Proceedings of the 1998 Winter Simulation Conference
D.J. Medeiros, E.F. Watson, J.S. Carson and M.S. Manivannan, eds.

INTERFERENCE RADIUS IN PCS RADIO RESOURCE MANAGEMENT SIMULATIONS

Michael Liljenstam
Rassul Ayani

Department of Teleinformatics
Royal Institute of Technology
KTH-Electrum, S-164 40 Kista (Stockholm), SWEDEN

ABSTRACT

Simulation is widely used for performance analysis of Personal Communication Systems due to their inherent complexity. However, resource demands (cpu-time and memory requirements) for this type of simulation are often high. Consequently, simulation fidelity demands must be carefully weighed against available computer resources in modeling. In models including measurements of interference from other transmitters, one question that arises is up to what distance other transmitters must be included. This parameter has a direct impact on the amount of work performed in each interference calculation, and is also of great importance for parallel and distributed models since it influences partitioning. In this paper, the impact of the interference (or interaction) radius on fidelity and execution time is studied for a case model.

1 INTRODUCTION

The number of subscribers in Personal Communication System (PCS) networks has grown rapidly over the last couple of years, and new systems are forthcoming with more bandwidth-intense services. Thus, PCS manufacturers and network operators need to tune existing networks and new designs for improved utilization of their given part of the frequency spectrum. Given the complexity of the systems, simulation is widely used to study the effects of proposed new strategies and improvements. However, due to the often extremely resource demanding nature of these simulations, it is important to model the system as efficiently as possible and limit the calculation work involved whenever model fidelity permits. In this paper we study the effect of one parameter, the radius of interference calculations, and its impact on model fidelity and execution time. Furthermore, if one wishes to pursue reduction in execution time and simulation of larger systems, this parameter could be important since it affects the possibility of parallelizing the model, i.e. applying Parallel Discrete Event Simulation (PDES) techniques. Parallelization of PCS models and models of related systems has attracted considerable attention in the last couple of years, e.g. (Carothers et al., 1995), (Greenberg et al., 1994), (Harju et al., 1993), and (Liu et al., 1996).

A Personal Communication System (PCS) provides wireless communication services to subscribers within its service area. The service area is populated by a number of Base Stations (BS), where the coverage of each BS is called a cell. Users carrying Mobile Terminals (MTs) roaming the service area may communicate through one of the BSs to reach other parts of the network or out through the fixed network. Since available radio frequency spectrum is limited, frequencies used in one cell are reused in other cells that are sufficiently far away not to cause substantial interference (frequency reuse). The problem of radio resource management is one important issue for good network performance (Jabbari et al., 1995). Zander (1996) defines the problem of Radio Resource Management as follows: for each call we should assign (i) a BS, (ii) a waveform, and (iii) a transmission power, so as to maximize the number of supported calls in the system while maintaining certain predefined Quality of Service criteria. Performing this assignment is the task of the Resource Allocation Algorithm (RAA). In a system based on Frequency Division Multiple Access (FDMA) and/or Time Division Multiple Access (TDMA), such as the GSM system, the selection of radio wave form equates to choosing a frequency and/or a time slot. In either case we term this a radio channel.

In simulation studies the Signal-to-Interference Ratio (SIR) for a given radio channel is often calculated and used by either the RAA to make decisions or used as a measure to evaluate algorithms since it reflects obtainable speech quality. The size of the radius used for the interference calculations in the simulator thus affects the precision of the SIR value. If the algorithm makes use of this value it may in turn influence other measures such as dropping rate, handovers, and system load.

One problem that faces us with this type of investigation is that the particulars of the simulation study will dictate what is considered acceptable fidelity. Furthermore, the actual effects of truncating the interaction radius is likely to vary with the specific RAA under study. This matter is further aggravated by the fact that there are a great number of parameters involved in the model that can potentially interact with the interference calculation depending on how it is used by the RAA. Consequently, it may be difficult to draw general conclusions based on experimental results. On the other hand, the complexity of the model makes the formulation of tractable analytical models difficult. After all, this is why we are interested in simulating it in the first place.

Nevertheless, we deem this an important problem since: (i) the amount of calculation work in each interference measurement is related to the number of stations that must be included, (ii) the size of system necessary to simulate is also related to this, and (iii) it has a direct impact on how an efficient partitioning of such a model can be carried out for parallel or distributed execution. Because it is related to system size it is our belief that some, more or less formalized, investigation of this type is part of most any simulation study of this type. However, since it is usually not considered central to the study these preliminary results are not published. Thus, we are not aware of any previously published results. However, the main contribution of this paper is to take a joint view of both the fidelity and execution time aspects for a sequential and a parallel simulation. Since generalization is difficult we propose here to study some limited, but relevant, cases in some depth. Cases that could hopefully provide an indication of what to expect before more detailed case by case analysis is attempted.

