Stochastic Modeling of e-Commerce Systems' Availability

I.S. Hristoski*, P.J. Mitrevski** and Z.G. Kotevski**
* "St. Clement of Ohrid" University in Bitola/Faculty of Economics, Prilep, Republic of Macedonia
** "St. Clement of Ohrid" University in Bitola/Faculty of Technical Sciences, Bitola, Republic of Macedonia
{ilija.hristoski, pece.mitrevski, zoran.kotevski}@uklo.edu.mk

Abstract - Besides other relevant components encompassing the dependability aspects of contemporar -Commerce systems (reliability, maintainability, safety, security, etc.), availability has always been considered the most prominent one, having minded its direct impact on Internet companies' reputation and financial revenues. Being a synonym for a characteris of a resource/system that is committable, operable, or usable upon demand to perform its designated or required function, availability is the single crucial attribute of each e-Commerce system and a basic prerequisite that makes the huge difference between success and failure. In order to assure and maintain pertinent Quality-of-Service (QoS) levels of Web services being delivered online, including high availability, stochastic predictive models have to be developed and evaluated on a regular basis. The aim of the paper is to address the most significant aspects of stochastic modeling of e-Commerce systems' availability, using the class of Generalized Stochastic Petri Nets (GSPNs), as well as Continuous-Time Markov Chai (CTMCs), a class of stochastic processes underlying GSPNs. A few basi -Commerce system configurations have been modeled and analyzed in the case of a corrective maintenance. The current possible analysis methodologies that address the concept of availability have been discussed, and adequate software too been reviewed, as well.

I. INTRODUCTION

As e-Commerce paradigm goes mainstream, a tremendous attention has been put on e-Commerce systems, which are expected to deliver high quality services online. Besides exhibiting high performances to e-Customers regarding the operational speed or response time, they also have to be highly dependable, i.e. highly reliable and available. Indeed, the assessment of performances, reliability and availability is a key step in the design, analysis and tuning of computer systems, especially e-Commerce systems. In general, the availability of Web services becomes one of the most significant characteristics that should be successfully addressed by companies which run secure trading businesses, based on Internet technologies, e.g. e-Commerce, electronic funds transfer (EFT) systems, e-Banking, online auctions, as well as online brokerage. For such businesses, the availability of Web services is a key QoS metrics, since the unavailability of the corresponding Web services may lead to terrific losses, often measured in millions of dollars per hour [1]. This is even more exaggerated with larg -Commerce systems that deploy mission-critical applications. It is not uncommon for large Web sites to be extremely complex, since they are built out of thousands of components including servers, firewalls, communication links, storage boxes, data centers and all sorts of software systems. On the other hand, the rush to become visible/operational online as soon as possible, often comes at the expense of lack of careful design and testing, leading to many system vulnerabilities. The lack of proactive and continuous capacity planning procedure may lead to performance problems, but also to an unexpected unavailability, caused by failed routers, LAN segments, or other components.

Downtimes may be financially devastating to such companies, since average downtime cost per hour may range from thousands to millions of dollars, depending on the industry [2]. For instance, the average hourly downtime cost in credit card transactions is estimated to be $6.5 million [3]. Recently, Emerson Network Power [4] has released a research report based on the Ponemon Institute study [5] that makes an insight to the full economic costs of unplanned data center outages, stating that the serious financial consequences can range from a minimum cost of almost $40,000 to a maximum cost of more than $1,000,000 per a single incident (more than $11,000 per minute). According to the same source, the average cost per a single downtime incident is estimated more than $500,000. In addition, based on a survey carried out by ITIC, DiDio [6] has revealed a significant fact saying that, while online companies cannot achieve a 'zero downtime' in practice, one out of ten of them needed an availability greater than 99.999% in 2010, i.e. a 'near zero-time downtime'. These figures are pretty much in line with observations claiming that "59% of Fortune 500 companies experienced a minimum of 1.6 hours of downtime per week" in 2010 [7].

