A FRAMEWORK FOR PERFORMANCE ANALYSIS OF E-BUSINESS APPLICATIONS

P. Mitrevski[1], G. Mančeski[2], M. Gušev[3]

[1]Faculty of Technical Sciences, St. Clement Ohridski University
Ivo Lola Ribar b.b., 7000 Bitola, Macedonia
[2]Faculty of Economics, St. Clement Ohridski University
Gjorče Petrov b.b., 7500 Prilep, Macedonia
[3]Institute of Informatics, Faculty of Natural Sciences and Mathematics
Sts. Cyril and Methodius University, Skopje, Macedonia
{pece.mitrevski, gjorgji.manceski}@uklo.edu.mk, marjan@ii.edu.mk

Abstract: Stochastic Petri Nets are recognised as useful modeling tools for analysing the performance and reliability of systems. In this paper we present a framework for performance analysis of e-Business applications. The customer behavior in a conceptual mainstream e-Business application is modeled using the class of Deterministic and Stochastic Petri Nets (DSPN). The average time spent by a customer, as well as the average number of customers in the system, can be calculated via time and space efficient algorithms for computing steady state solutions of DSPNs, whereas the use of a transient (time-dependent) analysis method allows to evaluate the general service time distribution. The proposed performance analysis framework reveals considerable potential for further research in this area.

Keywords: e-Business, e-Commerce, performance evaluation, capacity planning, Deterministic and Stochastic Petri Nets, Markov chains

1. Introduction

As e-commerce becomes more mainstream, e-commerce Web sites become an essential necessity for almost any business. From developing the business plan to selecting the hardware and software, each step requires hard work, proper diligence, and large amounts of research. Owning and operating an e-business is a continuous developmental process. Some phases, such as domain registration, obviously need to be done only once (unless a merchant decides to register another domain name), but most of the other steps are part of an ongoing process to keep an e-business up to date.

Predicting *when* the future load levels will saturate the system, and determining the most cost-effective way of delaying the saturation, are the main concern of *capacity planning* [12]. Future load levels are functions of the *natural evolution of existing workloads*, the *deployment of new applications and services*, as well as the *changes in customer behavior*. Therefore, capacity planning requires *predictive models*. The methodology of capacity planning is composed of three main planning processes: (a) *business and functional planning*, (b) *customer behavior planning*, and (c) *IT resource planning*. *Performance prediction* allows to estimate performance measures of a system (e.g. server-side and client-side response time, throughput, network utilization, resource queue length, etc.) for a given set of parameters: *system parameters* (e.g. network protocols used), *resource parameters* (e.g. disk seek time, network bandwidth, router latency), and *workload parameters* (e.g number of sessions per day, number of DB transactions per unit time, the total transmission time). Useful models are: *Customer Behavior Model Graph* – insight into the way customers interact with an e-commerce site, and *C/S Interaction Diagrams* – representation of all possible interactions between a customer and a site for a specific business function [12].

Rather differently, in this paper we employ the class of *Deterministic and Stochastic Petri Nets* (*DSPN*) [1, 4, 5, 7] in building *a framework for performance analysis* based on modeling the customer's behavior in a conceptual mainstream e-business application. The average time spent by a customer, as well as the average number of customers in the system, can be calculated via time and space efficient algorithms for computing steady state solutions of DSPNs, whereas the use of a transient (time-dependent) analysis method allows to evaluate the general service time distribution. The proposed performance analysis framework reveals considerable potential for further research in this area.

2. The lifecycle of a conceptual mainstream e-business application

The simplest description of a conceptual mainstream e-business application is that a customer makes a search through the products catalogue and adds the desired items to the shopping basket [11]. As soon as the customer decides to pay, he or she provides the system with the delivery address, supplies credit card information and places an order. The system verifies credit card authorisation and soon sends the customer an e-mail confirmation regarding the purchase. If, at some point, the server is unable to process customer's last request due to the length of inactivity (maximum allowed time between requests elapsed), the session expires.

Later on, when the products become available, the system forwards the shipping package to a delivery company and updates order status (the customer is presumably off-line).

