

Composite Performance and Availability Analysis of Wireless Communication Networks

Yue Ma, *Member, IEEE*, James J. Han, *Senior Member, IEEE*, and Kishor S. Trivedi, *Fellow, IEEE*

Abstract—With the increasing popularity of wireless communication systems, customers are expecting the same level of service, availability, and performance from the wireless communication networks as the traditional wire-line networks. Traditional pure performance model that ignores failure and recovery but considers resource contention generally overestimates the system's ability to perform a certain job. On the other hand, pure availability analysis tends to be too conservative since performance considerations are not taken into account. To obtain realistic composite performance and availability measures, one should consider performance changes that are associated with failure recovery behavior. In this paper, a brief review is first given over the advances in composite performance and availability analysis. Thereafter, three techniques for composite performance and availability analysis are discussed in detail through a queueing system in a wireless communication network. Numerical results show that an approximate model based on a framework originally proposed by Bobbio and Trivedi (BT approach) provides remarkably accurate predictions on system performance.

Index Terms—Availability, channel allocation, failure recovery, performability, stochastic reward net (SRN), wireless communication systems.

I. INTRODUCTION

WITH the increasing popularity of wireless communication systems, customers are expecting the same level of service, availability and performance from the wireless communication networks as the traditional wire-line networks. Due to the dynamic environment, such as the roaming of the mobile subscribers, maintaining a high radio frequency (RF) availability is one of the most challenging aspects in wireless networks. When a wireless network encounters failures, either due to software, hardware, environment, human error, or a combination of these factors, the network can generally provide its service continuously without interruption. However, the system capacity, that is, the number of active subscribers that the system can support, may decrease. The system performance such as throughput and response time may degrade. Traditional pure performance model that ignores failure and recovery but considers resource contention generally overestimates the system's ability to perform a certain job. On the other hand, pure availability analysis tends to be too conservative since performance

considerations are not taken into account. To obtain realistic composite performance and availability measures, one should consider performance changes that are associated with failure recovery behavior.

Over the last two decades, significant advances have been made in the development of techniques for evaluating the performance, availability, and reliability of computer and communication systems in an integrated way. In the late 1970s, Beaudry [2] developed measures which provide quantitative information about the tradeoffs between reliability and performance of degradable computing systems. Meyer [20] defined the concept of *performability*, where performance and reliability are considered in a unified manner. He also proposed a general framework for model-based performability evaluation. Since then, extensive research activities in performability modeling have been carried out ranging from model construction and solution through tool development and applications.

There are several approaches [23] that have been shown to be useful for composite performance and availability analysis. One approach is to combine the performance and availability models into a single monolithic model. The advantage of this approach is that it yields accurate results. However, this direct approach generally faces two problems, namely, *largeness* and *stiffness*. The largeness problem can be alleviated by using automated generation methods for Markov chains. These automated generation methods address only the model specification and generation issues. To tackle the largeness problem, two basic techniques can be applied: largeness tolerance and largeness avoidance [10]. Stiffness arises when the transition probabilities/rates of the Markov models are of widely varying orders of magnitude. This is clearly true in the performability models where the performance related rates are large and the failure related rates are small. Aggregation techniques [3] and stiffness-tolerance [18] can be applied in dealing with the stiffness problems.

Another widely applied approach in combined performance and availability analysis is the hierarchical modeling technique [19]. There are several advantages in using this approach. First, the largeness problem can be avoided through the "divide and conquer" strategy, where a large system is decomposed into several submodels [8]. Second, the stiffness problem can be resolved by separating the fast and slow rates from each other [4].

In this paper, we will illustrate three techniques for composite performance and availability analysis through evaluating an $M/M/C/C$ queueing system in a wireless communication network. These techniques include: exact composite performance and availability approach [23], pure performability approach [21] and the BT approach (proposed by Bobbio and Trivedi [4]).

Manuscript received September 6, 2000.

Y. Ma and J. J. Han are with Global Software Group, Motorola, Inc., Elk Grove Village, IL 60007 USA (e-mail: yue.ma@motorola.com; cjh048@email.mot.com).

K. S. Trivedi is with Center for Advanced Computing and Communication (CACC), Department of Electrical and Computer Engineering, Duke University, Durham, NC 27708-0291 USA (e-mail: kst@ee.duke.edu).

Publisher Item Identifier S 0018-9545(01)08524-3.

In Section II, we give a brief description of a simplified channel allocation scheme in a wireless network. In Section III, three techniques for composite performance and availability analysis are discussed in detail. Numerical results are presented in Section IV. We make our conclusions in Section V. In the Appendix, a brief introduction of stochastic reward net (SRN) is given.

II. A SIMPLE PURE PERFORMANCE MODEL FOR CHANNEL ALLOCATION

To illustrate different techniques for composite performance and availability analysis, we use a channel allocation scheme adapted from [15]. When a new call (NC) is attempted in a cell covered by a base station (BS), the NC is connected if an idle channel is available in the cell. Otherwise, the call is blocked. When a mobile station (MS) with an ongoing call travels across the cell boundaries, the channel in the old serving cell is released, and an idle channel is required in the target cell, which would be the new serving cell. This process is called *handoff*. If an idle channel exists in the target cell, the handoff call (HC) continues nearly transparently to the user. Otherwise, the HC is dropped.