The rest of this paper is organized as follows: Section 2 presents the system model we are considering and experimental results for this model are presented in Section 3. The implications of these results are discussed in Section 4, with conclusions and an outline of future work given in Section 5.

2 SYSTEM MODEL

In this study we consider a model of a cellular network system based on F/TDMA multiple access. It has its origin in the MaDRAS (Mobility and Dynamic Resource Allocation) Simulator (Andersin et al., 1995) developed by RS-lab at the Royal Institute of Technology in Stockholm and WINLAB at Rutgers, the State University of New Jersey. The model and the parallelized implementation (in the MobSim++ simulator) has previously been described in (Liljenstam and Ayani, 1996). The model is designed to be used for the study of various RAAs and includes enough detail to allow the study of algorithms based on measured interference. Typically, RAAs are compared with respect to carried system load under quality-of-service constraints such as call blocking and dropping probabilities.

The model consists of three submodels: the teletraffic model, the mobility model, and the propagation model. The teletraffic model describes the arrival process of new calls and their duration. The mobility model describes the movements of the mobile terminals (MTs). The propagation model describes propagation path loss using a statistical model for correlated log-normal shadow fading.

It is assumed that measurements of the link quality, which we define here as the Signal-to-Interference Ratio (SIR), are carried out periodically in the system as a part of the RAA. The results from these measurements are then used to e.g. check quality-of-service metrics and/or as a basis for decisions by the RAA. For instance, channel assignment and power adjustment decisions could be based on link quality measurements or some part of this information, such as received signal strength.

We start off by considering a system with Fixed Channel Assignment. Cells are grouped into clusters of size K and the available radio channels are divided among the cells within each cluster. Thus, the channel reuse distance in the system can be controlled by an appropriate choice of cluster size. In this case one would expect only limited effect from truncating the interaction radius since the system is already constructed to provide limited (negligible) co-channel interference. Due to variations in signal attenuation caused by obstacles in the terrain, in reality there is still possibility of significant interference. A situation that is typically countered by increased design margins. Some of these effects can be accounted for by including statistical models of shadow fading in the simulations. Consequently, once shadow fading models are incorporated, a stochastic element is introduced.

We assume that interfering signals are independent, the total received interference power is equal to the sum of powers received, and channels are modeled as flat fading channels. Different and orthogonal signal sets are used for up- and down-links. Furthermore, we are assuming orthogonal channels (only co-channel interference is significant) and hence define the downlink SIR γ_m for MT_m for a certain channel k as:

$$\gamma_m = \frac{G_{m,i} \cdot P_i}{\sum_{j \in I} G_{m,j} \cdot P_j + \eta_t}$$

where P_i is the transmitted power from the BS_i (the BS that MT_m is connected to), $G_{m,i}$ is the gain (path loss) from BS_i to MT m (at its current position), S is the set of base stations such that $I=\{j \neq i\ :\ BS_j \text{ is transmitting on channel } k\}$, and η_t is the thermal noise (constant). $G_{m,i}$ and $G_{m,j}$ depend on the terrain, but are regarded as constant for a certain position in the terrain throughout the simulation. The uplink SIR is defined analogously. Furthermore, the link gain, $G_{ij}(d_{ij})$, between transmitter j and receiver i is modeled as

$$G_{ij}(d_{ij}) = \frac{S_{ij}(d_{ij})}{d_{ij}^{\alpha}}$$

where d_{ij} is the distance between the transmitter and the receiver, and where the shadow fading component $S_{ij}(d_{ij})$ is modeled to be log-normally distributed with mean 0 dB and variance σ^2. The exponent α depends on terrain and type of system, and typically falls in the range of 2 to 5. Furthermore, the fading process is assumed to have the exponential correlation function proposed in (Gudmundson, 1993), which is derived from experimental results. Let $U_{ij}(d_{ij})$ = 10*log[$S_{ij}(d_{ij})$], then we have

$$E\{U_{ij}(d_{ij})U_{ij}(d_{ij}+x)\} = \sigma^2 \cdot \exp(-|x|/D)$$

where σ^2 is the log-variance, and the parameter D determines how fast the correlation decreases with distance.

3 EXPERIMENTS

To investigate this case we look first at how much of the total interference is omitted by truncating the interaction radius. This can be calculated analytically under certain assumptions. However, there is no closed form expression for the shadow fading model considered here. Instead, we have based our snapshot analysis on a set of Monte Carlo simulations. After this step, the question remains, however, as to how much impact the truncation will have on the output of an actual simulation. To this end, results from experiments with a time-stepped simulator, RUNE, using two simple RAAs are compared with the snapshot results. RUNE is a simulation environment developed at Ericsson Radio Systems AB. The version used here represents a very simple minimal configuration that was adapted to our assumptions in Section 2. Finally, the execution time impact is considered both for the sequential time-stepped RUNE simulator and for a parallel discrete-event simulator, MobSim++.

