", *IEEE Journal on Selected Areas in Communications*, Vol. SAC-5, No. 2, pp.261-269, February 1987.

[10] R. Prasad, "Throughput Analysis of Nonpersistent Inhibit Sense Multiple Access in Multipath Fading and Shadowing Channels", *European Trans. on Telecommunications and Related Technologies*, Vol. 2, No. 3, pp. 313-318, May,June, 1991.

[11] W.C.Y. Lee, *Mobile Communications Design Fundamentals*, Howard W. Sams & Co., Indiana, 1986.

[12] R. Prasad, A. Kegel, "Improved Assessment of Interference Limits in Cellular Radio Performance", accepted for publication *IEEE Transactions on Vehicular Technology*, 1991.

[13] R. Prasad, J.C. Arnbak, "Enhanced Throughput in Packet Radio Channels with Shadowing", *Electronic Letters,* Vol. 24, No. 16, pp. 986-987, August 1988.

[14] J.P.M.G. Linnartz, "Near-Far Effect in Some Random Access Radio Systems with Fading", *Proceedings of the 1990 Bilkent International Conference on New Trends in Communication, Control, and Signal Processing*, Vol. 1, pp.504-510, Bilkent University, Ankara, Turkey, July 1990.

[15] C. Namislo, "Analysis of Mobile Radio Slotted Networks", *IEEE Journal on Selected Areas in Communications*, Vol. SAC-2, pp. 583-588, July 1984.

[16] C. van der Plas, J.P.M.G. Linnartz, "Stability of Mobile Slotted ALOHA Network with Rayleigh Fading, Shadowing and Near-Far Effect", *IEEE Transactions on Vehicular Technology*, Vol. 39, No. 4, November 1990.

[17] H.J. Larson, *Introduction to Probability Theory and Statistical Inference*, third edition, Wiley, 1982.

[18] R. Prasad, J.C. Arnbak, "Comments on "Analysis for Spectrum Efficiency in Single Cell Trunked and Cellular Mobile Radio"", *IEEE Transactions on Vehicular Technology*, Vol. 37, NO. 4, pp. 220-222, 1988.

[19] S.C. Schwartz, Y.S. Yeh, "On the Distribution Function and Moments of Power Sums with Log-normal Components", *Bell Syst. Tech. J.*, Vol. 24, pp.316-317, September 1982.

[20] M. Abramowitz, I.A. Stegun, *Handbook of Mathematical Functions*, Dover Publications Inc., New York, 1972.

[21] W.H. Press, B.P. Flannery, S.A. Teukolsky, W.T. Vetterling, *Numerical Recipes in Pascal*, Cambridge University Press, Cambridge, 1989.

APPENDIX A SPATIAL DISTRIBUTION OF TRAFFIC

Let G_t denote the total traffic (in packets/slot) offered to the channel and let $G(\rho)$ denote the spatial distribution of traffic per unit area at normalized distance ρ (equation 3.10) then:

$$G_t = 2\pi \int_0^{\infty} G(\rho)\rho d\rho \tag{A.1}$$

The probability that a packet is transmitted within distance ρ from the base station $F(\rho)$, may then be given by:

$$F(\rho) = \frac{2\pi}{G_t} \int_0^{\rho} G(x) x dx \tag{A.2}$$

, with corresponding p.d.f:

$$f(\rho) = \frac{2\pi}{G_t} G(\rho)\rho \tag{A.3}$$

Using equation 3.9 and the property

$$f(\zeta_a) = f(\rho) \left| \frac{d\rho}{d\zeta_a} \right| \tag{A.4}$$

yields the p.d.f. of the area mean power.

Consider the case of a (quasi-)uniform distribution of traffic over the cell area [9],[14],[15].

Then $G(\rho)$ is (almost) equal to G_t/π within the cell ($\rho \leq 1$) and is (almost) zero outside (see fig A.1), and $f(\rho)$ (almost) equals 2ρ within the cell and is (almost) zero outside.