The dropping of a handoff call (HC) is considered more severe than the blocking of a new call (NC). One method [22] to reduce the dropping probability of HCs is to reserve a fixed or an adaptive (natural or fractional) number of channels exclusively for HCs. These exclusively reserved channels are referred to as *guard channels*. For example, if the total number of RF channels is C and the number of guard channels is g , then the number of RF channels available for both NCs and HCs is $C - g$. It should be noted that no specific *channels* are reserved as guard channels but only a specific *number* of channels are reserved.

For comparison purposes, we first present a pure performance model under the assumption that the channels in a wireless network never fail. To make the model easier to understand, we first present the model in form of an SRN (Fig. 1). Its corresponding continuous time Markov chain (CTMC) is shown in Fig. 2. Compared with the traditional CTMC, which is generally quite abstract from the system being modeled, a high-level description language such as SRN can specify a real-world system in a compact and intuitive way. Since the early 1990s, SRN has been used as a powerful modeling tool for performance, availability, and reliability analysis in communication networks [11]–[15], [24]. An introduction of SRN is given in the Appendix.

In Fig. 1, place CP is the channel pool for the cell. Initially, there are C idle channels which are accessible for both the NCs and the HCs. Transitions t_n and t_h^i represent the arrivals of NCs and HCs, respectively. Transition t_h^i is enabled with at least one idle channel in place CP . Otherwise, it is blocked. Transition t_n is disabled if there are less than $g + 1$ channels in place CP . This is represented by the multiple input arc from place CP to transition t_n and the multiple output arc from transition t_n to place CP . The resulting effect is that when transition t_n fires, only one token is moved from place CP to place T . The number of tokens in place T is the number of channels currently being utilized in the cell. Transitions t_d and t_h^o respectively represent the departure of a call, either due to the termination of the call or due to

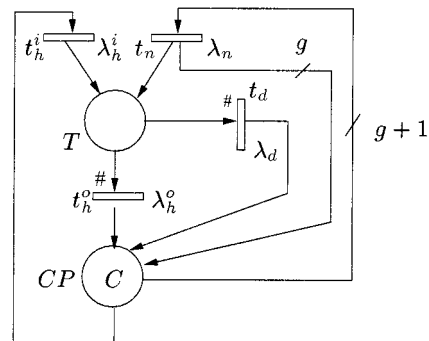


Fig. 1. SRN of a performance model for channel allocation.

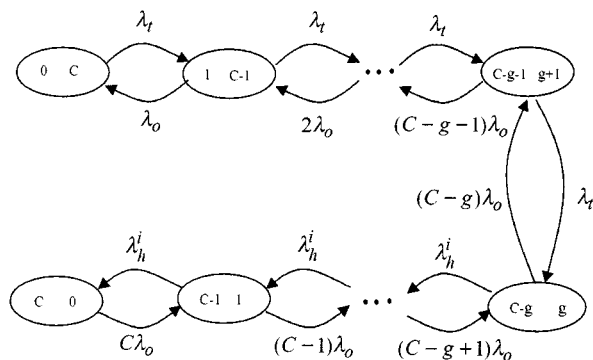


Fig. 2. CTMC for the SRN model in Fig. 1.

the MS leaving the cell. The clearing rate for a *single* call is λ_d . The handoff departure rate is λ_h^o . Notice that transitions t_d and t_h^o have marking dependent firing rates. The *actual* firing rates for transitions t_d and t_h^o are $k\lambda_d$ and $k\lambda_h^o$, respectively, where k is the number of tokens in place T . The marking dependency is indicated by the # signs next to the transitions in Fig. 1.

Let T_n denote the number of tokens in place T and, consequently, let $m = \{T_n, CP_n\}$ denote the marking of the SRN in Fig. 1. The CTMC for the SRN of the performance model is shown in Fig. 2, where $\lambda_t = \lambda_n + \lambda_h^i$ and $\lambda_o = \lambda_d + \lambda_h^o$. With the underlying infinitesimal generator \mathbf{Q} for the CTMC, numerical solution methods can be applied to get the desired different performance measures.

III. COMPOSITE PERFORMANCE AND AVAILABILITY ANALYSIS: EXACT AND APPROXIMATE APPROACHES

In this section, we will illustrate three techniques for combined performance and availability analysis through evaluating an $M/M/C/C$ queueing system with failure, repair, but no recovery of the channels. These techniques include: exact composite performance and availability approach [23], pure performability approach [21] and BT approach (proposed by Bobbio and Trivedi [4]). For the queueing system, we assume the call arrival rates for the new and handoff calls are λ_n and λ_h^i , respectively. The channel can fail with rate λ_f and is repaired with rate μ_r . A single repair facility is assumed. We also assume that the channel can be found in failure status *only* while it is being utilized. In other words, a channel is assumed

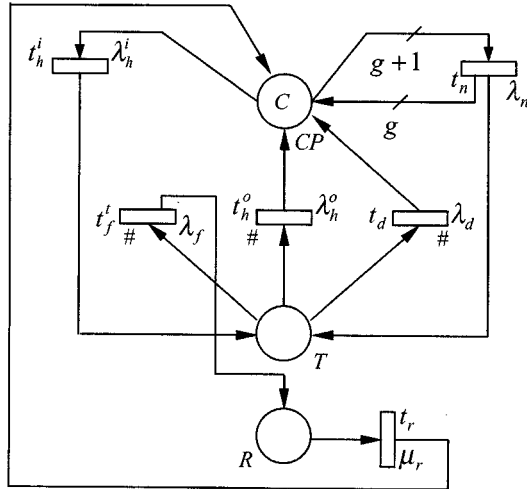


Fig. 3. Exact SRN model of the $M/M/C/C$ queueing system with channel failure and repair.

not to fail while it is idle. The failure and recovery of idle channels are discussed in [15], [16].