Experiments with the RUNE simulator were performed with a fixed system size of 225 cells and with a wrap-around of mobile movement and signal propagation. We identify two cases for determining system size: (i) it is dictated by the size of a real system under investigation, or (ii) the system is idealized and simulation fidelity dictates appropriate system size. The first case was assumed for the execution time experiments presented here. The results on simulation fidelity could serve as a guidance in the second case. Experiments with the MobSim++ simulator serve to indicate what impact the possibility of reducing message communication in a parallel model can have on execution time. Unfortunately, because the gain-map of the simulated region is implemented differently in MobSim++ and the parallel experiments were run on another machine with less memory, the system had to be limited to 144 cells. This will be discussed in more detail further on. Due to the large amount of communication, currently speedups on the shared memory machine are marginal at best for this partitioning approach. However, perhaps more importantly it has the potential of spreading large memory demands over multiple processor through distributed execution. In MobSim++, the gain data for the simulated region is pregenerated and stored in a gain-map requiring a large amount of memory. Using a geographical partitioning, the gain map can be divided over multiple processors.

3.1 System Snapshot

In order to determine how much of the total interference is left out by a limited interaction radius, some Monte Carlo simulations were carried out. In these simulations a static snapshot of the system is taken and the interference within a limited radius compared to a larger area that serves as a reference. Only the downlink is considered, so for each snapshot a mobile is placed somewhere uniformly in the center cell and interference from other base stations calculated. We are assuming a symmetric hexagonal cell layout and omit correlation in the shadow fading model since each snapshot is independent. Thermal noise is disregarded. The probability that a base station is active at any particular time (transmitting on the channel) is q. Interference from base stations in progressively larger circles of interferers were compared to a total of 72 interferers. In this study the first circle consisted of the 6 closest interfering BSs, the second circle of the previous BSs plus another 12 BSs, and the third circle of the previous plus another 18 BSs. The parameters used are given in Table 1 of the Appendix.

Results from these experiments are shown in Figure 1. The fraction of the total interference of the 72 base stations contributed by the first, second, and third circle of interferers respectively are plotted against base station activity (q), i.e. the probability that BS b will be transmitting on channel k at the given time). Each data point represents the average over 10,000 snapshots.

These results indicate that, for these parameters, limiting calculations to the closest interfering stations includes less than 75% of the total interference. Including the second or third circle improves this significantly. The difference between second and third circle is considerably smaller than the difference between first and second circle. Furthermore, decreasing activity will lower the included interference since there are fewer active base stations in the closer circles, thereby reducing their contribution to the total. It should be noted, however, that the total interference will be smaller, so SIR (or C/I) should improve.

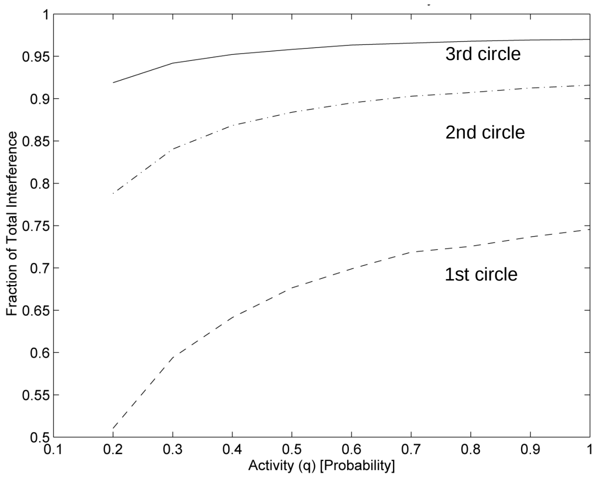

Figure 1: Fraction of Total Interference in the 1st, 2nd, and 3rd Circles

Since there are several parameters of potential significance in the model, a 2^k factor experiment (Law and Kelton, 1991) was carried out to determine the influence of each on the results. The mean response of each factor, i.e. its mean effect on the result (the fraction of total interference) all others factors being equal, is shown in Table 1. A positive response indicates that the error is reduced as the parameter increases from the “low” to the “high” setting.

Obviously, the parameter α is most important since it determines how quickly the signal attenuates with distance. The activity has a significant impact as well as the variance of the shadow fading σ^2. An increased variance increases the likelihood that far-away stations contribute more to the total interference. Cluster size appears to have little effect, however, and cell radius was excluded since it has little effect except for very small values (Zander, 1998).