3. The customer's behavior DSPN model

The DSPN in Figure 1 models the customer's behavior in a conceptual mainstream e-business application. The token in place P_{SEARCH} denotes that a customer is about to make a new search through the products catalogue, which is a time-consuming activity. The firing time of transition T_{END_SEARCH} is exponentially distributed with rate λ. The product is either found (firing of transition T_{FOUND} with probability p_{FOUND}), or not found (firing of transition T_{NOT_FOUND} with probability 1-p_{FOUND}). In addition, it is the customer's decision to add the product to the shopping basket (firing of transition T_{ADD} with probability p_{ADD}) or not (firing of transition T_{NOT_ADD} with probability 1-p_{ADD}). The firing of immediate transition T_{ADD} leaves a token in place P_{IN_BASKET} (removes all (zero or one) tokens and puts back one), just to indicate that the shopping basket is not empty (the number of products is irrelevant). Regardless of the search outcome, the customer can either (a) *make a new search* (probabilities $p_{CONTINUE\_1}$, $p_{CONTINUE\_2}$ and $p_{CONTINUE\_3}$, respectively), (b) *end shopping* without placing an order (probabilities p_{END_1}, p_{END_2} and p_{END_3}, respectively), or (c) *proceed to checkout* provided that the shopping basket is not empty (probabilities $p_{CHECKOUT\_1}$, $p_{CHECKOUT\_2}$ and $p_{CHECKOUT\_3}$, respectively). One can expect the probability of a new search to be the highest when no product was found, and the probability of ending without placing an order to be the lowest when the shopping basket is not empty.

Checking-out is also a time-consuming activity. The firing time of transition $T_{Review\text{-}Submit\text{-}Verify}$ is exponentially distributed with rate μ. The consumer *reviews* and *submits* delivery address and credit card information, credit card authorisation is *verified*, and *order is placed* (token in place P_{ORDER_PLACED}). Right away, the system sends the customer an e-mail confirmation regarding the purchase (firing of transition $T_{E\text{-}MAIL}$) and the lifecycle of the e-business application ends (token in place P_{END}).