A. Exact Composite Performance and Availability Model

First, we consider the exact composite model. The monolithic SRN model is shown in Fig. 3. Compared with Fig. 1, Fig. 3 has one more place (place R) and two more transitions (transitions t_f^t and t_r). Place R represents the place where the channels are being repaired or waiting to be repaired. Transition t_f^t represents the failure of a channel while transition t_r represents the repair of a channel. The corresponding CTMC model of Fig. 3 is shown in Fig. 4, where state (u, v) represents that there are u talking channels and v channels are in failure status. In Fig. 4, $\lambda_t = \lambda_n + \lambda_h^i$, $\lambda_o = \lambda_d + \lambda_h^o$, $a = C - g$, $b = C - g + 1$ and $q = C - 1$.

We denote the dropping and the blocking probabilities for the composite approach as P_d^c and P_b^c , respectively. To calculate P_d^c , the reward rate assignment is

$$(r_d^c)_j = \begin{cases} 1, & \text{if } [\#(CP_j)] = 0, \\ 0, & \text{otherwise.} \end{cases}$$

The reward rate for state j in the CTMC of SRN is denoted by $(r_d^c)_j$, and $\#(CP_j)$ represents the number of channels in place CP in marking (state) j . Thus, a reward rate of one is assigned to the states where the channel pool is empty, and a reward rate of zero is assigned to the other states. Then P_d^c is calculated by

$$P_d^c = \sum_{j \in \Omega} (r_d^c)_j \pi_j + P_R \quad (1)$$

where

- Ω set of all tangible markings;
- π_j steady-state probability of marking j ;
- P_R sum of weighted state occupancy probabilities when the number of channels in place R is nonzero.

The reward rate assignment for P_R is

$$r_R^j = \begin{cases} \#(R_j), & \text{if } \#(R_j) > 0, \\ 0, & \text{otherwise.} \end{cases}$$

In summary, the first term of (1) represents the dropping probability for incoming handoff calls when the channel pool is empty; the second term of (1) represents the dropping probability for ongoing calls when the channels they use are down.

The blocking probability P_b^c is calculated through

$$P_b^c = \sum_{j \in \Omega} (r_b^c)_j \pi_j$$

where

$$(r_b^c)_j = \begin{cases} 1, & \text{if } [\#(CP_j)] \leq g, \\ 0, & \text{otherwise.} \end{cases}$$

That is, a reward rate of one should be assigned to the states where the channel pool has less than $g + 1$ channels.

In a realistic environment, an exact approach generally encounters the largeness and the stiffness problems. A hierarchical approach is generally an ideal methodology for avoiding the above problems. In the next two sections, we will model the same channel allocation scheme through two hierarchical approaches.

B. Pure Performability Model

Now, we compute the dropping and blocking probabilities using the pure performability approach [21], which is a two-level Markov reward model. Such a two-level model is an approximation since we assume that in each state of the upper level model, the lower level model reaches steady state. The upper level model, as shown in Fig. 5, describes the failure and repair behavior of the $M/M/C/C$ system. Each state in Fig. 5 represents the number of nonfailed (either talking or idle) channels. The upper level model is a pure availability model. Notice that the transition rate from state i to state $i - 1$ ($i \in [1, C]$) is $i\lambda_f$. The index i includes both idle and talking channels. Therefore, in this example, the pure performability model incorporates an approximation of the exact composite model. The lower level model, as shown in Fig. 6, captures the pure performance aspect of the system. Each state represents the number of talking channels in the system. In Fig. 6, $N \in [1, C]$. In other words, for the upper level model in Fig. 5, there are C corresponding lower level pure performance models as depicted in Fig. 6. To get the numerical measures of the whole system, the lower level performance model is solved and its results are passed as reward rates to the upper level availability model.

We denote, respectively, P_d^p and P_b^p as the dropping and blocking probability in the pure performability model. The approximate dropping probability is obtained through

$$P_d^p = P_0^u + \sum_{i=1}^C (P_i^u \cdot P_{di}^l) + \sum_{i=0}^{C-1} (C - i) P_i^u \quad (2)$$

where P_i^u ($i \in [0, C]$) is the steady-state probability of the system being in state i of the upper level model and P_{di}^l is the dropping probability in the lower level model when $N = i$.