Table 1. 2^k Factor Investigation of Snapshot (Sensitivity Analysis of Parameters)

Factor	low ‘-’	high ‘+’	1 circle, Mean resp.	2 circles Mean resp.	3 circles Mean resp.
Activity (q)	0.3	0.7	0.1017	0.0600	0.0178
Cluster size (K)	4	12	-0.0048	-0.0017	-0.0006
Prop. att. (α)	3	4	0.1566	0.1256	0.0497
Fading s.d. (σ)	4	6	-0.0472	-0.0328	-0.0106

3.2 Fidelity Impact

The more fundamental question, of course, is how this will affect the actual simulation output, e.g. quality of service metrics. Typically, calls that suffer from inadequate signal quality for an extended period of time are dropped since the link is considered unusable and dropping the call might improve the situation for other users. Thus, there is often a threshold value above which SIR must be kept for all carried calls. One effect of truncating the interference calculations will then be that fewer calls will be dropped against this threshold.

A number of experiments were carried out with the RUNE simulator. The system simulated consisted of 5 by 5 clusters of 9 cells in a rhomboid area. The traffic load on the system was set to 2.3 Erlang per cell with 6 channels per cell giving a blocking probability of approximately 2%. A summary of the simulation parameters are given in Table 2 of the Appendix. Calls were dropped if either the up- or downlink SIR was below the outage level γ_o. Since the timestep size (*dt*) will determine the sampling frequency against the outage threshold it may consequently affect dropping probability. For this reason experiments were carried out with two different settings of *dt*, 2 s and 10 s, respectively. The larger step size causes an “undersampling” of the process (with lower probability of dropping). This effect somewhat resembles an averaging that is likely to be used in real systems.

Figure 2 shows the Cumulative Distribution Function (CDF) for the downlink SIR in one such experiment using *dt*=10 s. The outage SIR threshold γ_o was set to 11 dB corresponding to, according to Figure 2, an outage probability of approximately 0.1%.

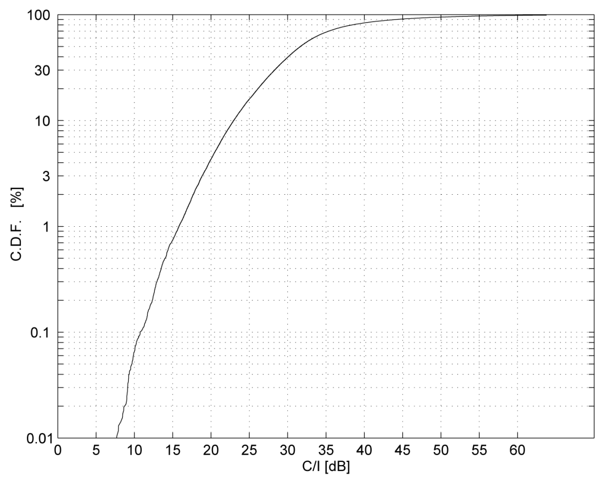

Figure 2: Downlink C/I CDF, γ_o = 11 dB, dt = 10 s

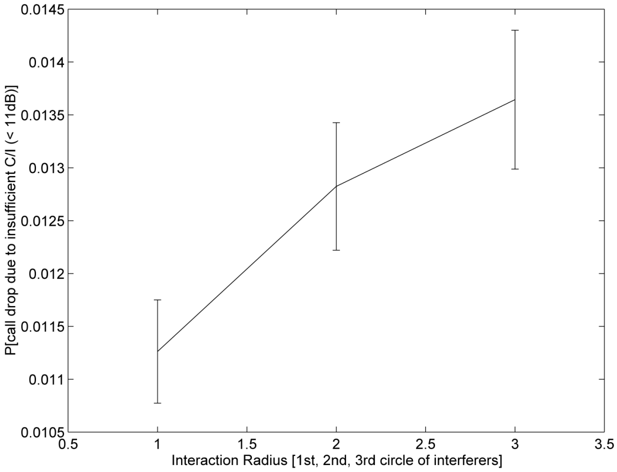

Figure 3: Call Dropping Probability due to Insufficient C/I (<11 dB), dt = 10 s

Results for call dropping probabilities due to insufficient C/I (less than 11 dB) are shown in Figure 3. Each datapoint is the average and 90% confidence interval for 20 experiments. The results indicate that there is a small, but statistically significant, difference in the results when only the closest interferers are included. As indicated by the snapshot experiments, however, the difference between second and third circle is smaller than the difference between first and second circle. And the difference is not statistically significant in these experiments.