However, the consequences of downtimes are far from being solely financial by nature, since their impact on the overall business performances also have long-term and intangible effects, like severe reputational damages, customer churn, as well as lost business opportunities, which can be devastating for doing business online.

All of these important insights point out the great urge of e-Commerce companies to assure and maintain pertinent QoS levels of Web services being delivered online, including high availability. Therefore, stochastic predictive models, both analytical and simulation-based ones have to be continually developed and evaluated.

II. AVAILABILITY: DEFINITION AND BASIC CONCEPTS

For each service request made to a system, there are several possible outcomes that can be generally classified into three disjoint categories, i.e. (1) the system may perform the service correctly, (2) the system may perform the service incorrectly, and (3) the system may refuse to perform the service. If the system does not perform the service at all, it is said to be down, failed, or unavailable. Availability belongs to the group of global metrics, which reflect the systemwide utility. It can be defined as a fraction of time during which a given sy is available/operational to service use requests [3] [8] [9]. This is recognized as steady-state availability. Yet another definition of availability says that it is the probability that a system/component is functioning properly at a given instance of time, no matter how much times it has been down before [9]. This is known as instantaneous, i.e. point, transient, or time-dependent availability. The unavailability of a system is a complement of its availability. It can be caused by many reasons which can be categorized in several subcategories by applying taxonomy following three different dimensions, including: the duration (e.g. permanent, recoverable, and transient failures), the effect (e.g. functional and performance failures), and the scope (e.g. partial and total failures) [3].

A concept very similar to that one of availability is reliability, which can be defined as the probability that a system/component is functioning properly and constantly over a fixed time period [10]. The difference between the two concepts is that reliability takes into account the corrective maintenance of the failed systems/components. In fact, the concept of availability is based on the notion that a given system/component alternates between two states: a state when it is operational (uptime, up period), and a state when it is not functional (downtime, down period).

It is a common practice to label computer systems by the number of '9's, representing their availability. Table 1 depicts a classification of computer systems according to how good their availability is, showing also the projected number of minutes of downtime per year, for each availability class [11]. Each e-Commerce site has to be able to make a real estimation of the needed availability class, taking into consideration potential revenue losses due to unavailability and upgrading costs.

TABLE I. CLASSES OF SYSTEMS VS. THEIR AVAILABILITY

Availability Class	Availability	Time Being Unavailable [min/year]	System Type
1	90.0%	52,560	Unmanaged
2	99.0%	5,256	Managed
3	99.9%	525.6	Well-managed
4	99.99%	52.56	Fault-tolerant
5	99.999%	5.256	Highly availabl
6	99.9999%	0.5256	Very highly ava
7	99.99999%	0.05256	Ultra available

Fig. 1 shows the relationship among the Mean Time to Failure (MTTF), the Mean Time to Repair (MTTR), and the Mean Time Between Failures (MTBF), which are the basic temporal concepts concerning availability.

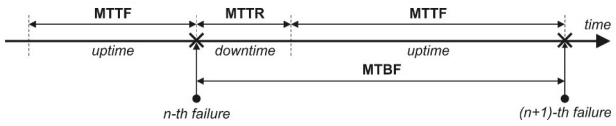

Figure 1. Relationship among MTTF, MTTR, and MTBF (Source: Menascé & Almeida, 2002, p. 420)

From Fig. 1, the following two expressions can be deduced:

$$Availability = \frac{MTTF}{MTBF} = \frac{MTTF}{MTTR + MTTF} \quad (1)$$

$$Unavailability = 1 - Availability = \frac{MTTR}{MTBF} = \frac{MTTR}{MTTR + MTTF} \quad (2)$$

III. COMMON MODELING APPROACHES TO ADDRESS AVAILABILITY

In general, when it comes to assess any measure that has been defined previously, there are several options available, including the following ones: (1) appliance of subjective, experience-based *ad hoc* procedures, i.e. rules of thumb; (2) taking measurements on the real system; (3) building prototypes and taking measurements; (4) construction of analytical models to obtain closed-form solutions; (5) obtaining a numerical solution using a simulation model either by using specialized software tools or by performing discrete-event simulation (DES). Each approach has its own strengths and weaknesses in terms of its accessibility, ease of construction/appliance, efficiency, accuracy, and availability of software tools.