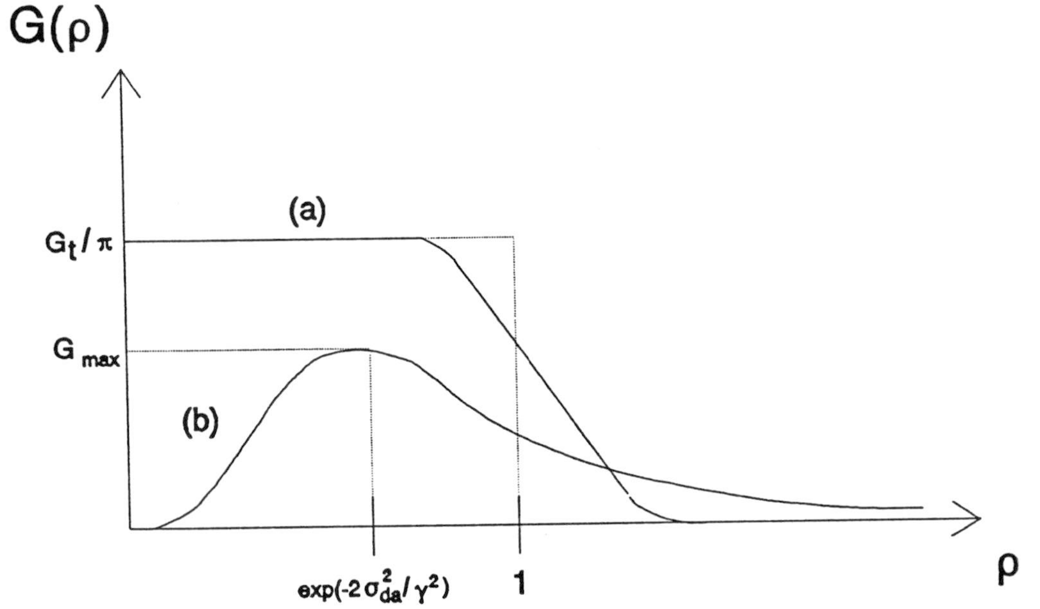

Figure A.1: Distribution of traffic for (a) quasi uniform and (b) log-normal distribution. $G(\rho)$ has maximum at $\rho = \exp(-2\sigma_{da}^2/\gamma^2)$.

A very undesirable and unrealistic property of a (quasi-)uniform distribution over the cell area, is that it "allows" packets to be transmitted very close to the station, causing infinite area mean power. All moments of the area mean power ($E\{\zeta_a^n\}$) will be undefined ($f(\zeta_a)=2\zeta^{-(\gamma+2)/\gamma}\gamma^{-1}$, determined by using equation 3.9 and A.4

and the fact that $f(\rho)=2\rho$). In that respect, the assumption of a log-normal spatial distribution with zero mean is a more realistic one [16]:

$$f(\rho) = \frac{1}{\sqrt{2\pi}\,\rho(\sigma_{da}/\gamma)} \exp\{-\frac{(\ln\rho)^2}{2(\sigma_{da}/\gamma)^2}\} \qquad \textbf{(A.5)}$$

$(\sigma_{da}/\gamma)^2$ Is the logarithmic variance, where $\sigma_{da}^{\,2}$ is the logarithmic variance of the area mean power ($\mathrm{var}(\ln\zeta_a) = \mathrm{var}(-\gamma\ln\rho) = \gamma^2\,\mathrm{var}(\ln\rho)$ [17]).

Using equation A.3, $G(\rho)$ becomes:

$$G(\rho) = \frac{\gamma G_t}{(2\pi)^{3/2}\rho^2\sigma_{da}} \exp\{-\frac{(\gamma\ln\rho)^2}{2\sigma_{da}^2}\} \qquad \textbf{(A.6)}$$

Now $\lim_{\rho\to 0} G(\rho)=0$ (for finite σ_{da}). $G(\rho)$ has a maximum at $\rho=\exp(-2\sigma_{da}^{\,2}/\gamma^2)$, hence by choosing a low value for σ_{da} packets are assumed to come from far away and vice versa (see figure A.1). This makes it possible to examen the near-far effect for different types of traffic.

Using equation A.4 and equation 3.9, $f(\varsigma_a)$ is given by:

$$f(\zeta_a) = \frac{1}{\sqrt{2\pi}\,\zeta_a\sigma_{da}} \exp\{-\frac{(\ln\zeta_a)^2}{2\sigma_{da}^2}\} \qquad \textbf{(A.7)}$$

Hence, the area mean power becomes a log-normal random variable with zero mean and all moments of the area mean power are finite (equation A.7 is equal to equation 3.11).

APPENDIX B ADDITIONAL RESULTS

This appendix contains additional throughput curves, to the ones given in chapter 4.