The approximate blocking probability is obtained through

$$P_b^p = \sum_{i=0}^g P_i^u + \sum_{i=g+1}^C (P_i^u \cdot P_{bi}^l) \quad (3)$$

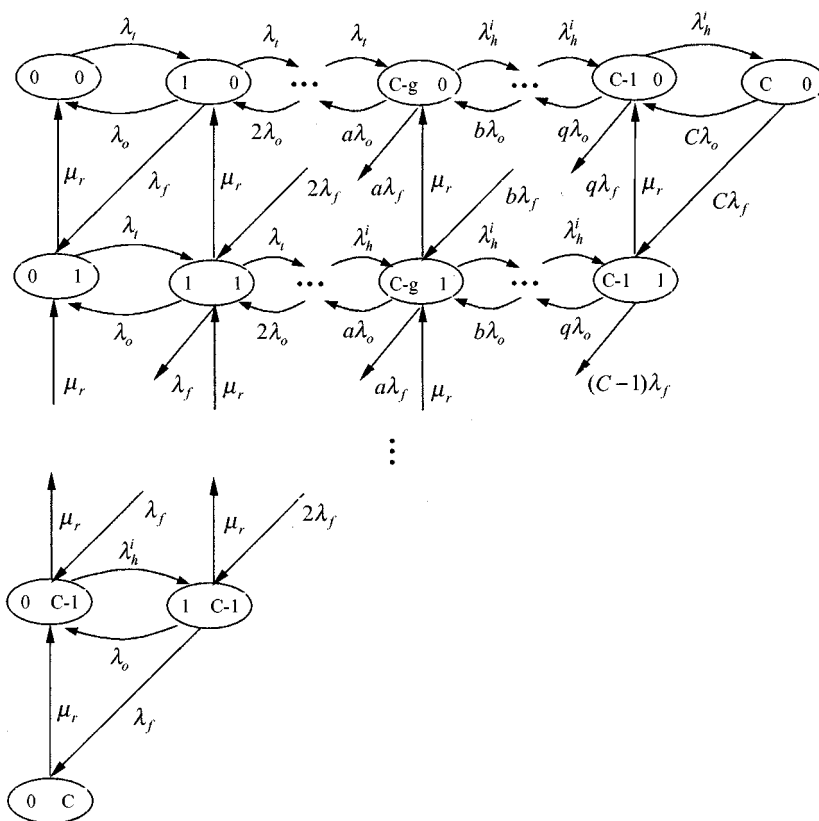


Fig. 4. The exact CTMC model of the $M/M/C/C$ queueing system with channel failure and repair.

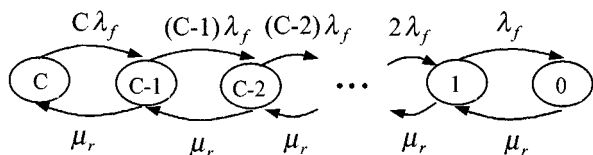


Fig. 5. Upper level availability model for the $M/M/C/C$ system.

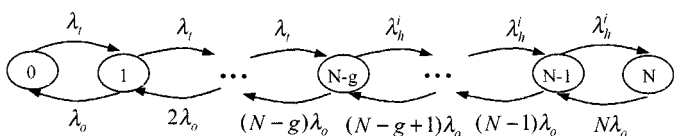


Fig. 6. Lower level performance model for the $M/M/C/C$ system.

where P_{bi}^l is the blocking probability in the lower level model when $N = i$. The steady-state probability P_i^u in (2) and (3) is calculated from

$$P_i^u = P_C^u \rho^{(C-i)} \frac{C!}{i!}$$

where

$$P_C^u = \frac{1}{\sum_{j=0}^C \rho^j \frac{C!}{(C-j)!}} \quad \text{and} \quad \rho = \frac{\lambda_f}{\mu_r}.$$

The steady-state probability P_k^l ($k \in [0, N]$) for the lower level model is obtained from

$$P_k^l = \begin{cases} P_0^l \cdot \left(\frac{\lambda_t}{\lambda_o}\right)^k \cdot \frac{1}{k!} & k \leq N - g \\ P_0^l \cdot \left(\frac{\lambda_t}{\lambda_o}\right)^{N-g} \cdot \frac{1}{k!} \cdot \left(\frac{\lambda_h^i}{\lambda_o}\right)^{k-(N-g)} & k > N - g \end{cases} \quad (4)$$

where you have the equation shown at the bottom of the page. Consequently, the dropping probability P_{di}^l in the lower level model is derived as

$$P_{di}^l = P_N^l = P_0^l \cdot \left(\frac{\lambda_t}{\lambda_o}\right)^{N-g} \cdot \frac{1}{N!} \cdot \left(\frac{\lambda_h^i}{\lambda_o}\right)^g. \quad (5)$$

$$P_0^l = \frac{1}{\sum_{k=0}^{N-g} \left(\frac{\lambda_t}{\lambda_o}\right)^k \frac{1}{k!} + \sum_{k=N-g+1}^N \left(\frac{\lambda_t}{\lambda_o}\right)^{N-g} \frac{1}{k!} \left(\frac{\lambda_h^i}{\lambda_o}\right)^{k-(N-g)}}.$$

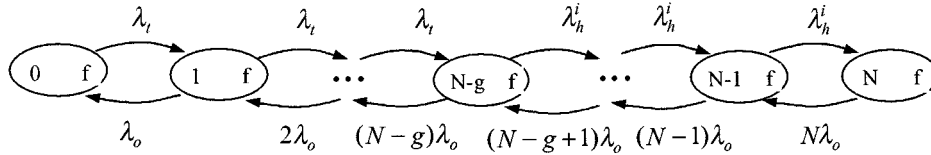


Fig. 7. CTMC for a generic fast recurrent subset of Fig. 4.

