UDC 621.391.31

ANALYSIS OF PRODUCTIVITY OF DISTRIBUTED SYSTEMS WITH SERVICE ORIENTED ARCHITECTURE UNDER CONDITIONS OF LIMITED LINK AND BUFFER RESOURCES OF TELECOMMUNICATION NETWORK

T.N. KOVALENKO
Kharkov national university of radioelectronics

Проведено аналіз продуктивності розподіленої системи із сервіс-орієнтованою архітектурою та ймовірності втрати запитів в системі SOA, а також залежності цих характеристик від пропускної здатності та обсягу буферів базової телекомунікаційної мережі в області високих рівнів навантаження та в умовах перевантаження.

- -

Analysis of productivity of a distributed system with service-oriented architecture and request loss probability in SOA system is done, as well as dependence of these characteristics on network throughput and buffer size at high load levels and under overload conditions.

- -

Проведен анализ производительности распределенной системы с сервис-ориентированной архитектурой и вероятности потери запросов в системе SOA, а также зависимости данных характеристик от пропускной способности и объема буферов базовой телекоммуникационной сети в области высоких уровней нагрузки и в условиях перегрузки.

Introduction

Implementation of "Everything as a service" (EaaS) concept has become one of the most significant trends in the field of infocommunications during the last several years. This concept implies that different types of resources (such as infrastructure, software, data resources) are provided to subscriber as services that can be available over Internet. The popularity of cloud computing and GRID technologies requires new tasks to be solved by telecommunication systems and networks as they become one of the main components that provide operability of such distributed systems. Most of the platforms providing such offers as Infrastructure as a service (IaaS), Software as a service (SaaS), Data as a service (DaaS), Platform as a Service (PaaS) are organized on the basis of distributed systems with Service Oriented Architecture (SOA). On the other hand, there is a lack of efficient mathematical models and methods for design and optimization of telecommunication systems and networks that provide interaction and data communication between the distributed components of such systems. Existing approaches for analysis of distributed systems, that can be found in works [1-5], pay the main attention to correctness of interaction algorithms when complex operation sequences are executed [3, 4], while telecommunication system (TCS) providing this interaction at the best is represented by a single link with constant delay [1, 5]. As a result evaluation of both productivity of distributed system and quality of provided services, as well as their dependency on characteristics of basic TCS becomes possible only after one of the proposed hardware and software solution was introduced [6]. Most of these engineering solution proposed by well-known manufacturers and vendors such as IBM [7], Hewlett-Packard [8], Sun Microsystems [9, 10], Microsoft [11], are based on Service Oriented Architecture.

I. Overview of basic SOA principles and data communication in SOA systems

SOA concept suggests one of the most efficient approaches to the solving of the application integration problem which is one of the most complex and vital problems in up-to-date information systems. Architecture of SOA systems and the main principles of organization of service interaction in such systems are described in [12-16]. When SOA is used, all system functions are represented as independent distributed components – services, that can be both whole applications and their separate functional modules. These services allow to provide different business-processes communicating over the network in a certain sequence. Services must have standardized interfaces to ensure their reusability for development of new applications as well as modification and enhancement of existing ones. Interaction of services is realized according the principle of "publishing-search-connection" (fig. 1): application that provides a certain service (service provider) places information about it in a service catalogue (repository). Service consumer – application that needs functionality of this service, finds information about it in repository to establish connection with this service and send a request.