The results are given for a capture ratio of z_o=4.0 (6.0 dB). Shadowing is modelled using a shadowing spread of σ_{oa}=6 dB. The near-far effect is modelled using a spatial spread of σ_{da}=8 dB. (If both shadowing and the near-far effect are included in a capture model, this results in a total spread of 10 dB). The probability that a packet is moved to stack level one after a conflict (p_s) equals 0.5.

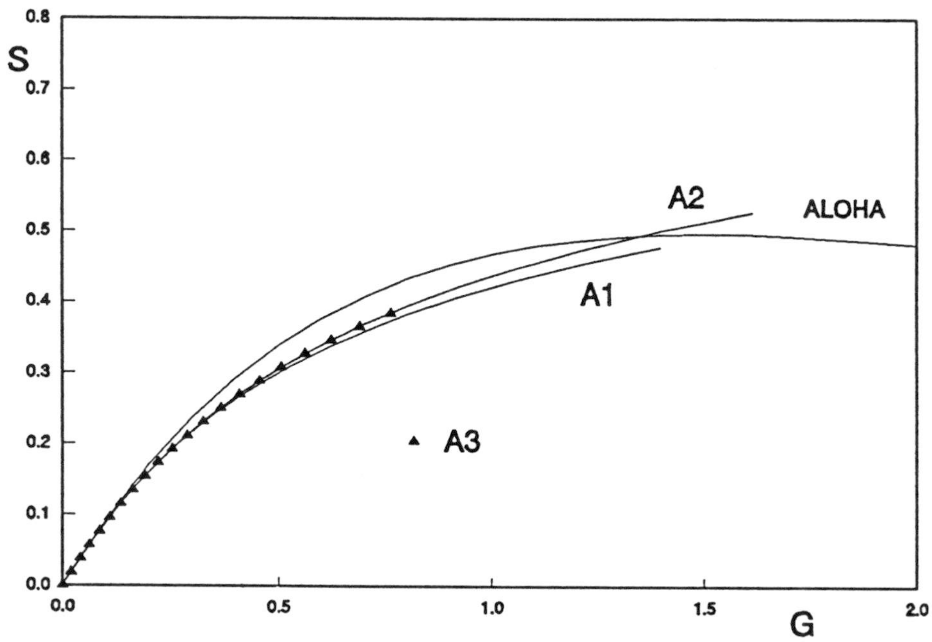

Figure B.1 Coherent Rayleigh fading only.

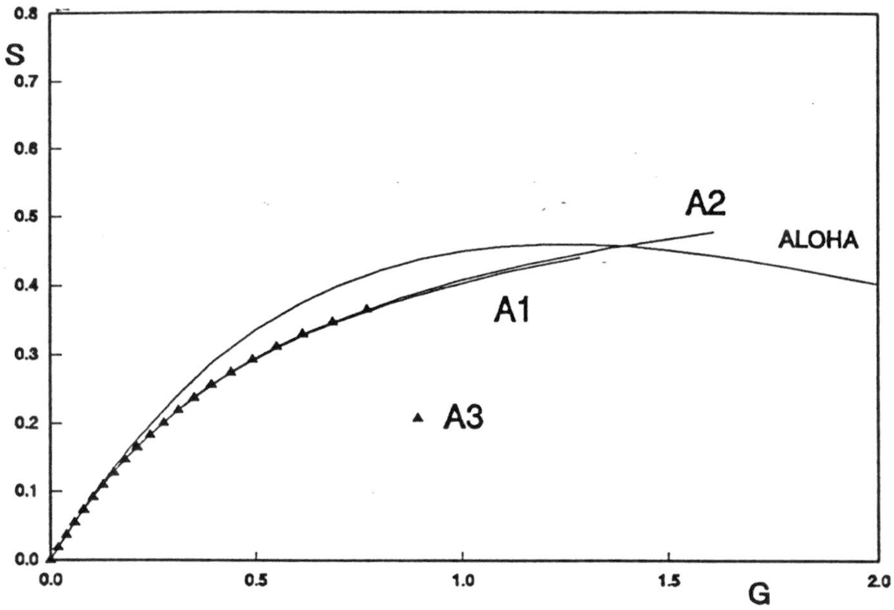

Figure B.2 Incoherent Rayleigh fading only.