Many specialized techniques have been developed so far, in order to address the concept of availability (along with the reliability) of systems, including the following probabilistic, discrete-state models [9]: combinatorial reliability models: series-parallel reliability block-diagrams (RBDs), fault trees (FTs), and reliability graphs; directed acyclic task precedence graphs; product-form queuing networks (PFQNs); Markov and semi-Markov models, including Markov reward models; Stochastic Petri Nets (SPNs). Recently, the usage of dynamic reliability block-diagrams has been proposed, as a natural extension of the ordinary reliability block-diagrams, which can be converted afterwards into Colored Petri Nets (CPNs) to perform a dynamic analysis of the behavioral features, including the correctness of the model itself. Also, a hierarchical approach that combines the advantages of the reliability block-diagrams and the class of Generalized Stochastic Petri Nets (GSPNs) has be presented recently, to quantify both the reliability and availability. In addition, the hierarchical composition approach has been used to evaluate the dependability measures of complex architectures, based on the appliance of both reliability block-diagrams and GSPNs.

It is also worthy pointing out the fact that due to steadily increasing complexity of real-world computer and communication systems, the usage of dedicated software tools for assessing dependability issues have been justified and encouraged, as well. These include DSPNexpress and TimeNET, general-purpose software environments that have been developed by academia, intended for obtaining steady-state and transient solutions for certain classes of stochastic Petri Nets, as well as commercially available software tools, like BlockSim® [1], a specialized, yet commercially available software tool which provides a system analysis using RBDs and/or FTA approach, or Availability Workbench™ (AvSim+ and RCMCost)[2], for system availability simulation and reliability centered maintenance, based on utilization of modeling methods such as FMECA, reliability block diagram (RBD) analysis and fault tree (FT) analysis.

IV. Modeling Availability with GSPNs and CTMCs

The class of Generalized Stochastic Petri Nets (GSPNs) has been initially introduced as a highly suitable modeling and evaluation tool for addressing performances of computing systems. Within GSPN each transition has been assigned a firing time which can be either exponentially distributed (timed transitions), or constant zero (immediate transitions). Immediate transitions always have priority over timed ones to fire. However, if several immediate transitions compete for firing, a specified probability mass function (pmf) is used to break the tie. On the other hand, if several timed transitions compete for firing, a race model is applied so that a transition whose firing time elapses first is the next one to fire. The finite reachability set of a bounded GSPN can be partitioned into two disjoint subsets consisting of vanishing and tangible markings. Vanishing markings comprise those in which at least one immediate transition is enabled, whilst tangible markings include those where only timed transitions or no transitions are enabled. From a given GSPN, an extended reachability graph (ERG) can be generated, containing the markings (both vanishing and tangible ones) of the reachability set as nodes, being connected with arcs showing the transitional rates to move from a given marking to another one. Based on ERG, a reduced reachability graph (RRG) can be constructed, comprised of only tangible markings. Actually, the resulting RRG of a given GSPN model is its underlying CTMC [12] [13].

The stochastic process underlying an arbitrary GSPN model is known as Continuous-Time Markov Chain (CTMC). In fact, Marsan, Conte, and Balbo [14] have proved that exactly one CTMC corresponds to a given GSPN under condition that only a finite number of transitions can fire in finite time with non-zero probability. CTMC is a mathematical model which takes values in some finite or infinitely countable set, known as state space S, and for which the time spent in each state takes non-negative real values, exponentially distributed. This continuous-time stochastic process is being characterized by the Markov property, known also as the 'memoryless property': the future behavior of the model/system (both remaining time in the current state and choosing next state) depends only on the current state of the model, and not on its past behavior. Each CTMC is being uniquely defined by (1) the state space S, (2) the corresponding transition rate quadratic matrix Q, known as an infinitesimal generator matrix, having dimensions equal to that of the state space S, and (3) the initial probability distribution row-vector, defined on the state space S. For states $i \neq j$, the elements q_{ij} of the matrix Q are non-negative, describing the rates the stochastic process transits from state i to state j. However, the elements q_{ii} ($i = j$) comprising the main diagonal are defined such that each row of the matrix Q sums to zero.