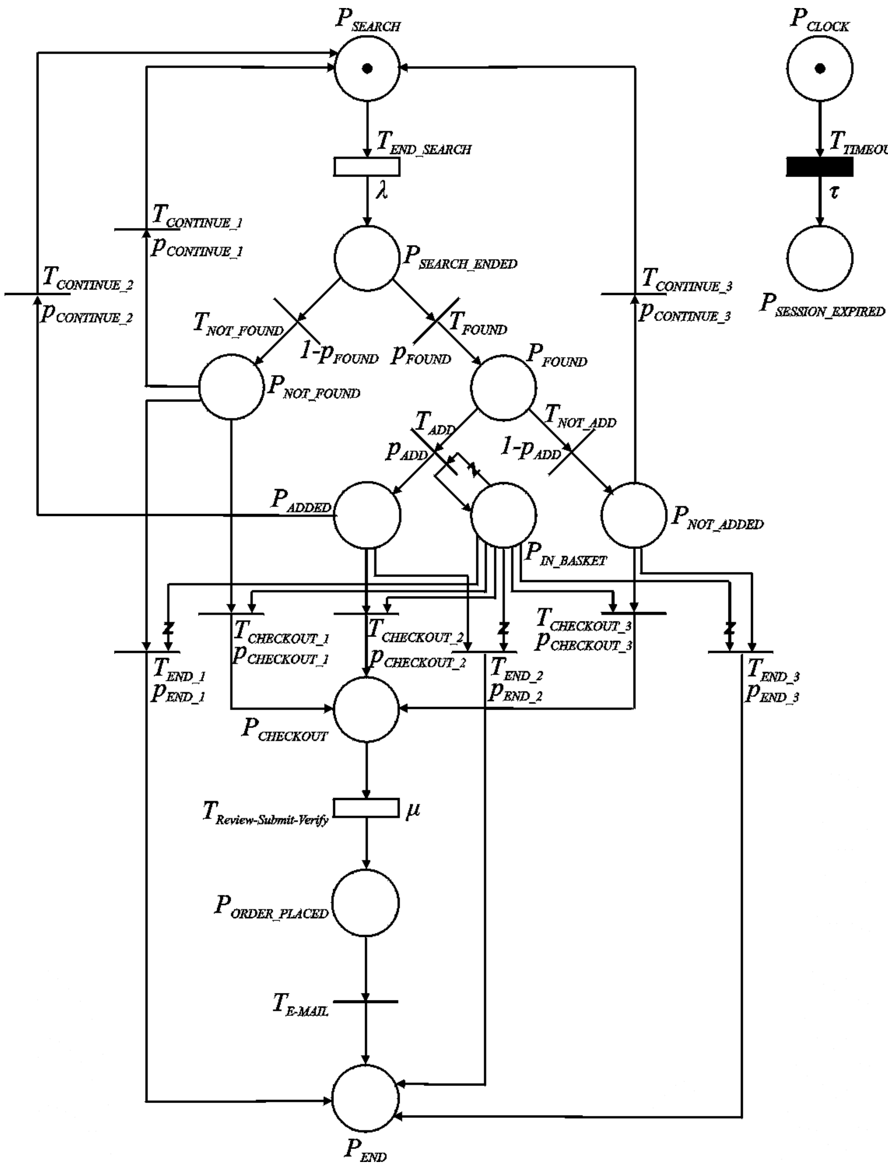

Figure 1: The customer's behavior DSPN model

In order to capture the mechanism of session expiration due to the length of inactivity, a timed transition $T_{TIMEOUT}$ with deterministic firing delay τ and *resampling* policy is introduced. Its *remaining firing time* (RFT) is *resampled* (set to τ) in each new tangible marking of the Petri Net. The firing of $T_{TIMEOUT}$ indicates that the server is unable to process further customer's requests because the maximum allowed time between requests elapsed (token in place $P_{SESSION\_EXPIRED}$).

4. Performance analysis methods

The classes of stochastic timed Petri nets that have been proposed for performance and reliability analysis of systems include Stochastic Petri Nets (SPN) [8], Generalized Stochastic Petri Nets (GSPN) [2], Extended Stochastic Petri Nets (ESPN) [10], and Deterministic and Stochastic Petri Nets (DSPN). In the SPN, a transition fires after an exponentially distributed amount of time (firing time) when it is enabled. The GSPN allows transitions with zero firing times or exponentially distributed firing times. The stochastic process underlying an SPN or a GSPN is a continuous-time Markov chain. The ESPN allows generally distributed firing times and exponentially distributed ones. Under some restrictions, the underlying stochastic process of an ESPN is a semi-Markov process. The DSPN allows transitions with zero firing times or exponentially distributed or deterministic firing times. As a result, the underlying stochastic process of a DSPN is neither a Markov nor a semi-Markov chain [6]. However, DSPNs can be solved analytically with a restriction that at most one timed transition with deterministic firing time to be enabled concurrently with exponentially distributed timed transitions. A steady state solution method for DSPNs appears in [7], and a transient solution method is given in [4]. Furthermore, in [5], the underlying stochastic process for a DSPN is shown to be a Markov regenerative process (MRGP).

5. Time and space efficient algorithms for computing steady state solutions of DSPNs

The idea is to study the evolution of the DSPN while $T_{TIMEOUT}$ is enabled in a continuous-time Markov chain (CTMC) to be solved at the transient time τ, at which time $T_{TIMEOUT}$ must fire unless it has been resampled in the meantime. An *embedded Markov chain* (EMC) is associated with each marking where the RFT of $T_{TIMEOUT}$ can be resampled, and different CTMCs, referred to as *subordinated Markov chains* (SMC), are employed for each marking in which $T_{TIMEOUT}$ can begin its firing time. Vanishing markings do not appear in the state spaces of the EMC and the SMCs. In reference to the time and space efficient algorithm for computing steady state solutions of DSPNs, which has been presented in [7], the following steps have to be carried out:

Step 1. Build the reachability graph of the DSPN ignoring the timing information associated to the transitions. Usually, normal lines are used for exponential, dashed lines are used for immediate, and heavy lines are used for deterministic transition firings.

Step 2. Find the set of markings where the RFT of the deterministic transition is resampled. These are the only markings which are also states of the EMC.

Step 3. For each marking in Step 2, define and solve the associated scaled SMC. The outgoing rates are multiplied by τ and the transient solution at time 1 is computed for this scaled SMC. Compute the expected time spent in each marking until $T_{TIMEOUT}$ either fires or is restarted.

Step 4. Define the holding time vector of the EMC.

Step 5. Define the probability of going from state i_1 to state i_2 in the EMC due to the firing of any number of immediate transitions.

Step 6. Solve the EMC to compute the expected total number of visits in its transient states up to steady state.

Step 7. Compute the cumulative sojourn times for each tangible transient marking of the DSPN up to steady state.

Consequently, *the average time spent by a customer in the system* is equal to the total time spent in all tangible transient markings of the DSPN. Since the number of customers entering the system is equal to those completing service, the Little's law holds: the *average number of customers in the system* is given by the product of the *average arrival rate of customers admitted to the system*, and the *average time spent by a customer in the system*.

