Proceedings of the 2009 Winter Simulation Conference
M. D. Rossetti, R. R. Hill, B. Johansson, A. Dunkin, and R. G. Ingalls, eds.

SURVIVABILITY MODELING WITH STOCHASTIC REWARD NETS

Poul E. Heegaard

Department of Telematics
Norwegian University of Science and Technology (NTNU)
Trondheim, N-7491, Norway

Kishor S. Trivedi

Pratt School of Engineering
Duke University,
Durham, NC 27708, USA

ABSTRACT

Critical services in a telecommunication network should survive and be continuously provided even when undesirable events like sabotage, natural disasters, or network failures happen. The network survivability is quantified as defined by the ANSI T1A1.2 committee which is the transient performance from the instant an undesirable event occurs until steady state with an acceptable performance level is attained. Performance guarantees such as minimum throughput, maximum delay or loss should be considered.

This paper demonstrates alternative modeling approaches to quantify network survivability, including stochastic reward nets and continuous time Markov chain models, and cross-validates these with a process-oriented simulation model. The experience with these modeling approaches applied to networks of different sizes clearly demonstrates the trade-offs that need to be considered with respect to flexibility in changing and extending the model, model abstraction and readability, and scalability and complexity of the solution method.

1 INTRODUCTION

Our society is critically dependent on a wide variety of telecommunication services, and telecommunication networks and services today are part of the national critical infrastructure that needs to be protected. Hence, evaluation of network survivability is of outmost importance under a variety of threats, like attacks, accidents, and failures, that may cause minor or major service degradations. Specifically, *survivability* is quantified by the transient performance after an undesired event has occurred, as specified by (ANSI T1A1.2 Working Group on Network Survivability Performance 2001).

In a multi-service telecommunication network it is essential to provide *virtual connections* between peering nodes ensuring an overall good utilization of the network resources, and at the same time providing differentiated and guaranteed Quality of Service and resilience requirements. The management of such virtual connections is a challenging task since virtual connections need to be continuously operational without unnecessary delays and with priority to highly critical services even when undesired events occur. Many management techniques exist that apply to different network layers, use pre-planned or reactive techniques, and utilize various setup methods with different resource utilization on local or global operational domain and scope of repair. See (Cholda et al. 2007) for an excellent classification of recovery techniques and recent state of the art.

A model for the evaluation of the virtual connection management needs to consider both the behavioral as well as the structural aspects of the system. This means that the model must capture how the performance of the virtual connection is affected by routing and rerouting, by failures, by traffic load variations, by changes in network capacities, and by different service requirements. Structural dependability models typically focus on the probabilities of terminal connectivity, while behavioral models, e.g., as proposed in (Gan and Helvik 2006), take the network dynamics into account and provide steady state service availability. Combining structural and behavior aspects is typically done using simulation models, stochastic Petri nets such as stochastic reward nets, or continuous time Markov chains, e.g., using Markov dependability models or queuing network models for performance analysis, or combined performance and dependability Markov reward type models as in (Meyer 1980, Haverkort et al. 2001, Trivedi 2001).

The main objective of this paper is through a simple example to show how different modeling approaches apply to survivability quantification. A thorough comparison of these approaches is currently being carried out. We use stochastic reward nets and Markov models for the analytic quantification of network survivability and cross-validate the results with simulations using a process-oriented modeling approach. In (Heegaard and Trivedi 2008) the authors studied the survivability in small networks, while in (Heegaard and Trivedi 2009) real-sized networks with performance measures including throughput, loss, and delay (average and distributions) of virtual connections were studied. In the cited paper a time and space decomposition of the model is applied to avoid a too rapid growth in the model size as the network size is increased.

The focus in this paper is on the experience of using the three modeling approaches applied to a small network survivability quantification example. In Section 2 the network survivability quantification is defined and the modeling approach and example are described in Section 3. The results from experiments and analysis on a small network example are presented in Section 4, while a brief qualitative discussion of the modeling approaches is included in Section 5 before the paper is concluded in Section 6.

2 NETWORK SURVIVABILITY

Survivable systems and survivable networks have been designed and evaluated in the literature for many years, see (Knight et al. 2003, Jäger et al. 2007, Mead et al. 2000). The many definitions of *survivability* can be summarized as

> **Survivability** is the *system's* ability to continuously deliver *services* in compliance with the given *requirements* in the presence of failures and other *undesired events*.

In the literature on survivability quantification we find that the concepts of service, requirements and undesired events are specified differently, though the above survivability definition applies in each case. The survivability literature describes *service* from something unspecified (Pioro and Medhi 2004, Guo 2007), a very general specification like "mission" (Mead et al. 2000), to a more specific definition such as "connected logical links" (Modiano and Narula-Tam 2001, Zhu and Lin 2005). The *requirements* are very general such as "fulfill its mission, in a timely manner" (Mead et al. 2000), "complies with its survivability specification" (Knight et al. 2003), "committed QoS continuously" (Jäger et al. 2007), "provide service continuity" (Pioro and Medhi 2004), very unclear "essential functions are still available" (Deutsch and Willis 1988), or closely linked to the application area "logical links remain connected" (Modiano and Narula-Tam 2001). Finally, the undesired events and their effects. commonly refer to *failures* (Pioro and Medhi 2004, Mannie and Papadimitriou 2006) or *failure scenarios* (Zhu and Lin 2005, Jäger et al. 2007) without any reference to the events that caused it. In (Mead et al. 2000) they explicitly indicate that "attacks, failures, and accidents" may cause the system failure and the service degradation.