The expression for the blocking probability P_{bi}^l of each lower level model is

$$P_{bi}^l = \sum_{k=N-g}^N P_k^l = P_0^l \cdot \sum_{k=N-g}^N \left(\frac{\lambda_t}{\lambda_o}\right)^{N-g} \frac{1}{k!} \left(\frac{\lambda_h^i}{\lambda_o}\right)^{k-(N-g)}. \quad (6)$$

C. BT Method

Finally, we calculate the probabilities for the dropped and blocked calls using the one-step aggregation technique originally proposed by the BT method [4]. This approximate algorithm first proceeds by separating the state transition rates into fast rates and slow rates. The fast rates are generally of several orders of magnitude larger than the slow rates. The state-space of CTMC is consequently divided into *fast* and *slow* states. Fast states are the ones with at least one fast outgoing transition. Slow states do not have any fast outgoing transition. The fast states can be further divided into several *recurrent* subsets and at most one *transient* subset. States in the fast recurrent subset are connected with each other via fast transitions, but are connected to any outside state only by slow transitions. The fast transient subset is connected to an outside state by means of at least one fast transition.

The key idea of the BT method is to separate the state space of a CTMC into fast recurrent subsets and/or a fast transient subset. Each recurrent subset is analyzed independently and is replaced by a slow state. The transient subset is analyzed to obtain conditional branching probabilities and is replaced with these probabilities. The resulting Markov chain is small and nonstiff. Conventional numerical techniques can be used to analyze this Markov chain.

Now consider the $M/M/C/C$ queueing system as shown in Fig. 4. We assume that the rates λ_t, λ_h^i and λ_o are fast, and μ_r and λ_f are slow. From the classification presented earlier in this section, state $(0, C)$ is the only slow state. The first C rows of states in Fig. 4 form, respectively, C fast recurrent subsets. Each subset can be modeled by a CTMC as shown in Fig. 7. The CTMCs in Figs. 6 and 7 are actually the same. Only the notations for the states are different. An approximate Markov chain (Fig. 8) is obtained by aggregating the recurrent subsets into slow states. In Fig. 8, the transition parameters β_i ($i \in [1, C]$) represent the expected number of talking channels in the fast recurrent subset and is given by

$$\beta_i = \sum_{k=1}^i k P_k^l$$

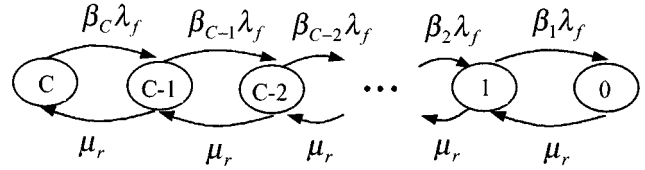


Fig. 8. Aggregated Markov chain for the exact model of Fig. 4.

where P_k^l is the steady-state probability for the CTMC in Fig. 7 and is obtained through (4).

We denote P_d^{BT} and P_b^{BT} as the dropping and the blocking probability for the BT method, respectively. The dropping probability P_d^{BT} is given through

$$P_d^{\text{BT}} = P_0^A + \sum_{i=1}^C P_i^A P_{di}^{\text{frs}} + \sum_{i=0}^{C-1} (C-i) P_i^A \quad (7)$$

where P_i^A ($i \in [0, C]$) is the steady-state probability of the system being in state i of the aggregated Markov chain and P_{di}^{frs} is the dropping probability in the fast recurrent subset when $N = i$.

The blocking probability P_b^{BT} is given by

$$P_b^{\text{BT}} = \sum_{i=0}^g P_i^A + \sum_{i=g+1}^C P_i^A P_{bi}^{\text{frs}} \quad (8)$$

where P_{bi}^{frs} is the blocking probability in the fast recurrent subset when $N = i$. The steady-state probability P_i^A in (7) and (8) is given by

$$P_i^A = P_C^A \rho^{(C-i)} \prod_{k=i+1}^C \beta_k, \quad i < C$$

where

$$P_C^A = \frac{1}{1 + \sum_{i=0}^{C-1} \rho^{(C-i)} \prod_{k=i+1}^C \beta_k} \quad \text{and} \quad \rho = \frac{\lambda_f}{\mu_r}.$$

The dropping probability P_{di}^{frs} and the blocking probability P_{bi}^{frs} are obtained from (5) and (6), respectively.