For a given CTMC, two types of evaluations are possible, including a transient and a steady-state analysis. The transient (time-dependent, instantaneous) behavior of a CTMC describes the temporal evolution of the modeled system in each single instance of time. The analysis of the steady-state behavior of a CTMC, also known as a limiting behavior, yields a stationary probability distribution, and refers to a study of the stochastic process' convergence when time tends to infinity ($t \to \infty$). The steady-state probability distribution depends neither on the initial probability distribution, nor on time [12] [15].

Since GSPNs and CTMCs are mutually equal and semantically identical modeling approaches, we proceed by addressing the availability of three specific configurations of e-Commerce systems, using those two modeling paradigms interchangeably. Both approaches are suitable for constructing analytical models to obtain closed-form solutions.

A. The Basic Configuration of an e-Commerce System

On a system level, the most simple configuration of a typical e-Commerce Web site consists of a single system, which alternates between two possible states: operational (up, available) and non-operational (down, unavailable). The occurrence of failures is a stochastic process, i.e. it is a Poisson process, since the following three criteria have been met: (1) failures occur consecutively, i.e. the probability that two failures will occur at the same point of time is equal to zero; (2) the number and intensity of failures in the future is independent on what have happened in the past; (3) the number of failures in the future is an independent and identically distributed (i.i.d.) random variable in time, i.e. the process is stationary. In addition, the times between any two consecutive failures comprise an i.i.d. random variable, exponentially distributed. Since the Markov property is fulfilled at each particular instance of time, the expected, i.e. the mean time to the next occurrence of a failure (MTTF) is a constant, given by $1/\lambda$, where λ is the failure rate. Consequently, the mean time to the next repair (MTTR) of the system, after it has failed down, is given by $1/\mu$, where μ is the repair rate. Knowing this, the availability of the system which is subject to failure and repair can be represented by a two-state homogeneous Continuous-Time Markov Chain (CTMC), depicted on Fig. 2. This is the simplest possible CTMC, which can be used to model the stochastic behavior of many real systems [12].

[1] BlockSim® is a registered trademark of ReliaSoft Corp.

[2] Availability Workbench™ is a registered trademark of Isograph, Inc.

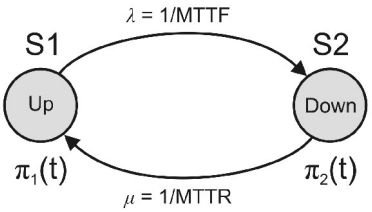

Figure 2. Two-state CTMC availability model (a standard configuration)

Since CTMC is the underlying stochastic process of the class of Generalized Stochastic Petri Nets (GSPNs), the corresponding GSPN model is shown on Fig. 3. Both representations are mutually equivalent regarding their semantics.

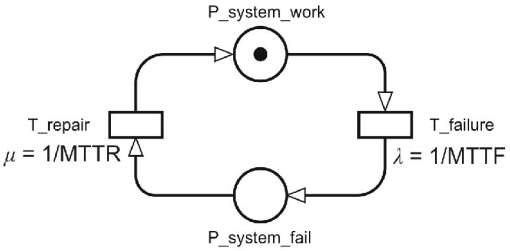

Figure 3. GSPN availability model (a standard configuration)