The example in this paper pertains to the quantification of survivability of virtual connections in telecommunication networks. For this example we say that the i) *service* is a virtual connection with specific quality requirements between specific peering nodes in the network, the ii) *requirement* is the maximum packet loss probability and end-to-end delay of non-lost packets in the virtual connections, and the iii) *undesired events* are link and node failures caused by attacks, accidents, and software and hardware failures. To quantify we use the definition given by ANSI T1A1 (ANSI T1A1.2 Working Group on Network Survivability Performance 2001):

> **Survivability quantification**. The measure of interest M has the value m_0 just before a failure occurs. The survivability behavior can be depicted by the following attributes: m_a is the value of M just after the failure occurs; m_u is the maximum difference between of m_0 and the value M after the failure; m_r is the restored value of M after some time t_r; and t_R is the relaxation time for the system to restore the value of M.

These attributes are illustrated in Figure 1. The measure of interest M will in this paper be performance metrics like the loss probability and the delay distribution of non-lost packets. Specifically, the transient system behavior immediately after the occurrence of a failure can be analyzed under our proposed approach. Early related work can be found in (Chen et al. 2002, Wang et al. 1996) and more recent work in (Cloth and Haverkort 2005, Liu and Trivedi 2006, Liu et al. 2004).

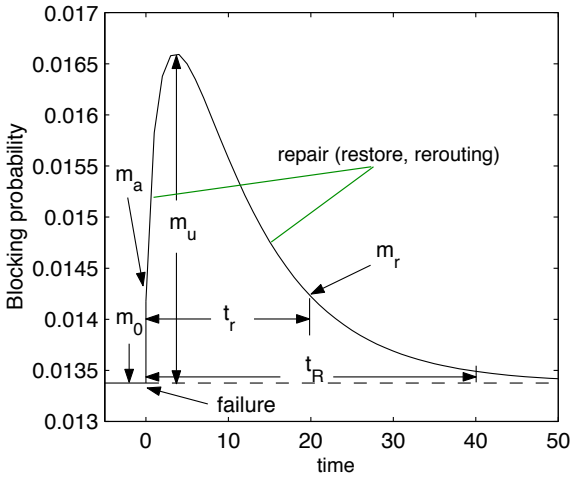


Figure 1: Survivability after first failure

3 NETWORK SURVIVABILITY MODELING

Network survivability models in this paper consider networks exposed to undesired events that cause links and nodes to fail, which are typically followed by a sudden change in the availability of network resources such as bandwidth of the transmission links, queuing positions (memory), and processor capacity. This will typically cause a performance degradation. Gradually the resources are restored through rerouting and by restoration of the failed links and nodes, which eventually results in fully restored performance.

This section introduces a network example for illustration of the survivability modeling approach that is outlined in the following. Two analytical and one simulation model are presented using this approach.

3.1 A 4 Node Example

To illustrate the network survivability modeling, a small network example with $n = 4$ nodes is specified as depicted in Figure 2. The performance of the virtual connection between $s = 1$ and $d = 4$ is evaluated after the failure of node 2 at time $t = 500$. Node i is an $M/M/1/n_i$ system with the parameters given in Table 1. The $\Gamma_i(x)$ is the arrival rate to node i in phase x. The phases are described in Section 3.2.2. The Γ's are obtained by solving a set of traffic equations (Bolch et al. 2006).

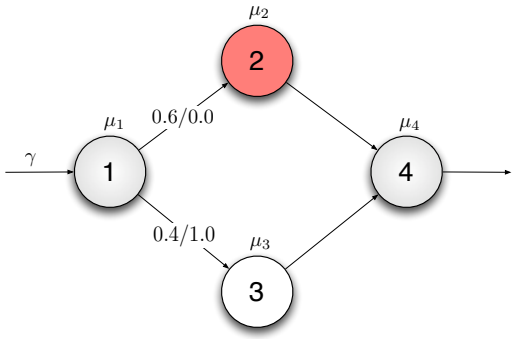


Figure 2: Network example with 4 nodes

Table 1: Parameters for network with 4 nodes

i	n_i	μ_i	$\Gamma_i(IV)$	$\Gamma_i(I)$	$\Gamma_i(II)$	$\Gamma_i(III)$
1	10	100.0	80.0	80.0	80.0	80.0
2	8	100.0	46.9	0.0	0.0	0.0
3	10	100.0	31.2	31.2	78.1	78.1
4	4	100.0	77.9	31.0	69.1	69.1

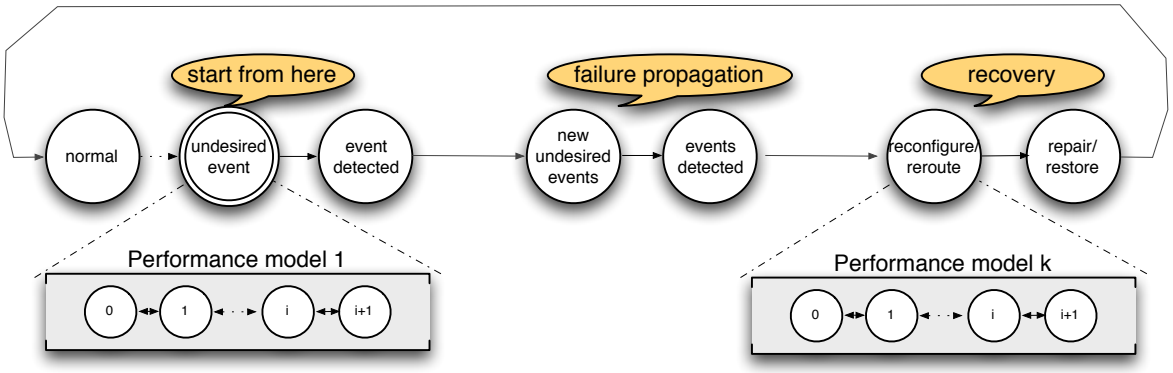


Figure 3: Sequence of failure propagation and recovery

3.2 Survivability Modeling Approach

The survivability model does not consider the frequency of undesired events because the focus is: given that an undesired event has occurred what is the nature of performance degradation just after such an event until the system stabilizes again. The survivability models are constructed by combining performance models with models of the different failure propagation and recovery phases in the system. In (Heegaard and Trivedi 2008) a phased recovery model of the rerouting and restoration was introduced. Figure 3 illustrates the modeling principle where the failure propagation and recovery are modeled as a sequence of phases, illustrated by an example where each phase being a state in a continuous time Markov chain (CTMC) model, and the transitions are caused by events like failure detection, rerouting completed, etc. At time t a (set of) undesired events are assumed to take place whence the transient period of interest begins. A change in the system state is triggered and the evolution of the system is followed through stages of failure propagation, detection, recovery and restoration/repair. Observe that we do not need to know the frequency of undesired events, e.g., the time till failure, because the failure is forced or triggered (dashed line in the figure).