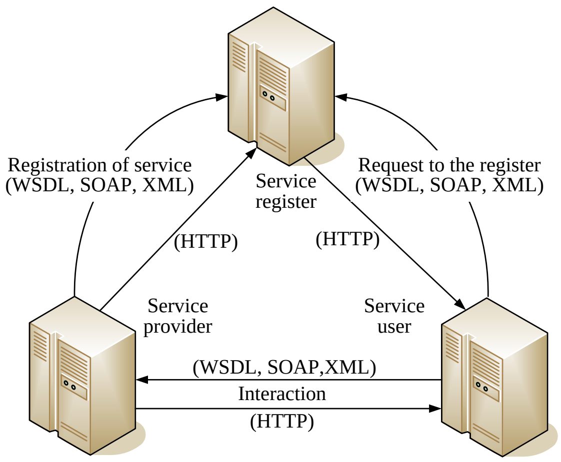


Fig. 1. Principle of "publishing-search-connection" in distributed SOA systems

Currently SOA uses the next basic standards of Web-services for description and communication:

– eXtensible Markup Language (XML) – for data representation;

– Web Services Definition Language (WSDL) – for description of available Web-services;

– Universal Description, Discovery, Integration (UDDI) – for creation of catalogue of Web-services that are accessible over network;

– Simple Object Access Protocol (SOAP) – for data communication.

A special technology known as Enterprise Service Bus (ESB) is used in distributed SOA systems for data transmission, dynamic routing of requests from applications of service consumers and for receiving of responses from applications of service providers as well as for other communication tasks [12, 14]. ESB provides a unified mechanism for transmission of service requests and receiving of service results, for required conversion of messages and transport protocols, for flow control, etc. (fig. 2). This control provides the required sequence of services for realization of a certain business-process.

Thus, characteristics of basic TCS, providing data communication between geographically distributed servers of service providers have a great influence on the characteristics of the whole distributed SOA system and quality of services provided by it [15, 16]. As a result, the task of mathematical modeling of distributed infocommunication SOA systems and analysis of how the boundedness of network resources effects the productivity and other performance characteristics of these systems becomes very important.

II. Modeling of distributed SOA systems with limited network resources using hierarchical timed Coloured Petri Nets

To estimate the influence of TCS characteristics on operability and efficiency of distributed SOA systems appropriate mathematical models and methods are required that are taking into account telecommunication infrastructure of such systems and its functional and structural properties. In this work timed hierarchical Coloured Petri Nets (CPN) were used to model a distributed SOA system and its dynamics.