The main difference between the pure performability approach and the BT method is the transition rates among the states in Figs. 5 and 8. Our original assumption is that a channel can fail *only* when it is being used. Through β_i , the BT method can reflect this assumption in a realistic way. For the performability approach, the upper level model (Fig. 5) is a pure availability model. It does not have the flexibility of separating the busy channels from the idle channels. As

TABLE I
 DROPPING PROBABILITIES FOR THE THREE APPROACHES

Traffic Load in Erlangs	Exact Composite Model P_d^c	Pure Performability Model P_d^p	δ_d^{p-c} $\times 10^{-2}$	BT Method P_d^{BT}	δ_d^{BT-c} $\times 10^{-5}$
8.4	5.279474e-05	1.625362e-04	207.8644	5.279475e-05	0.025588
9.6	6.122534e-05	1.634718e-04	167.0002	6.122536e-05	0.035546
10.8	7.431347e-05	1.690708e-04	127.5103	7.431352e-05	0.071830
12	1.057648e-04	1.930555e-04	82.5328	1.057649e-04	0.047759
13.2	1.912726e-04	2.711611e-04	41.7668	1.912729e-04	0.164226
14.4	4.005740e-04	4.732054e-04	18.1318	4.005751e-04	0.263740
15.6	8.371673e-04	9.028062e-04	7.8406	8.371695e-04	0.268599
16.8	1.620023e-03	1.679057e-03	3.6440	1.620025e-03	0.100704
18	3.137094e-03	3.189367e-03	1.6663	3.137098e-03	0.135955
19.2	4.896573e-03	4.943552e-03	0.9594	4.896577e-03	0.084778
20.4	7.049857e-03	7.092208e-03	0.6007	7.049863e-03	0.086668
21.6	9.049374e-03	9.088051e-03	0.4274	9.049379e-03	0.058167

 TABLE II
 BLOCKING PROBABILITIES FOR THE THREE APPROACHES

Traffic Load in Erlangs	Exact Composite Model P_b^c	Pure Performability Model P_b^p	δ_b^{p-c} $\times 10^{-3}$	BT Method P_b^{BT}	δ_b^{BT-c} $\times 10^{-5}$
8.4	2.093913e-06	2.094373e-06	0.219783	2.093920e-06	0.349242
9.6	1.766852e-05	1.767150e-05	0.168913	1.766861e-05	0.518583
10.8	1.008228e-04	1.008357e-04	0.127931	1.008232e-04	0.349376
12	4.223116e-04	4.223526e-04	0.097049	4.223127e-04	0.257460
13.2	1.376511e-03	1.376612e-03	0.073671	1.376514e-03	0.236296
14.4	3.646488e-03	3.646692e-03	0.055982	3.646498e-03	0.263852
15.6	8.124197e-03	8.124540e-03	0.042160	8.124217e-03	0.252060
16.8	1.568577e-02	1.568626e-02	0.030978	1.568580e-02	0.169981
18	2.855854e-02	2.855917e-02	0.022170	2.855858e-02	0.155143
19.2	4.355237e-02	4.355309e-02	0.016637	4.355242e-02	0.117703
20.4	6.145684e-02	6.145764e-02	0.012984	6.145691e-02	0.113732
21.6	7.954190e-02	7.954273e-02	0.010489	7.954197e-02	0.094169

a result, it can only model the situation where *both* idle and working channels can fail.

IV. NUMERICAL RESULTS

For the purpose of discussion, we assume that a set of $C = 26$ channels are assigned to each cell. The average travel time to cross from one cell to another ($1/\lambda_r^c$) is 6 min. The expected call holding time ($1/\lambda_d$) is 1.2 min. The average failure rate ($1/\lambda_f$) for each channel is once every 80 000 h. The expected repair time ($1/\mu_r$) is 30 min. The handoff arrival rate is obtained through fixed-point iteration [9], [15], [17].

The corresponding dropping and blocking probabilities are shown in Tables I and II. The relative errors of the pure performability approach and the BT approach are defined as follows: $\delta_d^{p-c} = (P_d^p - P_d^c)/P_d^c$, $\delta_d^{BT-c} = (P_d^{BT} - P_d^c)/P_d^c$, $\delta_b^{p-c} = (P_b^p - P_b^c)/P_b^c$ and $\delta_b^{BT-c} = (P_b^{BT} - P_b^c)/P_b^c$. As the traffic load increases, the dropping and blocking probabilities obtained from the pure performability model get closer to the exact values. This is as expected since when the traffic increases, the number of talking channels increases and the error

caused by assuming the idle channel failures decreases. From the numerical results, it is shown that the BT method gives a better approximation over the pure performability approach. Because in the aggregated Markov chain, the channel failure rate depends on the expected number of talking channels. This reflects the original assumption (only the talking channels can fail) in a realistic way.

V. CONCLUSION

During the last decade we have witnessed a tremendous growth within the wireless communication industry. Customers want speed and improved cost effective performance, but only if it comes with reliable services. This requires fundamental rethinking of the traditional pure performance model that ignores failure and recovery but mainly concentrates on resource contention. To reflect a real-world system in a realistic way, availability, capacity, and performance issues of a network should be considered in an integrated way.

In this paper, we illustrate three modeling approaches for composite performance and availability analysis. The three

modeling techniques include an exact composite model and two approximate models, pure performability model, and the BT model. For comparison purposes, a pure performance model is also presented. A high-level description language, stochastic reward net, as well as the continuous time Markov chain, are used to construct models for evaluating the performability measures of a channel allocation scheme in a wireless network. Numerical results show that an approximate model based on BT approach yields very accurate predictions on system performance.