3.2.1 Network Performance Model

The network is a graph $\vec{G} = (v, e)$ where v is the set of nodes and e is the set of links. The single- or multi-path routing of a virtual connection between source node s and destination d reduces $\vec{G}$ to a directed graph $\vec{G}_{[s,d]}$. The analytic network model assumes Poisson external arrivals and exponential service time distribution with an FCFS service discipline at each node and/or link, and the routing between node i and j is stochastic with time-independent probability r_{ij}. Depending upon whether the nodes or the links are the performance bottleneck, one of the following applies

- *Node centric* - the processing a packet in a node is the performance bottleneck,
- *Link centric* - the packet transmission delay at the network interface or the propagation delay over a link is the performance bottleneck,
- *Node and link centric* - packet processing, transmission and propagation are alternating bottlenecks.

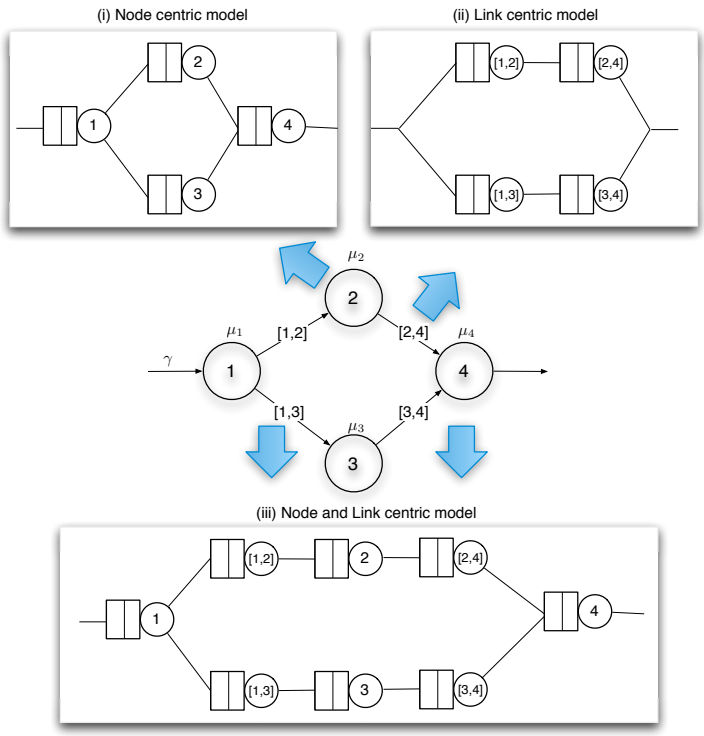


Figure 4: Node and/or link centric models of a four node example

In Figure 4 an illustration of the three possible network performance model views is given for the network example from Section 3.1.

3.2.2 Phased Recovery Model

The phased recovery model describes the "cycle" starting from an undesired event that causes one or multiple links or nodes to fail, and until the system is back to the state just before the undesired event. This can be modeled by phases where each phase may have a different set of available resources for the virtual connections, represented by (possibly) phase-dependent routing probabilities $\{r_{ij}(y)\}$ with corresponding phase-dependent arrival rates $\Gamma_i(y)$. A similar approach was taken in (Wang et al. 1996). But in the cited paper the recovery was not considered whereas here we model both rerouting and restoration of the network resources that brings the system back to fault free operation. In Figure 5 the life cycle of the failure and rerouting is described in four phases, $y = I, \cdots, IV$. The dotted (blue) lines in the figure illustrate the phases from the more general model from Figure 3 that are included.

- *Phase I*: Immediately after the failure the rerouting is activated but it takes some time before the rerouting is effective. Meanwhile, the packets are routed according to the original routing scheme.
- *Phase II*: When the rerouting is effective the link or node is still failed. The packets are routed according to a new routing scheme and will avoid these failed links or nodes (if possible).
- *Phase III*: On completion of repair the system returns to failure free state but the routing is yet to change.
- *Phase IV*: After the routing information is restored the network operates in fault free mode, which is an absorbing state for the purpose of survivability analysis as described in (Liu and Trivedi 2006).

This model is just one example of a phased recovery model. Observe that this does not exclude multiple simultaneous node and link failures, and see (Heegaard and Trivedi 2009) for extended phased recovery models for larger networks.