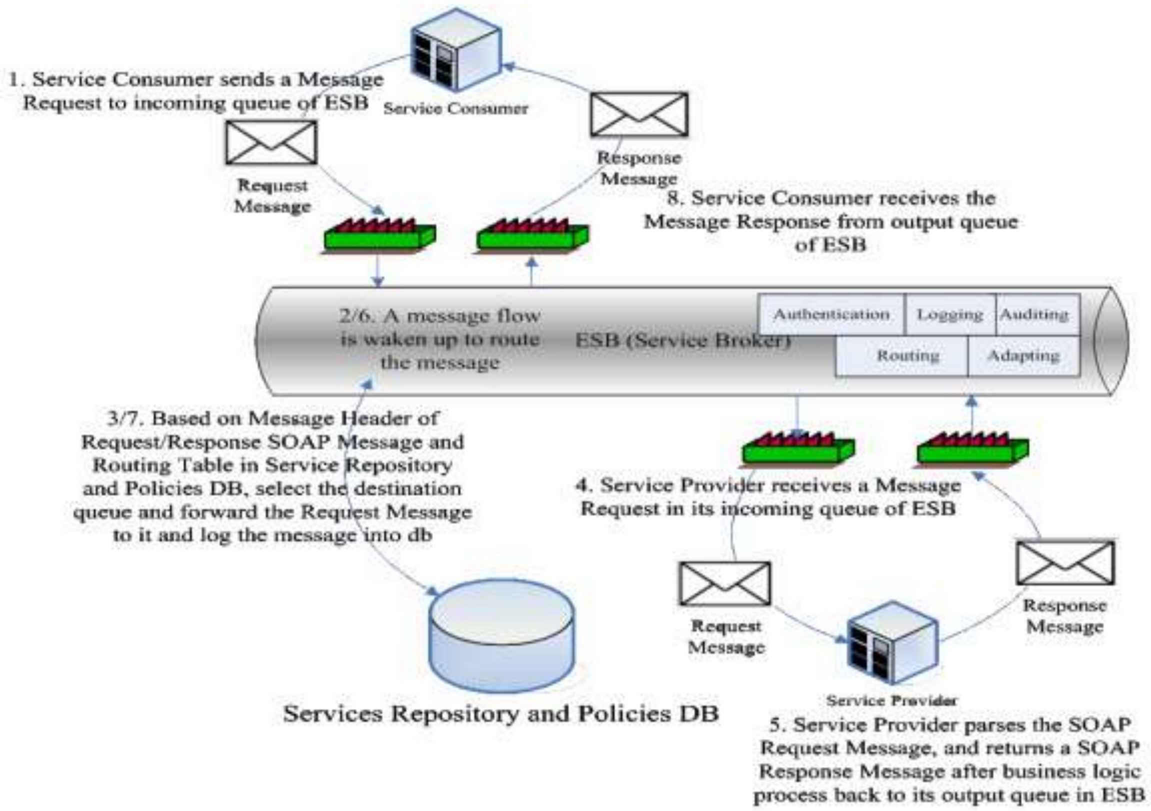


Fig. 2. The role of ESB in a structure of SOA system

CPN is a mathematical tool for modeling and analysis of stochastic dynamic systems and processes [17-19]. Hierarchical CPN consists of several CPN modules integrated in a single model of a complex system using specific places and transitions. Existing modules can be used in a model more than once, moreover new modules can be created on their basis. The hierarchical structure of such a model can be arbitrary complex because each module may consist of several smaller and/or more simple CPN modules, modules of different hierarchical levels can also have common places creating a so called "fusion set".

The top-down approach for system description was used, which involves building a generalized complex system model with progressive specification of models for procedures and processes that require more careful analysis. In this case, any transition of the CPN model can be converted to a substitution transition described with a separate CPN module [19]. The hierarchical structure of the proposed model is shown in fig. 3.

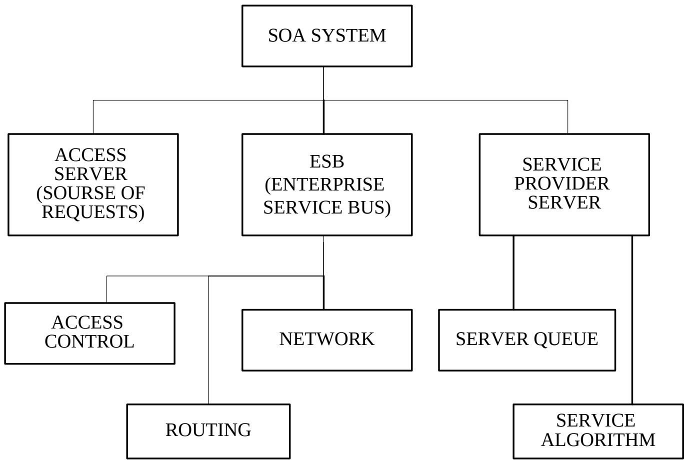


*Fig.*3. Hierarchical structure of CPN model

Top level represents the generalized model of a SOA system that describes its structure and main functional components: access servers, Enterprise Service Bus (ESB) and servers of service providers. Each of these components is described by corresponding CPN module of the next lower level of the hierarchy. The CPN module "Access Server" describes the generation of the input flow of service requests to the SOA system, enabling to set the size and intensity of data blocks. Service provider server also includes two CPN modules describing service discipline in a queue of requests to each server ("Server Queue" module) and the processing of requests for various types of services, taking into account the processing delays ("Service Algorithm" module).

The CPN module "ESB" models the interaction of SOA system components in a distributed environment. It consists of three main modules that implement the functions of the ESB: access control for the input flow of requests to the service bus (limitation of flow, service discipline in a queue of requests, etc.); routing procedures (which can implement, for example, the procedures for optimal network resources allocation) and process of message transfer over the network. The last one is modeled by a CPN module "Network", which enables to set the value of network delay for each transmitted message depending on its length, type, source and destination addresses, etc.