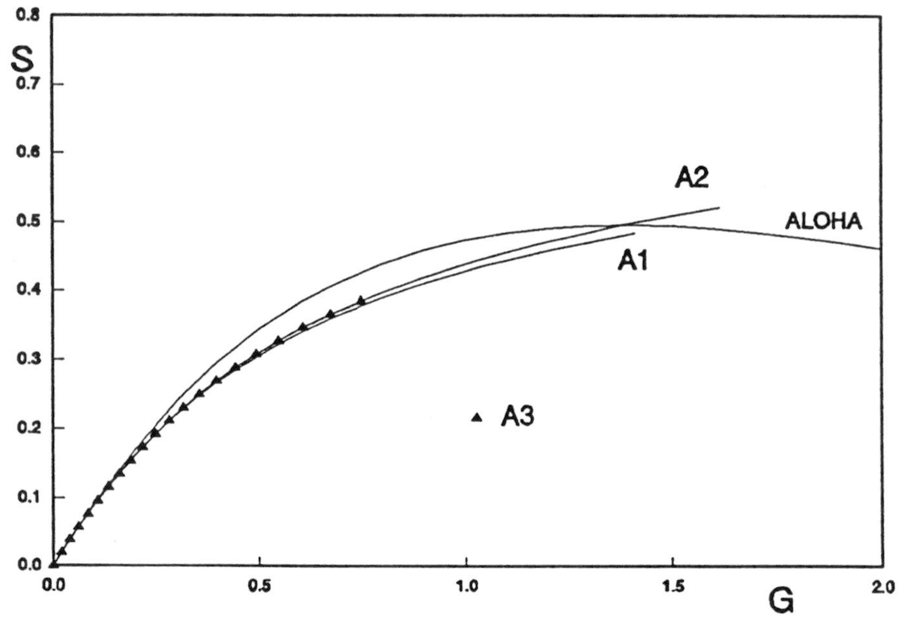

Figure B.3 Shadowing only.

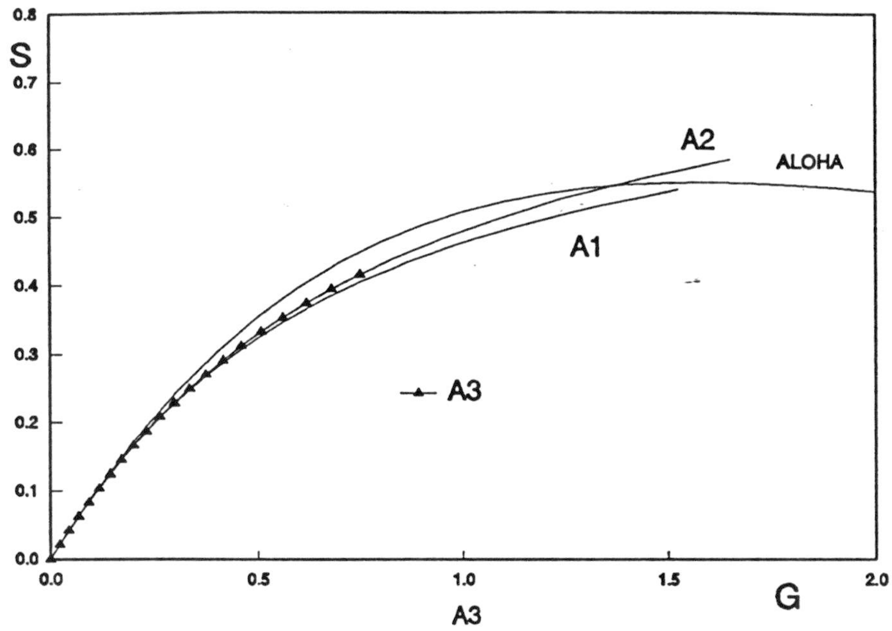

Figure B.4 Near far effect only.