In fact, the CTMC on Fig. 2 can be obtained directly from the GSPN model on Fig. 3, since the two states represent the two tangible markings within the reachability graph, deduced from the GSPN model. The infinitesimal generator matrix of the CTMC is given by (3):

$$Q = \begin{bmatrix} -\lambda & \lambda \\ \mu & -\mu \end{bmatrix} \tag{3}$$

The transient solution of the CTMC can be obtained from the Kolmogorov differential equation (4) [12]:

$$\frac{d\pi(t)}{dt} = \pi(t) \cdot Q \tag{4}$$

Within (4), $\pi(t)$ denotes the vector containing the transient probabilities, $\pi_1(t)$ and $\pi_2(t)$ of being in the states S1 and S2 (Fig. 2), respectively, i.e. $\pi(t) = \left[\pi_1(t) \quad \pi_2(t)\right]$, where $\pi_1(t) + \pi_2(t) = 1$. The resulting transient probability functions for both states, S1 and S2, are consequently given by (5) and (6) [12], and the availability as a function of λ and μ, in a given time instance $t = 0.5$, is presented on Fig. 4:

$$\pi_1(t) = \frac{\mu}{\lambda+\mu} + \frac{\lambda}{\lambda+\mu} \cdot \left(e^{-(\lambda+\mu)\cdot t}\right) \tag{5}$$

$$\pi_2(t) = \frac{\lambda}{\lambda+\mu} - \frac{\lambda}{\lambda+\mu} \cdot \left(e^{-(\lambda+\mu)\cdot t}\right) \tag{6}$$

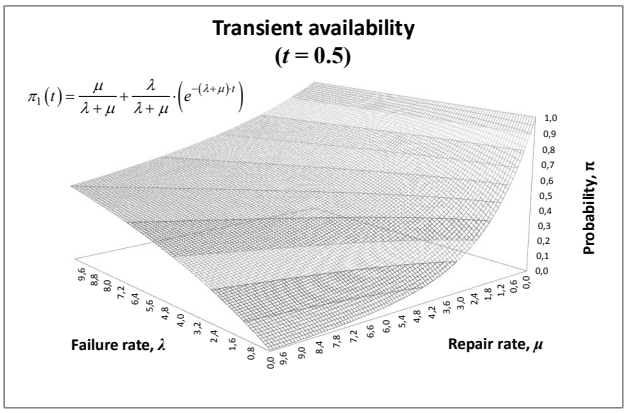

Figure 4. 3D surface showing the availability of a standard configuration, for λ and μ ranging from 0.0 to 10.0, and $t = 0.5$

So, the instantaneous (transient) availability of the system $A(t)$ in a specific time instance t can be obtained by using the expression for calculating $\pi_1(t)$, given by (5), whilst the instantaneous (transient) unavailability of the system is represented by $\pi_2(t)$, given by (6). The limiting (steady-state) availability A of the system can be obtained from the expression for calculating $\pi_1(t)$, by letting $t \to \infty$ (Fig. 5), i.e.

$$A = \lim_{t\to\infty} \pi_1(t) = \frac{\mu}{\lambda+\mu} = \frac{\frac{1}{MTTR}}{\frac{1}{MTTF} + \frac{1}{MTTR}} = \\ = \frac{MTTF}{MTTF + MTTR} = \frac{MTTF}{MTBF} \tag{7}$$

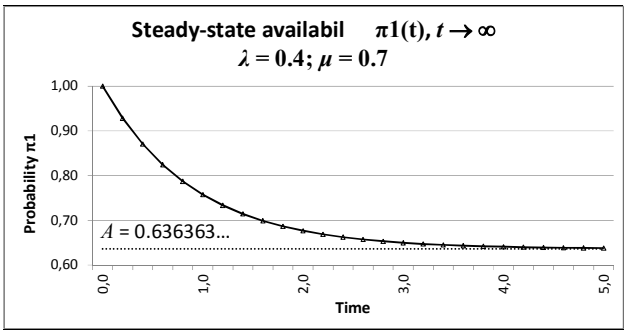

Figure 5. Asymptotic convergence ($t \to \infty$) of the transient availability $\pi_1(t)$ towards the steady-state availability A, for $\lambda = 0.4$ and $\mu = 0.7$

Note that the value of the steady-state availability A, given by (7), is identical to the one already given by (1).

B. Cold Standby Configuration With a Single Redundant Module

In order to improve the availability of a given system, a very common technique is to add an additional redundant module in a cold standby [16], waiting to be activated when the main module fails (Fig. 6).