The graphical representation and description of all CPN modules, represented in fig. 1 have been already given in the earlier works [20, 21]. In contrast to approach used in these work the current research takes into account the finiteness of buffer resources both in service provider servers and in telecommunication network. For that purpose two CPN models in the described set of models were modified: these are CPN modules "network" and "server queue". The modified CPN modules of server queue and network are presented in fig. 4 and 5 respectively.

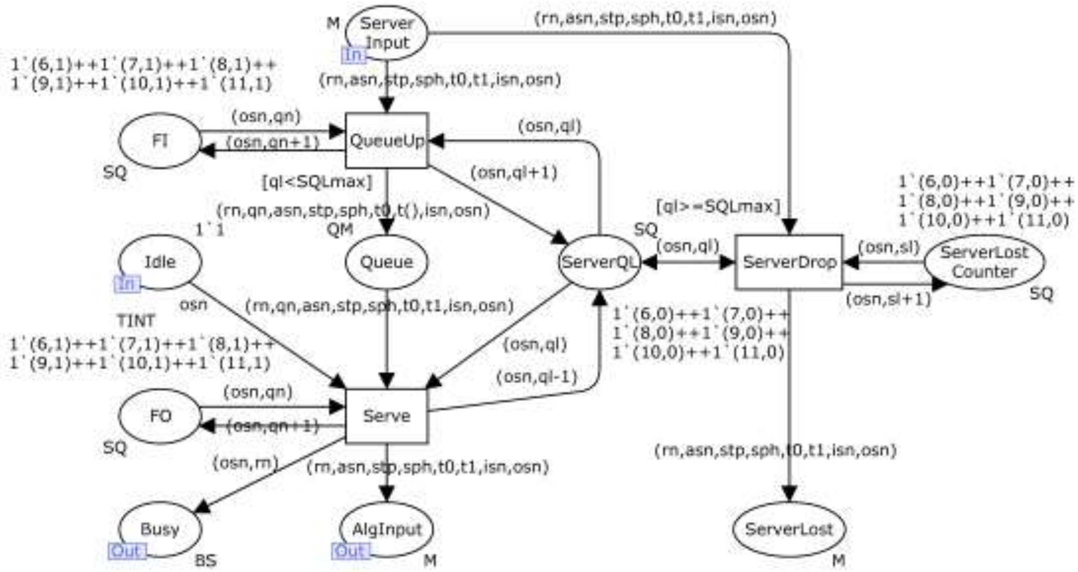


Fig. 4. CPN model of message transmission in a telecommunication network

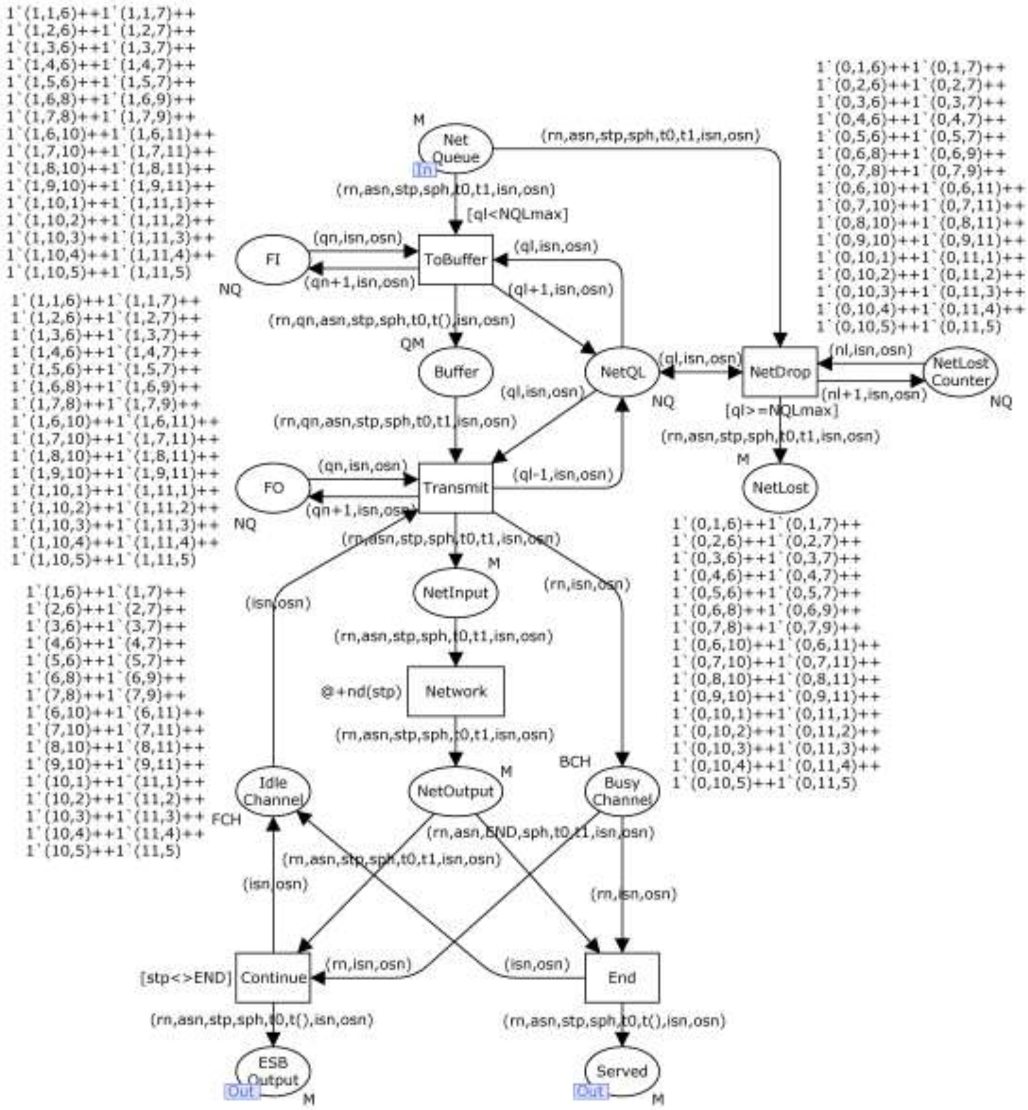


Fig. 5. CPN model of message transmission in a telecommunication network

On the contrary to the model developed in [20] the new model of server queue implies the limited buffer size at the service provider server. The current length of queue to each of the servers is observed in place "ServerQL". If it exceeds the given limit value "SQLmax" the request is dropped (transition "ServerDrop"). Positions "ServerLost" and "Server-LostCounter" are used to collect the statistics about the dropped requests and their quantity.

The main difference of the CPN module "network" from the one that was described in [21] is that the current length of queue to each of the network segments is observed in place "NetQL". If it exceeds the given limit value "NQLmax" the request is dropped (transition "NetDrop"). Positions "NetLost" and "NetLostCounter" are used to collect the statistics about the dropped requests and their quantity.