In order to relate these results to the snapshot analysis, we need to calculate the activity probability of the BSs. Some elementary queuing theory can be used to obtain an estimate of the activity. If we assume that the mobility in the system can be neglected for this purpose, each cell corresponds to a M/M/m/m m-server loss system (see e.g. Kleinrock, 1975). The system load in this experiments was 2.3 Erlang with 6 channels in each cell. Thus, the average number of occupied channels (in the snapshot situation) is calculated as

$$\bar{N} = \sum_{0}^{m} k p_k$$

where m is the number of channels and the state probability p_k is

$$p_k = \frac{\rho^k / k!}{\sum_{0}^{m} \rho^i / i!}$$

Consequently, in our case $\bar{N} \approx 2.25$. Assuming that channels are picked randomly, the probability that any one channel will be used at any one time (activity probability) is then approximately 0.38. From Figure 1 we see that the first circle only contributes with about 65% of the total interference, on the average. However, the impact this has on simulation output (C/I caused dropping probability, Figure 3) is small in absolute terms, in the order of 0.2%, but with a relative error of about 15%.

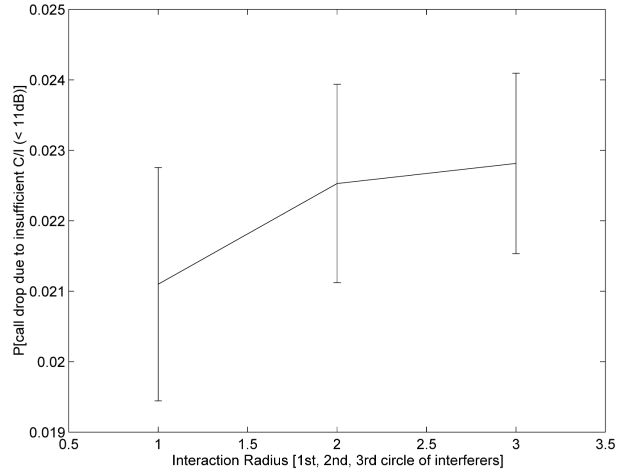

Figure 4: Call Dropping Probability due to Insufficient C/I (<16 dB), dt = 10 s

Increasing the outage threshold γ_o to 16 dB results in dropping probabilities shown in Figure 4. In this case the difference between the first and the second circle relative to the second and third circles are even more pronounced. In both cases, 11 and 16 dB drop threshold, the relative error going from first to second circle is in the order of 10%.

Reducing *dt* to 2 s and simulation runtime to 300 s produced the results in Figure 5. Probability of dropping has increased above 2% but the behavior appears to be similar with a relative error of about 10%. The increased uncertainty in the results due to shorter runlengths prevent statistically valid conclusions.

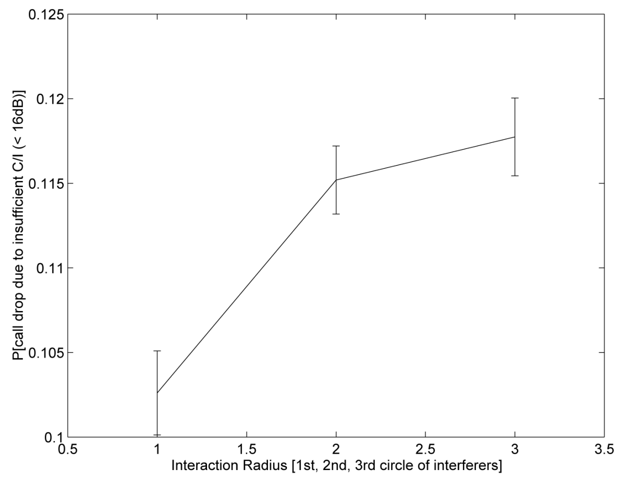

Figure 5: Call Dropping Probability due to Insufficient C/I (<11 dB), dt = 2 s

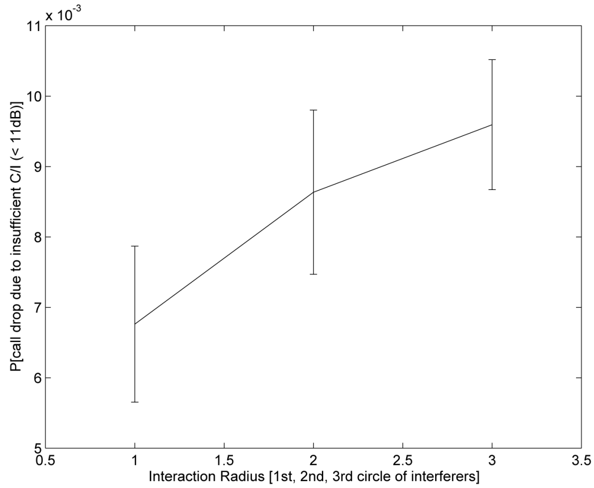

Figure 6: Call Dropping Probability due to Insufficient C/I (<11 dB) w/ Power Control, Drop dt = 10 s