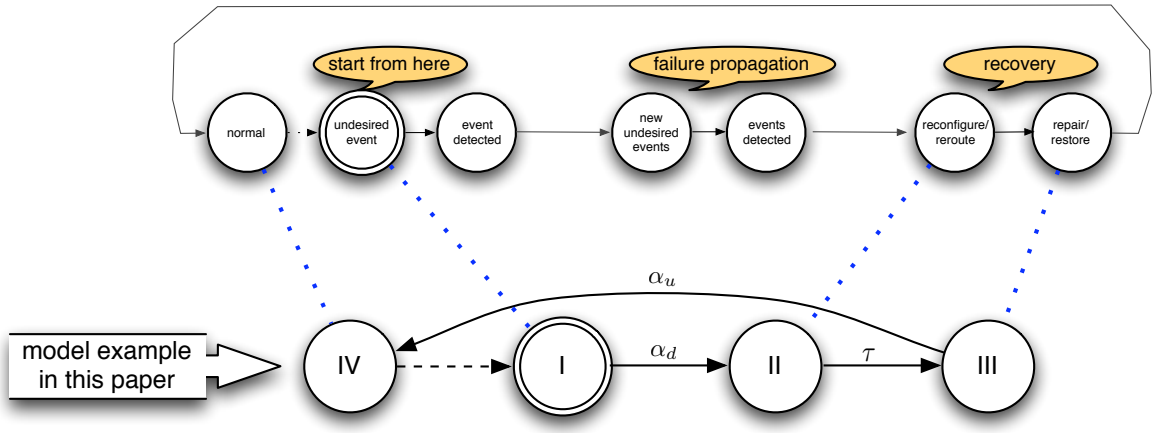


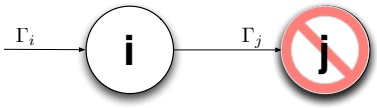
Figure 5: Phased recovery model of rerouting and restoration

3.3 Continuous Time Markov Chain Model

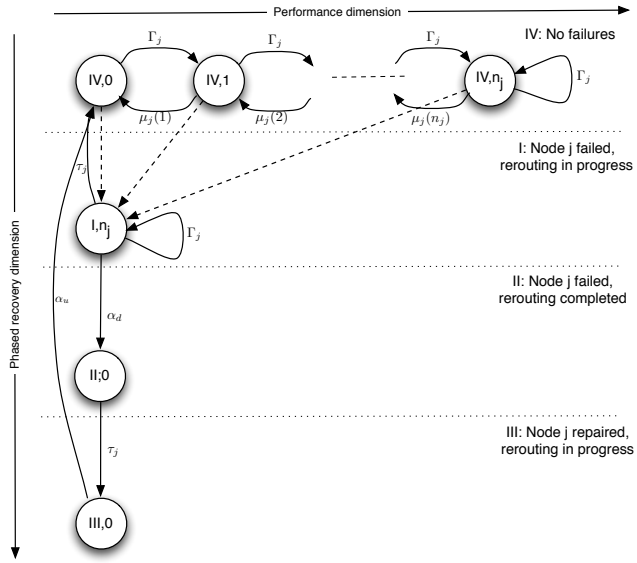
In the example network the pure performance model is a network of blocking queues. So even for steady state analysis normal solution method will be to resort to Markov chains; further in this case we need to carry out transient solution after combining the performance model with the phased recovery model. Continuous time Markov chain (CTMC) is a powerful paradigm for modeling and evaluation of such survivability models. It is clear that CTMC model in our case suffers from a rapid growth in the number of states that are required to represent that system behaviour which is a problem both graphically and for the solution methods. There are two basic approaches to deal with such largeness: largeness tolerance and largeness avoidance. In the former approach we use a more concise, high-level formalism to describe the model which is then automatically transformed into its underlying (albeit large) CTMC. One such approach is based on variants of stochastic Petri nets that we will discuss in Section 3.4. The second approach is to solve a set of smaller submodels to produce the overall result. Such largeness avoidance methods often but not always produce approximate results. Depending upon the application and parameter values, the approximation error is small enough to be acceptable. To avoid largeness we assume that the transient behavior can be modelled in each node separately, akin to a product-form solution in (Jackson 1957) or the BCMP networks in (Baskett et al. 1975). This space decomposition splits the survivability model into independent node (and/or link) models and obtains the arrival rates solving a set of traffic equations. The node dynamics depends on whether a link connected to this node, or the node itself, has failed or not. The CTMC models for the two cases in our network example are as follows:

1. CTMC model for the *node that has failed* (see Figure 6(b)). Immediately after the undesired event all the packets that are sent to node j are lost, hence all transitions lead to state (I, n_j) where all resources are unavailable and no packets will be served. Here n_j is the number of messages at node j.
2. CTMC model for the *non-failed nodes* (see Figure 6(c)). Immediately after the undesired event the network state is changed from (IV, x_i) to (I, x_i). This means that no packets are lost but the arrival rates are changed. For some nodes the arrival rates are unchanged, but for the nodes that used to receive packets from node j the arrival rate is reduced. In phase II the rerouting is completed and the new arrival rate depends on the position of i relative to j.

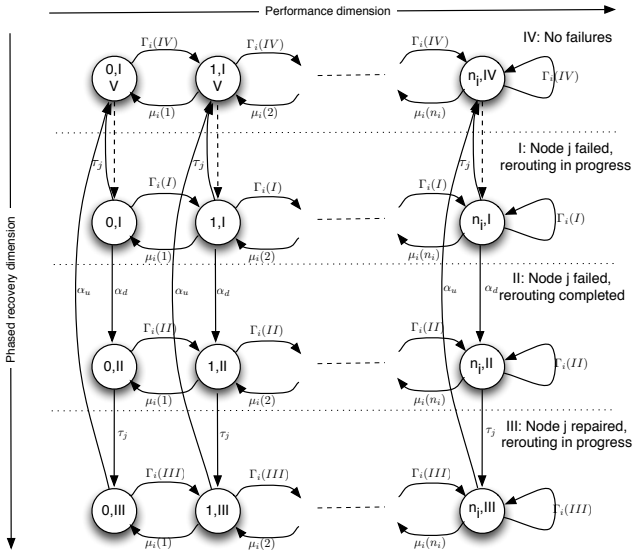
There is no easy way to obtain closed form solutions from the models in Figures 6(b) and 6(c). But, numerical solutions can be obtained by means of tools like SHARPE in (Sahner et al. 1996, Hirel et al. 2000) for rather large systems. However, as the size of the network node model increases, caused either by a more sophisticated recovery model or by an increase in the number of buffers n_i, the solution methods become resource demanding and slow. Thus we need to continue to look for improvement in solution methods.