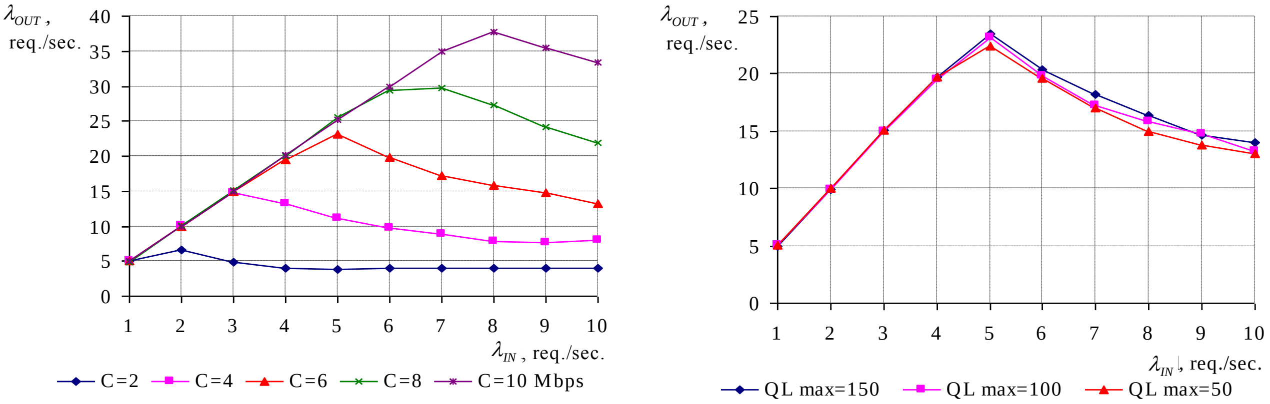
III. Analysis of productivity of distributed SOA system with limited buffer and link resources of telecommunication network

To study the productivity of a distributed SOA system and its dependence on network throughput and buffer size a number of simulation experiments were taken using a set of developed models and a specialized program package "CPN Tools" for executing and performance analysis of Petri Net models. The next characteristics were analyzed during the simulation: intensity of the output flow of served requests, request loses probability as a result of network and server buffers overflow, productivity of SOA system. System performance was considered under high and extremely high load level, the input data for simulation is presented in tab. 1.

Table 1. Input data

The mean value of intensity of input flow of service requests from a single access server, λ_{IN}, req./sec.	1…10
Time interval between consequent requests in the input flow	random, exponentially distributed
The mean value of throughput of telecommunication network connecting the servers of server-providers, C, Mbps	2…10
Time of message transmission over the network	random, exponentially distributed
Message size, Kbytes	125
The mean value of productivity of service providers' servers, m, req./sec.	50
Time of processing of request at server	random, exponentially distributed
Network and server buffer size, messages	50…150

The topology of modeled distributed system is the same as it was considered in earlier works [21, 22]: the system consists of five servers providing access to services and six servers of two alternative service providers, each proposing three analogous set of services. Intensities of input flows of service requests at all of the access servers were considered to be equal to each other, as well as server productivities and values of throughput of telecommunication networks connecting the servers. The result of analysis of output flow are shown in fig. 6.