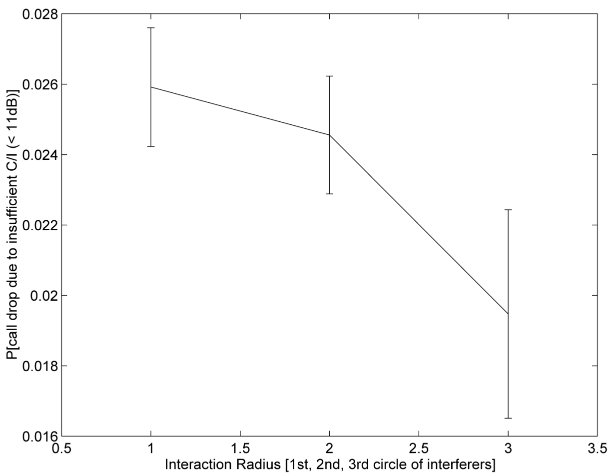

Figure 7: Call Dropping Probability due to Insufficient C/I (<11 dB) w/ Power Control, Drop dt = 2 s

To study what effect changing the RAA might have on the results, a simple power control scheme was implemented. In each step, the transmission power on both links are adjusted according to

$$P^{(n)} = max(P_{min}, min(P_{max}, P^{(n-1)}(\gamma_t/\gamma^{(n-1)})))$$

where $n \geq 1$, $P^{(n-1)}$ is the power used in step *n-1*, $\gamma^{(n-1)}$ is the measured SIR from step *n-1*, and γ_t is a target SIR value. This is a slightly modified version of a power control algorithm used in the MaDRAS simulator (Andersin et al., 1995) where we set $P^{(0)} = P_{max}$ to simplify implementation. Additional parameters are given in Table 3 of the Appendix. Power control and BS/channel allocation was performed with a time step size of 2 s to improve tracking of power adjustments and to limit the number of new calls initiated in each time step, since they enter the system transmitting at full power. However, drop threshold checking was still performed at 10 s intervals.

Results are shown in Figure 6 and indicate a decrease in dropping compared to non-power control. However, the relative error from first to third circles has increased, to 30%, compared to previous experiments. We hypothesize that these effects are due to stronger coupling between stations introduced by the power control mechanism.

Interestingly, using dt = 2 s also for drop threshold checking changes the behavior markedly as seen in Figure 7. Reduced interaction radius now appears to over- rather than under-estimate the dropping probability. A possible explanation is that this effect is caused by slower convergence of the power control mechanism due to the truncated radius.

3.3 Execution Time Impact

In the RUNE simulator it is somewhat difficult to fully exploit the limited interaction radius to reduce execution time due to its implementation. The simulator is implemented in MATLAB (MATLAB, 1992) code and consequently based on vector and matrix operations. This approach, although generally efficient, means that eliminating some transmitters corresponds to sparsening out some of the matrices. Limiting calculations to relevant elements requires extracting these elements, performing the calculation, and then reconstructing the resulting matrix. This is a rather cumbersome operation and time constraints prevented us from fully carrying out the necessary code modifications. Some preliminary experiments were carried out with partial modifications to measure execution time, and the results (using power control, *dt* = 2, and runlength of 300 s) are shown in Figure 8. In this case, truncating interference calculation leads to a decrease in execution time of up to 15%. (Experiments were carried out on a SUN Ultra Enterprise 4000 with 2 Gbyte of memory.)

Another case where we would expect a substantial impact on simulator execution time is when partitioning the simulation for parallel or distributed execution. Depending on partitioning approach chosen, the interaction radius can greatly affect the amount of communication between processors and, thus, the performance. Experiments were carried out with the MobSim++ simulator on a SUN SparcCenter 2000 shared memory machine with 8 processors and 512 Mbyte of memory. The simulator is described in more detail in (Liljenstam and Ayani, 1996) and some previous experiments with different approaches to partitioning are reported in (Liljenstam and Ayani, 1997). Using a model where each BS (or cell) constituted a logical unit (Logical Process) and a number of BSs using the same channel set were mapped to each processor, execution time was measured while varying the interaction radius. In this model, every *T* seconds and independently for each mobile, handover decisions are made and the link quality (SIR) is calculated for up- and downlinks and compared against the γ_o threshold. Since other stations, potentially using the same radio channel, are mapped on other processors every link quality calculation requires message communication proportional to the number of stations involved. Parameters used for the MobSim++ experiments are give in Table 4 of the Appendix.