(a) Node j has failed



(b) CTMC model of failed node



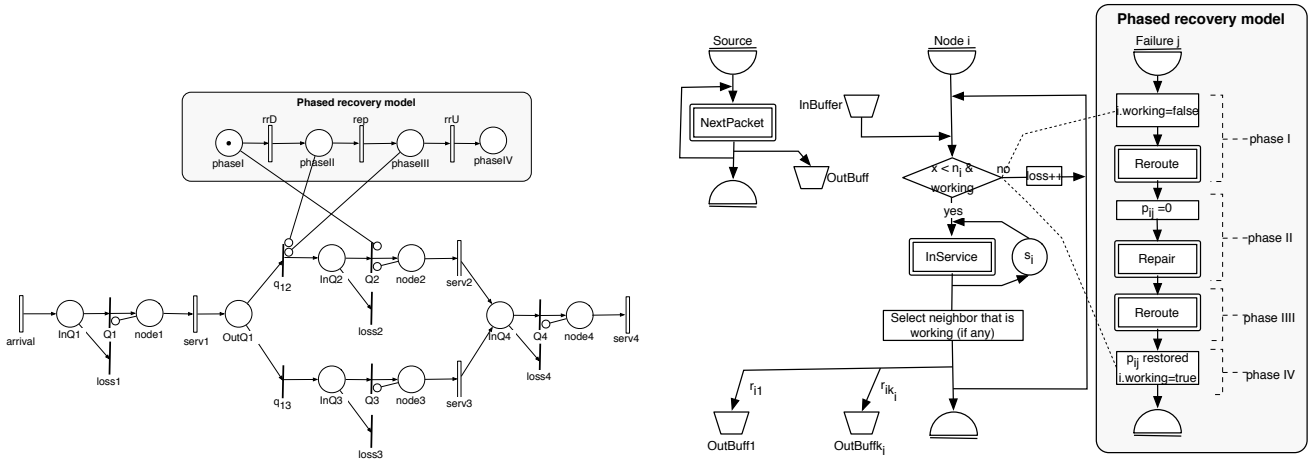
(c) CTMC model of non-failed node

Figure 6: CTMCs in failure state

3.4 Stochastic Reward Net Model

Largeness tolerance starts with a more concise description of the model. We use the formalism of stochastic reward net. Stochastic reward net (SRN) model is stochastic Petri net with many advanced features including enabling functions (or guards), variable cardinality arcs and reward definition at the net level. It is also a specially suited paradigm for the reliability, availability, performability and survivability models. An additional advantage is that an SRN model can be solved by analytic numeric method or by discrete-event simulation via the SPNP package. Figure 7(a) shows the SRN model of the 4-node network from Section 3.1. The phased recovery model is highlighted by a light gray box. The packets are tokens that are generated by the timed transition "arrival" into the place "InQ1". If there are less than n_1 tokens in place "Node1" the immediate transition "Q1" is enabled. If not, the "loss1" transition is enabled upon its firing, the token is removed and a packet loss is counted. The same structure is replicated for each node. The routing is determined by probabilities on the immediate transitions out of the place "OutQ1". The rerouting, failure and repair cycle is modeled at the top of the SRN model in Figure 7(a) where the tokens in the "phasey" places will constrain the token passing of the failed node (Node 2 in this example) through guard function or inhibitor arcs as illustrated in the figure.

The performance metrics in the SRN model are obtained by assigning reward rates that depend on the markings in different places. However, to obtain the analytical transient measures from the SRN model a complete multidimensional CTMC model is generated and solved. This is computationally demanding and increases exponentially both with the number of buffer positions and with the number of places. Decomposition of the SRN model as proposed in (Ciardo and Trivedi 1993) utilizes near-independence between nodes. A similar approach is taken as the first step in Section 3.3 to avoid the largeness problem.



(a) Stochastic reward net model (SRN)

(b) Simulation model

Figure 7: Complete model

3.5 Simulation Model

A process-oriented discrete event simulation model is shown in Figure 7(b). The model is described using the semi-formal activity diagrams that are applied for model descriptions in Discrete Event Modeling on SIMULA (DEMOS) presented in (Birtwistle 1997). The *source* process (denoted ENTITY) is generating packets at the ingress router. The handling of a packet in a node is modeled as a *node* process (ENTITY) that describes the packet life cycle which may be interrupted by a *failure* process (ENTITY) at instances of undesired events. This failure process models the phased recovery model and is highlighted by a light gray box. In each node, when the "InBuffer" contains at least one packet (token, a resource denoted BIN) the node proceeds to the next step and checks if the number of tokens exceeds the maximum buffer size and if the node is currently working. The packet is then served and sent to the next node by a random selection among the currently available buffers. The server is modeled by a resource (denoted RES). If no buffers are available due to failure and all routing probabilities are 0, then the packet is counted as lost. The failure is modeled to the right in the figure. At the instant of a failure (phase *I*) the "working" attribute of the "Node *j*" process is set to "false". After the rerouting time (indicated as double lined rectangle) all routing probabilities into the failed node are set to 0 (phase *II*). After repair and rerouting (phase *III*), the routing probabilities are restored back to their initial values and the "working" attribute id is set to "true" again (phase *IV*). The performance metrics in the simulator are obtained by a measurement process that reads counters at regular time intervals. In Figure 7(b) the process details are described while Figure 8 shows the connection between the processes (ENTITIES).

3.6 Summary of Model Components

In Table 2 the model objects and resources are listed. The "components" in a CTMC model is basically global system state vectors that describe the various system states with respect to the modeling objective. As opposed to the state-view in the CTMC models, the SRN and the simulation models in this paper are both process oriented with an event-view. Slightly different approaches are taken to model the node processors and the queues. The SRN models the processor implicitly (given by the marking in the node places) and the queue explicitly as a place, while the simulation model includes the processor as resource but the queue is implicitly modeled as a queue in front of the resource object in DEMOS.