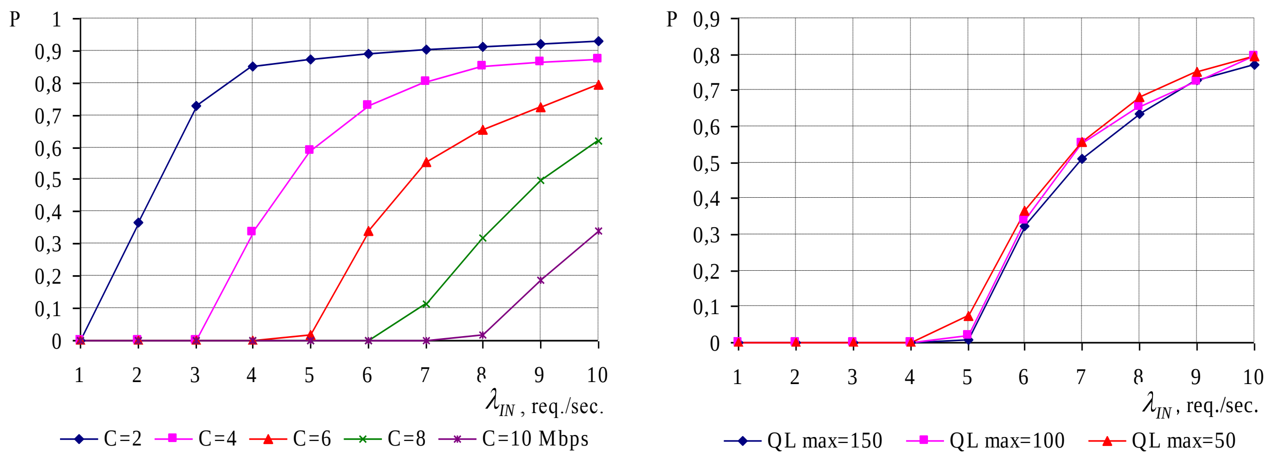


a) b)

Fig. 6. Dependence of output flow intensity on input flow intensity at different values of network throughput (a) and buffer size (b)

As it could be seen from graphics, shown in fig. 6, the dependency of output flow intensity is more significant on network throughput than on buffer size at given values of SOA system characteristics. As intensity of input flow increases the intensity of output flow also increases until a certain limit value of system productivity will be reached. System performance under extremely high load level conditions is characterized by decreasing of output flow intensity while the input flow intensity continues to grow – so one can see that SOA system is overloaded.

The simulation results has shown that at given input data during the overload requests are lost as a result of only network buffer overflow while none of the requests are lost because of server buffer overflow. The results of studying of request loses probability are shown in fig. 7. From graphics depicted in this figure we can conclude that request loses probability depends on network throughput much more greater than on buffer size at given input data.



a) b)

Fig. 7. Dependence of request loses probability on input flow intensity at different values of network throughput (a) and buffer size (b)

The histogram showing dependence of productivity of distributed SOA system under consideration on network throughput and buffer size is shown in fig. 8. The system productivity was determined as the maximum value of intensity of the output flow of served requests at given network throughput and buffer size. As one can see from the results, presented in fig. 8, the dependency of SOA system productivity on buffer size at given input data is negligible as compared with dependency of system productivity on network throughput.

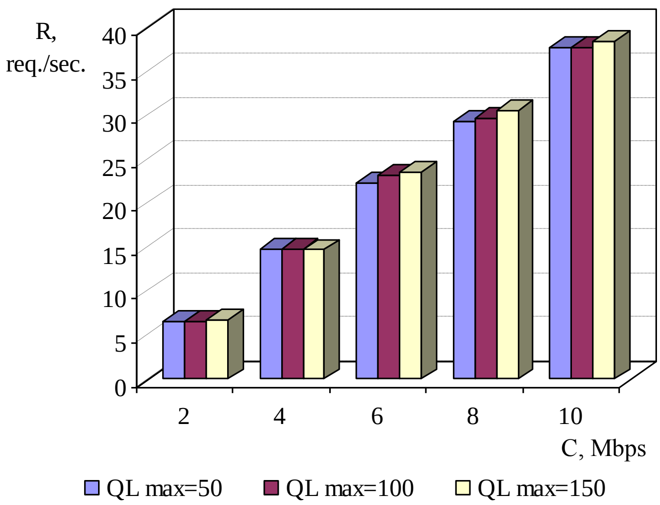


Fig. 8. Productivity of the SOA system

Conclusion

The work presents the results of studying of distributed SOA system performance when link and buffer resources of its basic telecommunication network providing data communication in the system are restricted. For that purpose a set of models simulating a distributed SOA system operation were used. The models were built on the basis of Coloured Petri Nets. The hierarchical approach proposed in the earlier works was used, but modules representing the network and server resources were modified such that to allow taking into account the limited size of network and server buffers.

Analysis of SOA system productivity and request loss probability is done, as well as research of dependence of these characteristics on network throughput and buffer size. The results received in simulation experiments have shown that at given range of values of input data the most influence on request loss probability and productivity of distributed SOA system is made by network throughput, while increase of buffer size from 50 to 150 messages causes very insignificant refinement of studied characteristics. The diagrams received in this research also can be used to determine the limit value of intensity of input flow of service requests exceeding of which leads to the system overload resulting in decreasing of intensity of output flow of served requests and sharp growth of request loss probability.