Results, shown in Table 2 (averaged over three replications), indicate a substantial increase in execution time as more interferers are included. Due to memory constraints from storing a large map of terrain signal gain data, the system was limited to 144 BSs (4 by 4 clusters) and the cell radius was reduced to 500 m. As there is no signal wrap-around in this simulator, the smaller system size accounts for the relatively small increase in execution time between 2nd and 3rd circles of interferers. Most of the interferers present in the system are already included when using the 2nd circle so increasing the interaction radius further only adds a marginal number of interferers. (The difference in execution time between 2nd and 3rd circles is not statistically significant at 90% confidence).

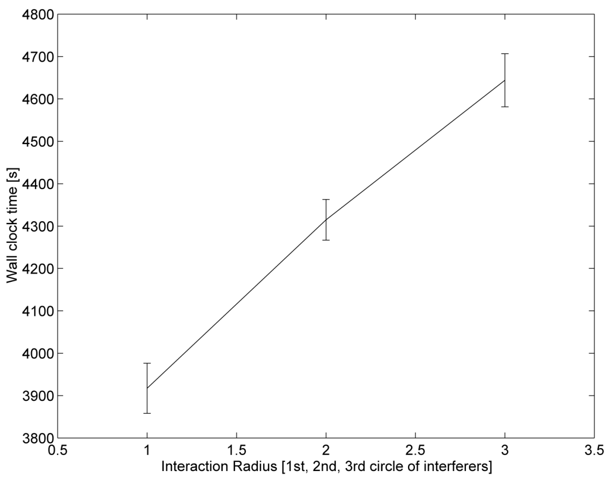

Figure 8: Wall Clock Execution Time for RUNE

Table 2:Wall Clock Execution time for MobSim++

	1st circle	2nd circle	3rd circle
Execution time	780 s	1939 s	2004 s

4 DISCUSSION

Snapshot analysis indicated that for a typical case using an FCA system, as much as 25-45% of the total interference may be lost by limiting link quality calculations to only the closest interfering stations. The actual size of the error will, not surprisingly, mainly depend on how quickly signals fade with distance. But, it will also depend on the variance in shadow fading and the traffic load on the system. However, when more complete simulations were carried out the relative error induced was only approximately 15%. For the case without power control it appears sufficient to measure over the 18 closest interfering cells. Adding another 18 cells provides only insignificant improvement. Once a simple form of power control was incorporated, the relative error increased to some 30%. We hypothesize that power control and DCA algorithms increases the coupling in the system due to the feedback inherent in the algorithms. Thus, for systems using power control and/or DCA it becomes increasingly important to include more of the interfering stations.

For the RUNE simulator, a reduction in execution time of 15% was measured at the price of 15% in relative output error. A 7% reduction in execution time could be achieved at a statistically insignificant error increase. However, a different simulator implementation or more substantial modifications to this simulator should make it possible to better exploit the reduction in work. Experiments with a parallel simulator showed that for parallel execution, once communication becomes an issue, the interaction radius has a more profound impact on execution time. In this case a greater than 50% reduction is achieved if calculations are limited to the six closest interferers.

5 CONCLUSIONS

Snapshot analysis of an FCA system indicates that potentially large errors can be induced if the interaction radius is truncated too closely. In the propagation model used in this study, the error depends on how quickly signals attenuate with distance, the traffic load in the system, and the variance of the shadow fading. The actual error introduced in simulation output also depends on the Resource Allocation Algorithm being used. For a simple case performing little more than link quality control, the relative error in simulations was reduced from the error indicated by snapshot analysis. However, introducing power control increased the simulation output error and we hypothesize that this is due to increased “coupling” in the system caused by the feedback nature of the algorithm.

Preliminary experiments with a time stepped sequential simulator indicate that, for cases when this is deemed acceptable, significant reductions in execution time can be achieved by truncating the interaction radius. Experiments with a parallel simulator indicated that reductions in the amount of message communication by reduced interaction radius can lead to substantial reductions in execution time for parallel and distributed models.

Future work includes investigation of the effects of interaction radius on simulations of DCA systems and extended simulator performance experiments.

ACKNOWLEDGEMENTS

The authors wish to thank Ericsson Radio Systems AB for allowing us to use the RUNE simulation environment. We thank our colleague Dr. Robert Rönngren for reading a draft of this paper and contributing in constructive discussions, and Dr. Bo Hagerman at Ericsson Radio Systems AB for initial help with the snapshot analysis. We also want to thank the Swedish Institute for Computer Science for kindly allowing us to use their SUN SparcCenter 2000 for parallel simulation performance experiments.

APPENDIX: PARAMETER SETTINGS

Table 1: Parameters for Snapshot Experiments.