4 CROSS VALIDATION OF MODELS

The cross-validation of analytic models and simulations are conducted by analytic evaluations of the stochastic reward net both in SHARPE (Hirel et al. 2000, Sahner et al. 1996) and SPNP (Ciardo et al. 1996), the CTMC models are numerically solved in SHARPE and the closed form solution is proof-checked in Mathematica. (http://reference.wolfram.com/mathemat

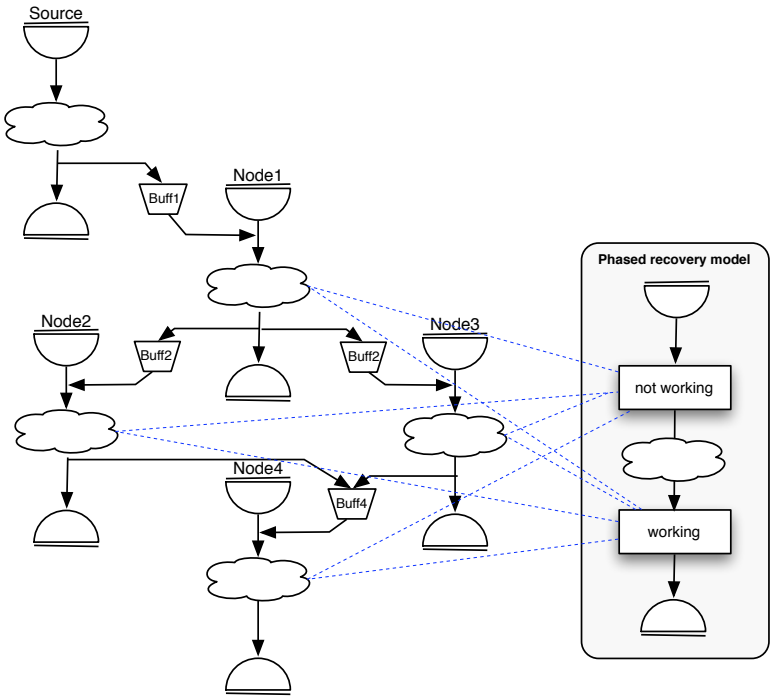


Figure 8: Meta simulation model connecting nodes and phased recovery model

ica/guide/Mathematica.html) The simulation models are implemented using the programming language SIMULA (Kirkerud 1989) with the DEMOS (Discrete Event Simulation on SIMULA (Birtwistle 1997)) class library. The DEMOS simulator is customized to validate the decomposition assumptions by implementing the survivability models without the node independence assumptions and including all details in the transient period under changing conditions. All experiments have been conducted on a MacBook Pro (2.16 Ghz Core 2 Duo, 2 GB RAM, OS X 10.5) with execution times less than 5 seconds in all cases for this small 4-node example.

The four node network example introduced in Section 3.1 uses $\alpha_d = \alpha_u = 0.01$ and $\tau = 0.001$ as the parameters in the phased recovery model. The estimated performance metrics are computed from $R = 90$ simulation runs. The results are denoted *Decomposed CTMC model*, *SRN model* and *Simulations* in the figure. The loss probability and the average number of packets in the system at different time epochs t are shown in Figures 9(a) and 9(b), respectively.

From the results in Figure 9 we observe that after the undesired event (a failure in node 2 at $t = 500$) the main route of the virtual connection between node 1 and 4 is no longer available and the loss ratio is high and the packet throughput low. After a short time period the rerouting takes effect and all traffic is routed via node 3 instead. This leads to an increase in the packet throughput and a decrease in the loss ratio, eventually back to the normal operation level m_0 when the node 2 restored and the virtual connection is routed through node 2 again.

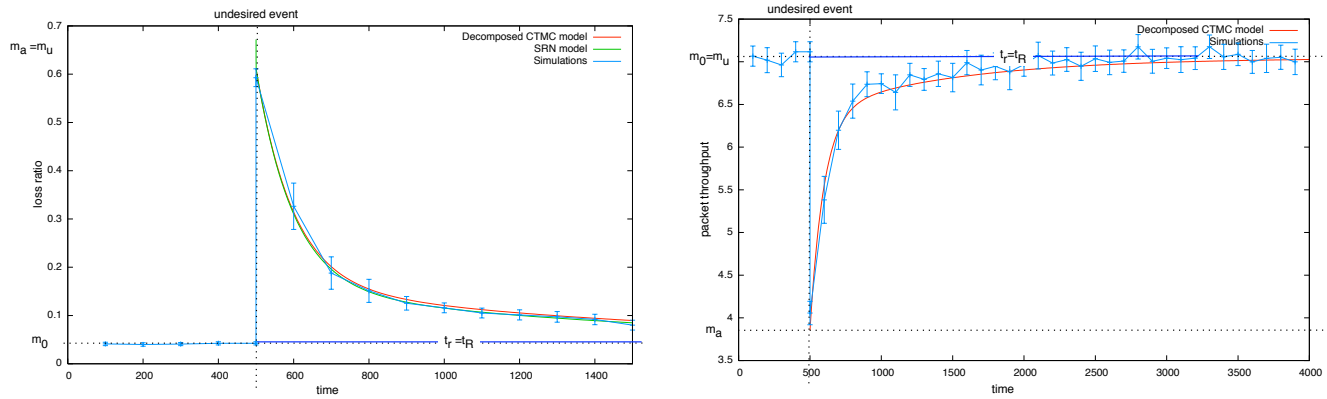
The main observations from the experiments are that the simulations and SRN models show perfect fit as expected since the modeling assumptions are identical, and that the CTMC models capture the transient performance very well even though the decomposed model considers nodes independently.