The performance- and reliability-related characteristics of the spare module are usually not as good as those of the main module's, since it is likely to be a cheaper variant of the main module, thus delivering lower QoS levels of Web services online.

The extended reachability graph (ERG) of the GSPN model presented on Fig. 6 is portrayed on Fig. 7, whilst the reduced reachability graph (RRG), containing only the tangible markings (i.e. the states S1, …, S5), is given on Fig. 8. The tangible markings are presented by ovals, whilst the only vanishing marking is depicted by a rectangle. In fact, the RRG is the underlying CTMC of the GSPN model.

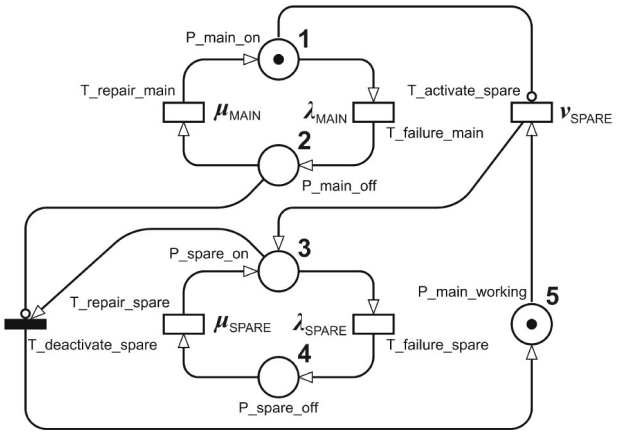

Figure 6. GSPN availability model
(a cold standby configuration with a single redundant module)

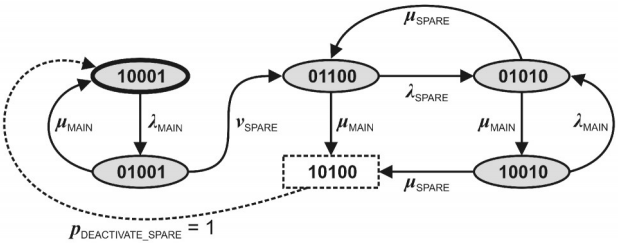

Figure 7. Extended reachability graph (ERG) of the GSPN model

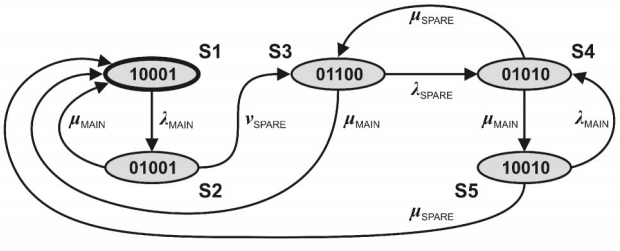

Figure 8. Reduced reachability graph (RRG) of the GSPN model

The infinitesimal generator matrix Q, given on Fig. 9, can be derived directly from the RRG on Fig. 8.

Transient probabilities are given by a row-vector $\pi(t) = \left[\pi_1(t) \;\; \pi_2(t) \;\; \pi_3(t) \;\; \pi_4(t) \;\; \pi_5(t)\right]$, since the state space S of the corresponding CTMC have exactly five states (S1, …, S5), having $\sum_{i=1}^{5} \pi_i(t) = 1$. The transient and steady-state probabilities can be derived in the same way as shown previously. The total availability of the system is simply a sum of steady-state probabilities corresponding to states S1, S3 and S5.

$$Q = \begin{bmatrix} -\lambda_{MAIN} & \mu_{MAIN} & \mu_{MAIN} & 0 & \mu_{SPARE} \\ \lambda_{MAIN} & -(\nu_{SPARE} + \mu_{MAIN}) & 0 & 0 & 0 \\ 0 & \nu_{SPARE} & -(\lambda_{SPARE} + \mu_{MAIN}) & \mu_{SPARE} & 0 \\ 0 & 0 & \lambda_{SPARE} & -(\mu_{MAIN} + \mu_{SPARE}) & \lambda_{MAIN} \\ 0 & 0 & 0 & \mu_{MAIN} & -(\lambda_{MAIN} + \mu_{SPARE}) \end{bmatrix}$$