Parameter	
Cluster size	9
Cell radius	1000 m
Propagation α	4
Propagation σ	6

Table 2: Parameters for RUNE Simulator. (w/o Power Control)

Parameter	
System size	225 cells
Cluster size	9
Channels per cell	6
Cell radius	1000 m
Propagation α	4
Propagation σ	6
Propagation D	110 m
Thermal noise	-118 dBm
Offered load	2.3 Erlang/ cell
Average call length	120 s
Average mobile velocity	14 m/s
Outage SIR (γ_o)	11 dB
Time step (Δt)	10 s
Simulation length	1500 s

Table 3: Parameters for Power Control

Parameter	
Target SIR (γ_t)	30 dB
Max power	1 W
Min power	10 mW
Time step (dt)	2 s
Simulation length	300 s

Table 4:Parameters for MobSim++ Simulator

Parameter	
System size	144 cells
Cluster size	9
Channels per cell	10
Cell radius	500 m
Propagation α	4
Propagation σ	6
Propagation D	100 m
Thermal noise	10^{-15} W
Offered load	2.3 Erlang/ cell
Average call length	120 s
Average mobile velocity	25 m/s
Outage SIR (γ_o)	10 dB
Time between measurements (T)	0.5 s
Simulation length	1500 s

REFERENCES

Andersin, M., M. Frodigh, and K.-E. Sunell. 1995. “Distributed Radio Resource Allocation in Highway Microcellular Systems”, *Fifth WINLAB Workshop on Third Generation Wireless Information Networks*, Rutgers University, New Jersey.

Carothers, C., R. Fujimoto, and Y.-B. Lin. 1995. “A Case Study in Simulating PCS Networks Using Time Warp”, *9th Workshop on Parallel and Distributed Simulation*, Lake Placid, NY.

Greenberg, A., B. Lubachevsky, P. Wright, and D. Nicol. 1994. “Efficient Massively Parallel Simulation of Dynamic Channel Assignment Schemes for Wireless Cellular Communications”, *8th Workshop on Parallel and Distributed Simulation*, Edinburgh, Scotland.

Gudmundson, M. 1993. “Correlation Model for Analyzing Handoff Algorithms”, *IEEE Trans. Veh. Tech.*, vol. VT-42, pp. 351-356.

Harju, J. and M. Salmi. 1993. “Parallel Simulation of the GSM mobile communication network on Minisuper-computer”, *Proceedings of the 1993 Simulation Multi-conference on the High Performance Computing Symposium*, The Society For Computer Simulation, CA

Jabbari, B., G. Colombo, A. Nakajima, and J. Kulkarni. Jan. 1995. “Network Issues for Wireless Communications”, *IEEE Communications Magazine.*

Kleinrock, L. 1975. “Queueing Systems, Volume 1: Theory”, John Wiley and Sons Inc., New York.

Law A. and W. Kelton. 1991. “Simulation Modeling and Analysis, Second Edition”, New York: McGraw-Hill.

Liljenstam, M. and R. Ayani. 1996. “A Model for Parallel Simulation of Mobile Telecommunication Systems”, *in Proceedings of the International Workshop on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, San José, CA.

Liljenstam, M. and R. Ayani. 1997. “Partitioning PCS for Parallel Simulation”, *In Proceedings of the Fifth Workshop on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*, Haifa, Israel.

Liu, W., C.-C. Chiang, H.-K. Wu, V. Jha, M. Gerla, and R. Bagrodia. 1996. “Parallel Simulation Environment for Mobile Wireless Networks”, *In Proceedings of the 1996 Winter Simulation Conference*, San Diego, CA.

Zander, J. 1996. “Radio Resource Management - an Overview”, *In Proceedings of the IEEE Annual Vehicular Technology Conference (VTC’96)*, Atlanta.

Zander, J. 1998. Personal communication with Professor Zander

“MATLAB Reference Guide”, 1992, The MathWorks Inc., 24 Prime Park Way, Natick, Mass.

AUTHOR BIOGRAPHIES

MICHAEL LILJENSTAM is a Ph.D. candidate in the Department of Teleinformatics, Royal Institute of Technology (KTH), Stockholm, Sweden. His research interests include parallel and distributed simulation techniques and simulation of wireless communication systems. He received an M.Sc. in Computer Science from KTH in 1993.

RASSUL AYANI is a visiting Professor at the National University of Singapore, School of Computing while on leave from the Royal Institute of Technology (KTH), Sweden. His current research interests are in parallel architectures, parallel algorithms, and parallel/distributed simulation. He received his first degree from University of Technology in Vienna (Austria), his master degree from University of Stockholm and his Ph.D. from KTH. Dr. Ayani is an Associate Editor of the ACM Transactions on Modeling and Computer Simulation (TOMACS).