5 QUALITATIVE COMPARISONS

The modeling paradigm and the *readability* given by the graphical notation and complexity are comparable for SRN and the activity diagrams of the process-oriented simulation models (POS). They both take an "event-view" as opposed to the "state-view" of the CTMC. This means that if your main focus is on the states and the evolution of the states, a CTMC model is easier to read than SRN or POS. On the other hand, since the SRN and POS have a higher level of abstraction they will produce smaller and more compact graphical representations. However, the SRN and POS models will also suffer

Table 2: Modeling components and state definitions

	Continuous time Markov chain	**Stochastic reward nets**	**Process-oriented, event-driven simulation**
Node	A dimension in the CTMC model	Place	Entity
Processor	Number of packets in service (0/1)	*implicit (marking in place)*	Resource (RES)
Link	Valid state transition	*implicit (connected places)*	*implicit (connected ENTITIES)*
Queue	Number in states	Place	*implicit (queue linked to RES)*
Packet	Number in states	Token	Resource (BIN)
State	State vector of resource utilization	Combination of markings	Operational modes or nodes and links and number of packets in service and queue
Event	Change in state vector	Firing: moving tokens from a place to the next	End of wait (HOLD) and seizure and release of resource



(a) Loss probability, $l(t)$

(b) Mean number in the system, $N(t)$

Figure 9: Performance in 4 node network

from reduced readability when the network size, amount of interactions, conditional event, increase. In particularly, taking hidden details like firing or event conditions into account that are not visible in the graphical representation.

The *scalability* is a problem not only with respect to graphical representation but also for analytical solvers to deal with the state explosion. This is common for CTMC, but also for SRN (and other Petri net models) because they need to expand the model to a state model before the analytical quantification of the network survivability can be determined. In a simulator the scalability problem is related to an event "explosion" because a huge number of events needs to be simulated, e.g., packet arrivals and service, to estimate the survivability measures of interest.

All approaches are characterized by a high level of *modeling flexibility* but the simulation approach is the most flexible in that an arbitrary level of modeling details and general stochastic distributions can easily be included.

As pointed out above the analytic *solution methods* for SRN and CTMC are the same and meet the same challenges. The SRN and CTMC can also be simulated by discrete event simulations but this put restrictions on the generality of the stochastic distributions in the models that the process-oriented, event driven simulation approach is less exposed to. Of course, extended types of Petri nets such as Markov regenerative stochastic Petri nets (MRSPN) can be used (Choi et al. 1994). To meet the state explosion problem a time and space decomposition of the model was introduced in (Heegaard and Trivedi 2009). This is an efficient and applicable approach in modeling of network survivability where the time granularity in the performance and phased recovery models are significantly different.

6 CLOSING REMARKS

This paper compares stochastic reward nets and continuous time Markov chain models for survivability assessments and cross-validate with a process-oriented simulation model. The experience with these modeling approaches applied to networks of different sizes clearly demonstrates the trade-offs that need to be considered with respect to model abstraction, complexity, and flexibility. Graphically SRN scales better than CTMC. Analytically they need to solve the same CTMC and will have to deal with an exponential increase in model size which makes analytical solutions inefficient or intractable. Simulations are an attractive alternative when the analytical modeling assumption of SRN and CTMC does not hold. Simulations become inefficient when the number of events (e.g., packet arrivals and departures in a network model) increases. For the small example in this paper the evaluation efficiency is not a problem for any of the cases but then as the network size and load increases this should be studied. A comparison of the execution cost of the analytical and simulation approaches is an interesting next step.

REFERENCES

ANSI T1A1.2 Working Group on Network Survivability Performance 2001, February. Technical report on enhanced network survivability performance. Technical Report TR No. 68, ANSI.

Baskett, F., K. M. Chandy, R. R. Muntz, and F. G. Palacios. 1975. Open, closed, and mixed networks of queues with different classes of customers. *J. ACM* 22 (2): 248–260.

Birtwistle, G. 1997. Demos - a system for discrete event modelling on simula.

Bolch, G., S. Greiner, H. de Meer, and K. S. Trivedi. 2006. *Queueing networks and markov chains: Modeling and performance evaluation with computer science applications*. 2 ed. ISBN 0471565253. John Wiley.

Chen, D.-Y., S. Garg, , and K. S. Trivedi. 2002, September. Network survivability performance evaluation: A quantitative approach with applications in wireless ad-hoc networks. In *ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM' 02)*. Atlanta, GA: ACM.

Choi, H., V. G. Kulkarni, and K. S. Trivedi. 1994. Markov regenerative stochastic petri nets. *Perform. Eval.* 20 (1-3): 337–357.

Cholda, P., A. Mykkeltveit, B. Helvik, O. Wittner, and A. Jajszczyk. 2007. A Survey of Resilience Differentiation Frameworks in Communication Networks. *Comm. Surveys and Tutorials* 9 (4): 2–30.

Ciardo, G., A. Blakemore, P. F. Chimento, J. K. Muppala, and K. S. Trivedi.. 1996. Automated generation and analysis of Markov reward models using stochastic reward nets. In *Linear Algebra, Markov Chains and Queuing Models*, ed. C. Meyer and R. Plemmons, 145–191: Springer.

Ciardo, G., and K. S. Trivedi. 1993. A Decomposition Approach for Stochastic Reward Net Models. *Performance Evaluation* 18 (1): 37–59.

Cloth, L., and B. R. Haverkort. 2005. Model Checking for Survivability. In *Proceedings of the Second International Conference on the Quantitative Evaluation of Systems (QEST'05) on The Quantitative Evaluation of Systems*, 145–154. Washington, DC, USA: IEEE Computer Society.

Deutsch, M. S., and R. R. Willis. 1988. *Software quality engineering: a total technical and management approach*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc.

Gan, Q., and B. E. Helvik. 2006, 3-5 April. Dependability modelling and analysis of networks as taking routing and traffic into account. In *Proceedings of The Second EuroNGI Conference on Next Generation Internet Design and Engineering*. Valencia, Spain: IEEE.

Guo, L. 2007. A new and improved algorithm for dynamic survivable routing in optical WDM networks. *Computer Communications* 30 (6): 1419–1423.