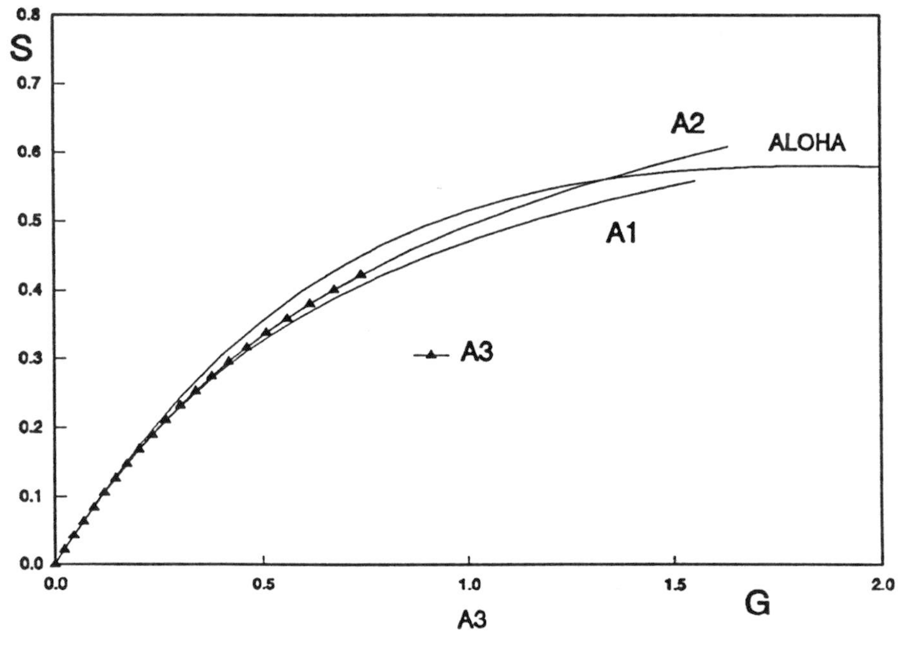

Figure B.5 Combined coherent Rayleigh fading and shadowing.

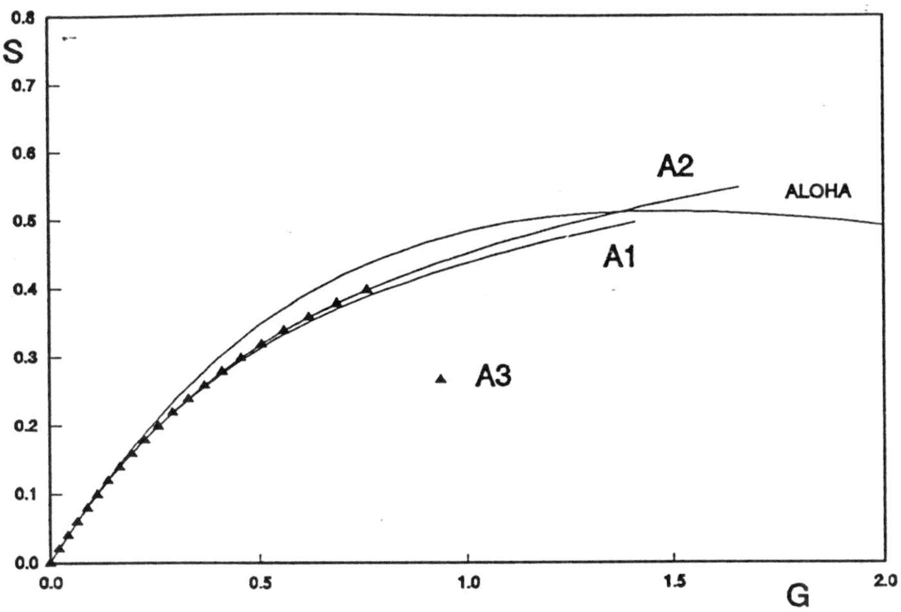

Figure B.6 Combined incoherent Rayleigh fading and shadowing.

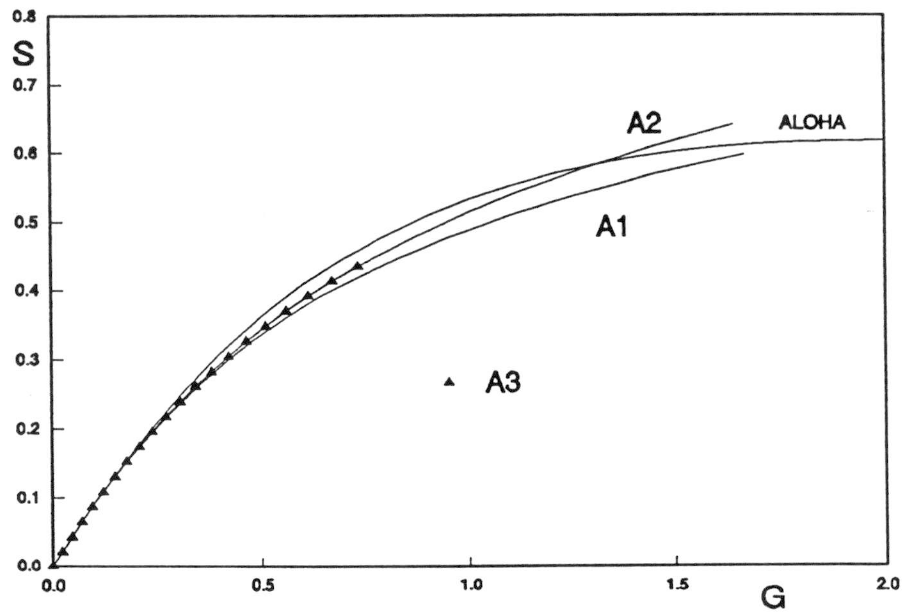

Figure B.7 Combined coherent Rayleigh fading and near far effect.