Figure 9. The infinitesimal generator matrix Q

C. *Horizontally Scaled Configuration*

In order to improve performances, many e-Commerce Web sites have implemented horizontal scaling (i.e. scaling out) by multiplying identical systems to work in parallel in a cluster. This is especially case with particular components/subsystems, e.g. Web servers that have usually proved out to be bottlenecks in the whole system. By scaling out, the total capacity is increased, along with performances, and the whole system becomes highly available, i.e. if a single system fails, it will not affect the e-Customer's ability to continue using slightly degraded Web services. The CTMC of such configuration including $N = 2$ systems working in parallel is given on Fig. 10.

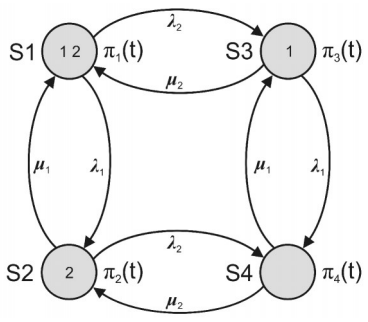

Figure 10. CTMC modeling the availability of a system with two modules working in parallel

It can be easily shown that the above CTMC can be deduced directly from the GSPN model on Fig. 11.

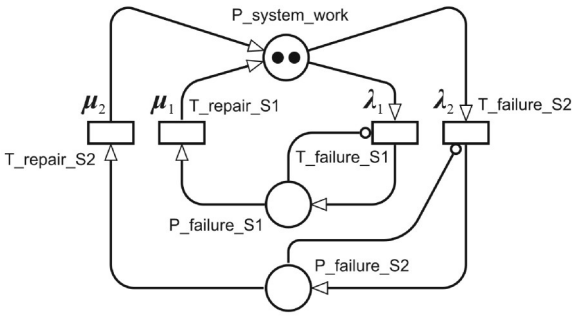

Figure 11. GSPN availability model
(system with two modules working in parallel)

The corresponding infinitesimal generator matrix Q of the CTMC is given by (8).

$$Q = \begin{bmatrix} -(\lambda_1+\lambda_2) & \lambda_1 & \lambda_2 & 0 \\ \mu_1 & -(\mu_1+\lambda_2) & 0 & \lambda_2 \\ \mu_2 & 0 & -(\mu_2+\lambda_1) & \lambda_1 \\ 0 & \mu_2 & \mu_1 & -(\mu_1+\mu_2) \end{bmatrix} \quad (8)$$

Since each of the N systems workin in parallel can be in one of the two possible states (available/non-available) at each single instance of time, the total number of states within the CTMC equals 2^N for this configuration.

V. CONCLUSION

In highly demanding business environments, such as e-Commerce, the corrective maintenance costs and inoperability associated with downtime periods are quite incompatible with the nature of online Web applications and services run by electronic stores, which are expected to be available for their potential e-Customers around the world 24/7 per year. Availability of e-Commerce systems is considered one of the main service level goals of any electronic business, since low availability can cost an online business a significant lost revenue, reduced market share, and bad publicity.

Throughout the previous sections, three different configuration of e-Commerce systems have been considered regarding their availability. In all three cases a corrective maintenance is supposed to take plac , whilst the case of preventive one has not been considered at all. This fact has facilitated the process of stochastic modeling, since all resulting Petri Net-based models belong to the class of GSPNs, which are less time-consuming to analyze compared to models based on utilization of Deterministic and Stochastic Petri Nets (DSPNs). The case of preventive maintenance, which is out of the scope of this paper, assumes inclusion of scheduled downtimes in regular time periods, which necessarily imposes utilization of the class of DSPNs and corresponding analysis methods.

The application of stochastic Petri Nets and Markov chains has proven to be a powerful and an effective tool for analyzing availability aspects of various contemporary e-Commerce systems' configurations. Stochastic Petri Nets possess an immense semantic power to capture the behavior of an arbitrary system/configuration, which exhibits phenomena like synchronization, concurrency, blocking, mutual exclusion, parallelism etc. Despite the fact that Markov chains, applied directly, provide great flexibility, they are not always intuitive to be built from scratch, and the size of their state space grows much faster than the number of systems/components involved, making both the specification and analysis difficult to carry out.