Haverkort, B. R., R. Marie, G. Rubino, and K. Trivedi. 2001. *Performability modelling*. Wiley.

Heegaard, P. E., and K. S. Trivedi. 2008, June 24-27. Survivability quantification of communication services. In *The 38th Annual IEEE/IFIP International Conference on Dependable Systems and Networks*, 462–471. Anchorage, Alaska, USA.

Heegaard, P. E., and K. S. Trivedi. 2009. Network survivability modeling. *Comput. Netw.* 53 (8): 1215–1234.

Hirel, C., R. A. Sahner, X. Zang, and K. S. Trivedi. 2000. Reliability and Performability Modeling Using SHARPE 2000. In *TOOLS '00: Proceedings of the 11th International Conference on Computer Performance Evaluation: Modelling Techniques and Tools*, 345–349: Springer-Verlag.

Jackson, J. R. 1957, Aug. Networks of waiting lines. *Operations Research* 5 (4): 518–521.

Jäger, B., J. Doucette, and D. Tipper. 2007. Network survivability. In *Information Assurance Dependability and Security in Networked Systems*, ed. Y. Qian, J. Joshi, D. Tipper, and P. Krishnamurthy. Elsevier.

Kirkerud, B. 1989. *Object-oriented programming with simula*. Addison Wesley.

Knight, J., E. A. Strunk, and K. J. Sullivan. 2003. Towards a rigorous definition of information system survivability. In *DISCEX*. Washington DC.

Liu, Y., V. B. Mendiratta, and K. S. Trivedi. 2004. Survivability Analysis of Telephone Access Network. In *ISSRE '04: Proceedings of the 15th International Symposium on Software Reliability Engineering*, 367–378. Washington, DC, USA: IEEE Computer Society.

Liu, Y., and K. S. Trivedi. 2006. Survivability quantification: The analytical modeling approach. *International Journal of Performability Engineering* 2 (1): 29–44.

Mannie, E., and D. Papadimitriou. 2006, March. Recovery (protection and restoration) terminology for generalized multi-protocol label switching. Technical Report IETF RFC-4427, IETF.

Mead, N. R., R. J. Ellison, R. C. Linger, T. Longstaff, and J. McHugh. 2000. Survivable network analysis method. Technical Report 013, CMU/SEI.

Meyer, J. Aug. 1980. On evaluating the performability of degradable computing systems. *IEEE Transactions on Computers* C-29 (8): 720–731.

Modiano, E., and A. Narula-Tam. 2001. Designing survivable networks using effective routing and wavelength assignment (rwa). *Optical Fiber Communication Conference and Exhibit, 2001. OFC 2001* 2:TuG5–1–TuG5–3.

Pioro, M., and D. Medhi. March 2004. *Routing, flow and capacity design in communication and computer networks*. ISBN 0125571895. Morgan Kaufmann Publishers.

Sahner, R. A., K. S. Trivedi, and A. Puliafito. 1996. *Performance and reliability analysis of computer system: An example-based approach using the sharpe software package*. Kluwer Academic Publishers.

Trivedi, K. S. 2001. *Probability and statistics with reliability, queuing, and computer science applications*. 2 ed. ISBN 0-471-33341-7. John Wiley and Sons.

Wang, C.-Y., D. Logothetis, K. S. Trivedi, and I. Viniotis. 1996, March. Transient behavior of ATM networks under overloads. In *IEEE INFOCOM' 96*, 978–985. San Francisco, CA: IEEE.

Zhu, Y., and R. Lin. 2005, 7-10 November. Dynamic survivable routing in wdm networks with shared risk link groups. In *Network architectures, management, and applications III*. Shanghai, China.

AUTHOR BIOGRAPHIES

POUL E. HEEGAARD is Associate Professor and Head of Department at Department of Telematics, Norwegian University of Science and Technology (NTNU). Heegaard received his Siv.ing. (M.S.E.E. in '89) and his Dr. Ing. (PhD in '98) degrees from the University of Trondheim (now NTNU). Heegaard was a Research Scientist and Senior Scientist at SINTEF Telecom and Informatics (1989-1999) and Senior Research Scientist at Telenor R&I (1999-2009). In the academic year 2007/08 he was a visiting professor at Duke University, Durham, NC. His research interests cover performance, dependability and survivability evaluation of communication systems. Special interests are rare event simulation techniques, IP network monitoring and modeling, and distributed, autonomous and adaptive management and routing in communication networks and services. His e-mail address is poul.heegaard@item.ntnu.no, and his web page can be found at <www.item.ntnu.no/∼poulh/>.

KISHOR S. TRIVEDI holds the Hudson Chair in the Department of Electrical and Computer Engineering at Duke University, Durham, NC. He has been on the Duke faculty since 1975. He is the author of a well known text entitled, Probability and Statistics with Reliability, Queuing and Computer Science Applications, published by Prentice-Hall; a thoroughly revised second edition (including its Indian edition) of this book has been published by John Wiley. He has also published two other books entitled, Performance and Reliability Analysis of Computer Systems, published by Kluwer Academic Publishers and Queueing Networks and Markov Chains, John Wiley. He is a Fellow of the Institute of Electrical and Electronics Engineers. He is a Golden Core Member of IEEE Computer Society. He has published over 420 articles and has supervised 42 Ph.D. dissertations. He is on the editorial boards of IEEE Transactions on dependable and secure computing, Journal of risk and reliability, international journal of performability engineering and international journal of quality and safety engineering. He is the recipient of IEEE Computer Society Technical Achievement Award for his research on Software Aging and Rejuvenation. His research interests are in reliability, availability, performance, performability and survivability modeling of computer and communication systems. He works closely with industry providing short courses on reliability, availability, performability modeling and in the development and dissemination of software packages such as SHARPE and SPNP. His e-mail address is kst@ee.duke.edu, and his web page can be found at <http://www.ee.duke.edu/∼kst/>.