6. Transient (time-dependent) analysis methods

The transient analysis can be carried out using the theory of *Markov regenerative processes* [4, 5]. The transition probability matrix $\mathbf{V}(t)$ satisfies the generalized *Markov renewal equation*:

$$\mathbf{V}(t) = \mathbf{E}(t) + \mathbf{K} * \mathbf{V}(t) \tag{1}$$

where $\mathbf{E}(t)$ is a matrix that describes the behavior of the marking process between two transitions epochs of the EMC (*local kernel*), opposed to the kernel $\mathbf{K}(t)$ which is *global* in the marking process. Given the state transition probability matrix and the initial probability distribution, the state probability at time t can be computed.

Consequently, the *general service time distribution* is given by the probability that the marking process is in one of the absorbing states (token in place P_{END} or token in place $P_{SESSION\_EXPIRED}$) at time t. The system may then be regarded as an *M/G/k* queue – one of the classical, but yet unsolved queuing systems, with Poisson input, general service time distribution and k serving facilities. Nevertheless, the queue can be analyzed as a piecewise-deterministic Markov process [3, 9].

7. Conclusion

Stochastic timed Petri nets provide a good framework to model the dynamic behavior and examine the performance of concurrent and asynchronous systems. In this paper we used the class of Deterministic and Stochastic Petri Nets (DSPN) in order to build a framework for performance analysis of e-business applications. The presented model satisfies the sufficient conditions for a stochastic timed Petri Net to be analytically solvable: (a) at most one general transition is enabled in a marking, and (b) the distribution of the firing time of a general transition is marking-independent. The DSPN model captures the customer's behavior in a typical e-business application, whereas the analysis of the stochastic processes underlying the Petri Net, either steady-state or transient, allows to estimate a range of performance measures.

8. References

1. Ajmone-Marsan, M., Chiola, G., "On Petri nets with Deterministic and Exponentially Distributed Firing times", in *Lecture Notes in Computer Science*, Vol. 266, pp. 132-145, Springer-Verlag, 1987
2. Ajmone-Marsan, M., Conte, G., Balbo, G., "A Class of Generalized Stochastic Petri Nets for the Performance Evaluation of Multiprocessor Systems", *ACM Transactions on Computer Systems*, Vol. 2, No. 2, pp. 93-122, May 1984
3. Breuer, L., "Operator-Geometric Solutions for the M/G/k Queue and its Variants", in *Proc. 11th INFORMS Applied Probability Society Conference*, New York, July 2001
4. Choi, H., Kulkarni, V., Trivedi, K., "Transient Analysis of Deterministic and Stochastic Petri Nets", in *Proc. 14th International Conference on Application and Theory of Petri Nets*, Chicago, June 1993
5. Choi, H., Kulkarni, V., Trivedi, K., "Markov Regenerative Stochastic Petri Nets", in *Proc. 16th International Conference on Computer Performance Modeling, Measurement and Evaluation*, pp. 339-358, Rome, Italy, 1993
6. Ciardo, G., German, R., Lindemann, C., "A Characterisation of the Stochastic Process Underlying a Stochastic Petri Net", in *IEEE Transactions on Software Engineering*, Vol. 20, No. 7, July 1994
7. Ciardo, G., Lindemann, C., "Analysis of Deterministic and Stochastic Petri Nets", in *Proc. 5th Int. Workshop on Petri Nets and Performance Models*, pp. 160-169, Toulouse, France, 1993

8. Ciardo, G., Muppala, J., Trivedi, K., "SPNP: Stochastic Petri Net Package", in *Proc. 3rd Int. Workshop on Petri Nets and Performance Models (PNPM'89)*, pp. 142-151, Kyoto, Japan, December 1989
9. Davis, M., "Piecewise-deterministic Markov processes: A general class of non-diffusion stochastic models", *J. R. Stat. Soc.*, *Ser. B*, 46, pp. 353-388, 1984
10. Dugan, J., Trivedi, K., Geist, R., Nicola, V., "Extended Stochastic Petri Nets: Applications and Analysis", in E.Gelenbe, editor, *Performance '84*, pp. 507-519, Elsevier Science Publishers B.V. (North-Holland), Amsterdam, Netherlands, 1985
11. Jerke, N., *"Visual Basic Developer's Guide to E-commerce with ASP and SQL Server"*, Sybex International, 2000
12. Menasce, D. A., Almeida V. A. F., *"Scaling for E-Business: Technologies, Models, Performance and Capacity Planning"*, Prentice Hall PTR, 2000