APPENDIX INTRODUCTION TO SRN

SRN[6] is an extension of Petri net (PN) [1], [5], which is a high-level description language for formally specifying complex systems. A PN is a bipartite directed graph with two types of nodes: *places* and *transitions*. Each place may contain an arbitrary (natural) number of *tokens*. For a graphical presentation, places are depicted as circles, transitions are represented by bars and tokens are represented by dots or integers in the places. Each transition may have zero or more *input arcs*, coming from its input places; and zero or more *output arcs*, going to its output places. A transition is *enabled* if all of its input places have at least as many tokens as required by the multiplicities of the corresponding input arcs. When enabled, a transition can *fire* and will remove from each input place and add to each output place the number of tokens corresponding to the multiplicities of the input/output arcs. A *marking* depicts the *state* of a PN which is characterized by the assignment of tokens in all the places.

Generalized stochastic Petri nets (GSPNs) [1] extend the PNs by assigning a *firing time* to each transition. Transitions with exponentially distributed firing times are called *timed* transitions while the transitions with zero firing times are called *immediate* transitions. A marking in a GSPN is called *vanishing* if at least one immediate transition is enabled; otherwise it is called a *tangible* marking. Under the condition that only a finite number of transitions can fire in finite time with nonzero probability, it can be shown that a given GSPN can be reduced to a homogeneous continuous time Markov chain (CTMC) [1].

In order to make more compact models of complex systems, several extensions are made to GSPN, leading to the SRN. One of the most important features of SRN is its ability to allow extensive marking dependency. In an SRN, each tangible marking can be assigned one or more *reward rate(s)*. Parameters such as the firing rate of the timed transitions, the multiplicities of input-output arcs and the reward rate in a marking can be specified as functions of the number of tokens in any place in the SRN. For an SRN, all the output measures are expressed in terms of the expected values of the reward rate functions. To get the performance and reliability/availability measures of a system, appropriate reward rates are assigned to its SRN. In this paper, we use the tool stochastic Petri net package (SPNP) [7] to specify and solve the SRN models.

REFERENCES

- [1] M. Ajmone-Marsan, D. Kartson, G. Conte, and S. Donatelli, *Modeling with Generalized Stochastic Petri Nets*. New York: Wiley, 1995.
- [2] M. D. Beaudry, "Performance-related reliability for computer systems," *IEEE Trans. Comput.*, vol. C-27, pp. 540-547, June 1978.
- [3] A. Bobbio and K. S. Trivedi, "Computing cumulative measures of stiff Markov chains using aggregation," *IEEE Trans. Comput.*, vol. 39, pp. 1291-1298, Oct. 1990.
- [4] —, "An aggregation technique for the transient analysis of stiff Markov chains," *IEEE Trans. Comput.*, vol. C-35, pp. 803-814, Sept. 1986.
- [5] G. Bolch, S. Greiner, H. de Meer, and K. S. Trivedi, *Queueing Networks and Markov Chains, Modeling and Performance Evaluation with Computer Science Application*. New York: Wiley, 1998.
- [6] G. Ciardo, A. Blakemore, P. F. Chimento Jr., J. K. Muppala, and K. S. Trivedi, "Automated generation and analysis of Markov reward models using stochastic reward nets," in *Linear Algebra, Markov Chains and Queueing Models*, C. Meyer and R. Plemmons, Eds. New York: Springer-Verlag, 1993, vol. 48, pp. 145-191.
- [7] G. Ciardo, J. K. Muppala, and K. S. Trivedi, "SPNP Users Manual, Ver. 5.01," Duke Univ., Durham, NC, Tech. Rep., 1998.
- [8] G. Ciardo and K. S. Trivedi, "A decomposition approach for stochastic reward net models," *Performance Evaluation*, vol. 18, no. 1, pp. 37-59, July 1993.
- [9] G. Haring, R. Marie, R. Puigjaner, and K. S. Trivedi, "Loss formulae and their optimization for cellular networks," *IEEE Trans. Veh. Technol.*, vol. 50, pp. 664-673, May 2001.
- [10] B. R. Haverkort, "Approximate performability and dependability modeling using generalized stochastic Petri nets," *Performance Evaluation*, vol. 18, pp. 61-78, 1993.
- [11] S. Hunter, T. Phillip, and K. S. Trivedi, "Combined performance and availability analysis of a switched network application," in *Proc. Int. Conf. Communications (ICC'97)*, Montréal, Québec, Canada, June 1997.
- [12] O. C. Ibe, H. Choi, and K. S. Trivedi, "Performance evaluation of client-server systems," *IEEE Trans. Parallel Distributed Syst.*, vol. 4, pp. 1217-1229, Nov. 1993.
- [13] F. J. Jaimes-Romero, D. Muñoz-Rodríguez, C. Molina, and H. Tawfik, "Modeling resource management in cellular systems using Petri nets," *IEEE Trans. Veh. Technol.*, vol. 46, pp. 298-312, May 1997.
- [14] Y. Ma, J. J. Han, and K. S. Trivedi, "Call admission control for reducing dropped calls in code division multiple access (CDMA) cellular systems," in *Proc. IEEE INFOCOM 2000*, Tel-Aviv, Israel, Mar. 2000, pp. 1481-1490.
- [15] —, "Channel allocation with recovery strategy in wireless networks," *Eur. Trans. Telecommun. (ETT)*, vol. 11, no. 4, pp. 395-406, July-Aug. 2000.
- [16] —, "A method for multiple channel recovery in TDMA wireless communications systems," *Comput. Commun.*, vol. 24, no. 12, pp. 1147-1157, July 2001.
- [17] V. Mainkar and K. S. Trivedi, "Sufficient conditions for existence of a fixed point stochastic reward net-based iterative models," *IEEE Trans. Software Eng.*, vol. 22, pp. 640-653, Sept. 1996.
- [18] M. Malhotra, J. K. Muppala, and K. S. Trivedi, "Stiffness-tolerant methods for transient analysis of stiff Markov chains," *Microelectron. Reliability*, vol. 34, no. 11, pp. 1825-1841, 1994.
- [19] M. Malhotra and K. S. Trivedi, "A methodology for formal specification of hierarchy in model solution," in *Proc. 5th Int. Workshop Petri Nets and Performance Models (PNPM93)*, Toulouse, France, Oct. 1993.
- [20] J. F. Meyer, "On evaluating the performability of degradable computing systems," *IEEE Trans. Comput.*, vol. C-29, pp. 720-731, Aug. 1980.
- [21] —, "Performability: A retrospective and some pointers to the future," *Performance Evaluation*, vol. 14, no. 3-4, pp. 157-196, Feb. 1992.
- [22] N. D. Tripathi, J. H. Reed, and H. F. Vanlandingham, "Handoff in cellular systems," *IEEE Personal Commun.*, vol. 5, pp. 26-37, Dec. 1998.
- [23] K. S. Trivedi, J. K. Muppala, S. P. Wooleet, and B. R. Haverkort, "Composite performance and dependability analysis," *Performance Evaluation*, vol. 14, no. 3-4, pp. 197-215, Feb. 1992.
- [24] C.-Y. Wang, D. Logothetis, and K. S. Trivedi, "Transient behavior of atm networks under overloads," *Proc. IEEE INFOCOM 96*, pp. 978-985, Mar. 1996.