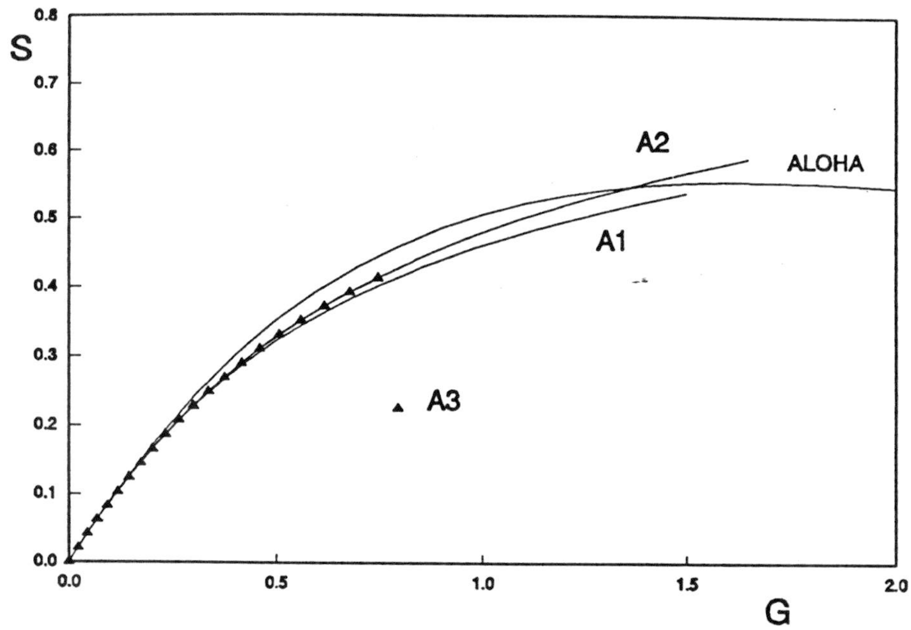

Figure B.8 Combined incoherent Rayleigh fading and near far effect.

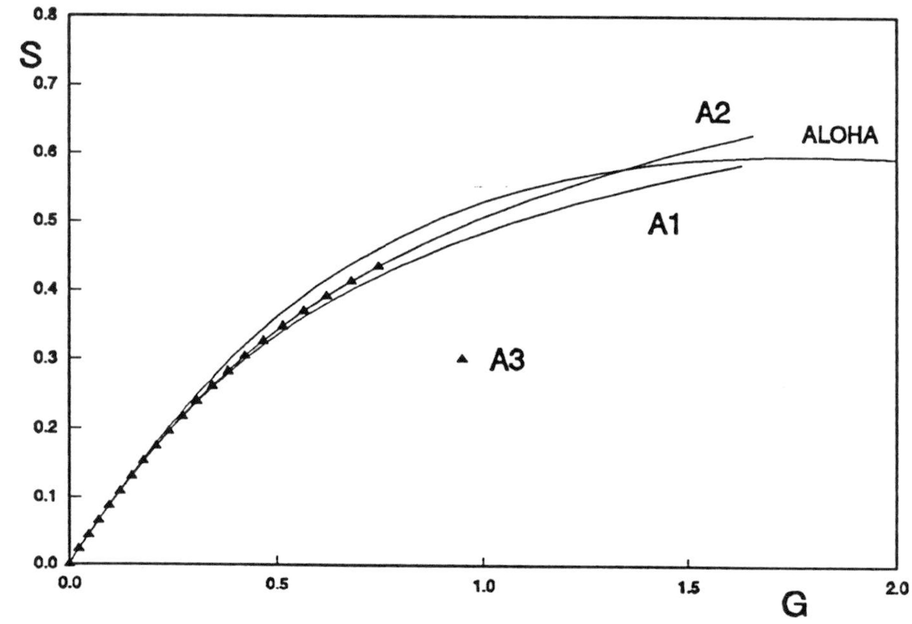

Figure B.9 Combined shadowing and near far effect.