REFERENCES

[1] D. Patterson, "A simple way to estimate the cost of downtime," Proceedings of The 16th USENIX Conference on System Administration, LISA '02, USA, pp.185–188, 2002.

[2] V. McCarthy, "Performance tools kill the gremlins on your net," Datamation, vol. 42(15), pp. 88–91, 1996.

[3] D. A. Menascé and V. A. F. Almeida, Capacity Planning for Web Services: Metrics, Models, and Methods, Upper Saddle River: Prentice Hall PTR, 2002, p. 13, 420, pp. 417–419.

[4] Emerson Network Power, "Understanding the cost of data center downtime: an analysis of the financial impact of infrastructure vulnerability," Liebert Corp., Columbus, OH, USA, 2011. Retrieved June 2014 from http://emersonnetworkpower.com/en-US/Brands/Liebert/Documents/White%20Papers/data-center-uptime_24661-R05-11.pdf

[5] Ponemon Institute, "Calculating the cost of data center outages: benchmark study of 41 US data centers," Ponemon Institute, Traverse City, MI, USA, 2011. Retrieved June 2014 from http://www.emersonnetworkpower.com/documentation/en-us/brands/liebert/documents/white%20papers/data-center-costs_24659-r02-11.pdf

[6] L. DiDio, "Trends in high availability and fault tolerance," Information Technology and Intelligence Corp. (ITIC), 2010. Available at http://searchcio.techtarget.com/podcast/Trends-in-high-availability-and-fault-tolerance. Last accessed June 2014.

[7] A. Arnold, "Assessing the financial impact of downtime," Tech. report, Vision Solutions, 2010. Available at http://www.it-director.com/business/costs/content.php?%20cid=12043. Accessed June 2014.

[8] R. Jain, The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling. New York: John Wiley & Sons, 1991, p. 37, 40.

[9] R. A. Sahner, K. S. Trivedi, and A. Puliafito, Performance and Reliability Analysis of Computer Systems: An Example-Based Approach Using the SHARPE Software Package. Dordrecht: Kluwer Academic Publishers Group, 1996, p. 30.

[10] D. E. Long, A. Muir, and R. Golding, "A longitudinal survey of Internet host reliability". HP Labs Technical Report HPL-CCD-95-4, Palo Alto, CA, USA, 1995. Retrieved June 2014 from http://www.hpl.hp.com/techreports/95/HPL-CCD-95-04.pdf

[11] J. Gray, "FT 101", A Presentation on Fault Tolerance at University of California, Berkeley, CA, USA, 2000. Available at http://research.microsoft.com/en-us/um/people/gray/talks/UCBerkeley_Gray_FT_Availability_talk.ppt

[12] G. Bolch, S. Greiner, H. de Meer, and K. S. Trivedi, Queueing Networks and Markov Chains, 2nd ed., Hoboken: John Wiley & Sons, 2006, pp. 64–71, 96–97, 101–112, 213–215.

[13] M. Ajmone Marsan, G. Balbo, G. Conte, S. Donatelli, and G. Franceschinis, Modelling with Generalised Stochastic Petri Nets, 1st ed., Wiley, 1995, pp. 101–156.

[14] M. Ajmone-Marsan, G. Conte, and G. Balbo, "A class of Generalized Stochastic Petri Nets for the performance evaluation of multiprocessor systems," ACM Transactions on Computer Systems, vol. 2(2), pp. 93–122, 1984.

[15] W. J. Stewart, Probability, Markov Chains, Queues, and Simulation: The Mathematical Basis of Performance Modeling, 1st ed., Princeton: Princeton University Press, 2009, pp. 253–265.

[16] P. J. Mitrevski and I. S. Hristoski, "Behavioral-based performability modeling and evaluation of e-commerce systems," Electronic Commerce Research and Applications, 2014, in press.