Yue Ma (S'97-M'99) received the Cand.Mag. and Cand.Scient. degrees, both in informatics, from University of Oslo, Norway, in 1993 and 1995, respectively, and the M.S. and Ph.D. degrees in computer science from Duke University, Durham, NC, in 1998 and 1999, respectively.

He is currently a Lead Software Engineer at Motorola. His research interests are in architecture, availability, performance, quality of service, and reliability analysis of wireless and broad-band access networks.

James J. Han (S'83–M'86–SM'91) received the undergraduate degree in electrical engineering from Chiao Tung University, China, the graduate degree in electrical engineering from the Graduate School of Academia Sinica, China, and the M.S. (applied mathematics) and Ph.D. (electrical engineering) degrees from The Ohio State University, Columbus, OH, in 1985, respectively.

He is a Project Manager of the Global Software Group, Motorola, Elk Grove Village, IL. After teaching electrical and computer engineering in Southern Illinois University, Edwardsville, and the Illinois Institute of Technology, Chicago, as a Professor for six years, he joined AT&T Bell Laboratories, Columbus, OH, and worked on communication system architectures, system performance, and software development. He has worked on research and development projects in wireless communication systems since he joined Motorola. He now manages projects on high availability wireless communication systems, including GSM, iDEN, CDMA, GPRS, and Internet and IP-based wireless communication systems. He also manages high-availability projects in broad-band and wide-band wireless and wire-line communication systems such as wireless G3 and G4 systems and cable communication systems. His research interests are in the next generation wireless communication systems and high availability communication systems, including hardware, software, and the overall systems.

Kishor S. Trivedi (M'86–SM'87–F'92) received the B.Tech. degree from the Indian Institute of Technology, Bombay, and the M.S. and Ph.D. degrees in computer science from the University of Illinois, Urbana-Champaign, in 1968, 1972, and 1974, respectively.

He holds the Hudson Chair in the Department of Electrical and Computer Engineering, Duke University, Durham, NC. He also holds a joint appointment in the Department of Computer Science at the same university. He is the Duke-Site Director of an NSF Industry-University Cooperative Research Center between North Carolina State University, Raleigh, and Duke University for carrying out applied research in computing and communications. He has been on the Duke faculty since 1975. He has served as a Principal Investigator on various AFOSR, ARO, Burroughs, DARPA, Draper Laboratory, IBM, DEC, Alcatel, Telcordia, Motorola, NASA, NIH, ONR, NSWC, Boeing, Union Switch and Signals, NSF, and SPC funded projects and as a consultant to industry and research laboratories. He is a codesigner of HARP, SAVE, SHARPE, SPNP, and SREPT modeling packages, which have been widely circulated. He has supervised 33 Ph.D. dissertations. He is the author of *Probability and Statistics with Reliability, Queuing and Computer Science Applications*, (Englewood Cliffs, NJ: Prentice-Hall, 1982) and has recently published *Performance and Reliability Analysis of Computer Systems* (Norwell, MA: Kluwer, 1995) and *Queueing Networks and Markov Chains* (New York: Wiley, 1998). His research interests are in reliability and performance assessment of computer and communication systems, topics of which he has published over 300 articles and lectured extensively.

Dr. Trivedi is a Golden Core Member of IEEE Computer Society. He was an editor of the IEEE TRANSACTIONS ON COMPUTERS from 1983 to 1987.