The trend for further researches covers study of SOA system performance under self-similar input traffic research, analysis of influence of link failure in a basic telecommunication network on characteristics of distributed infocommunication systems with service-oriented architecture, as well as development and efficiency analysis of methods of traffic and network resources control and methods of overload prediction and avoidance in SOA systems.

Reference:

1. *Топорков В.В.* Модели распределенных вычислений. — М.: Физматлит, 2004. — 320 с.
2. *Lee E.A., Sangiovanni-Vincentelli A.* A framework for comparing models of computations // IEEE Trans. On CAD of Integrated Circuits and Systems. — 1998. — Vol.17, No.12. — P. 1217 – 1229.

3. *Воеводин В.В., Воеводин Вл.В.* Параллельные вычисления. — СПб.: БХВ-Петербург, 2002. — 608 с.

4. *Корнеев В.В.* Параллельные вычислительные системы. — М.: Нолидж, 1999. — 320 с.

5. *Banner Th., Ernst R., Osterlinng A.* Scalable performance scheduling for hardware-software cosynthesis // European Design Automation Conf.: Proc. — Brighton, UK: IEEE CS Press, 1995. — P. 164 – 169.

6. *Хойссер М.* Тестирование SOA: примеры [Электронный ресурс] // Директор информационной службы. — 2008. — № 12. — Режим доступа: http://www.osp.ru/cio/2008/12/5585238/.

7. Working smarter with a Smart SOA foundation [Электронный ресурс]. — Режим доступа: ftp://public.dhe.ibm.com/software/solutions/pdfs/7946_Working_smarter_Brochure.pdf.

8. HP SOA Systinet and JBoss Enterprise SOA Platform. Service-oriented architecture infrastructure and governance solution [Электронный ресурс]. — Режим доступа: http://h20195.www2.hp.com/V2/GetPDF.aspx/4AA2-6915ENW.pdf.

9. *Ed Ort.* Service-Oriented Architecture and Web Services: Concepts, Technologies, and Tools [Электронный ресурс]. — Режим доступа: http://java.sun.com/developer/technicalArticles/WebServices/soa2/SOATerms.html.

10. Oracle White Paper — Oracle SOA Suite 11*g* [Электронный ресурс]. — Режим доступа: http://www.oracle.com/us/technologies/soa/soa-suite/oracle-soa-suite-11g-198758.pdf.

11. Microsoft and SOA Software [Электронный ресурс]. — Режим доступа: http://www.soa.com/solutions/microsoft/microsoft_and_soa_software/.

12. *Пузыня С., Пырлина И.* Корпоративная архитектура и SOA [Электронный ресурс] // Открытые системы. — 2010. — №2. — Режим доступа: http://www.osp.ru/os/2010/02/13001450/.

13. *Nicolai M. Josuttis*. SOA in Practice: The Art of Distributed System Design (Theory in Practice). — O'Reilly Media, 2007. — 352 p.

14. *Дубова Н.* SOA: подходы к реализации [Электронный ресурс] // Открытые системы. — 2004. — № 6. — Режим доступа: http://www.osp.ru/os/2004/06/184450/.

15. *Сергей Кузнецов*. SOA с гарантией качества [Электронный ресурс] // Открытые системы. — 2008 — № 7. — Режим доступа: http://www.osp.ru/os/2008/07/5479587/.

16. *Брент Слипер*. 5 проблем SOA [Электронный ресурс] // Computerworld Россия. — 2004. — № 38. — Режим доступа: http://www.osp.ru/cw/2004/38/81416/.

17. *Popovskij V., Barkalov A., Titarenko L.* Control and Adaptation in Telecommunication Systems: Mathematical Foundations. — Berlin, Heidelberg: Springer Berlin Heidelberg, 2011. — 564 p.

18. Математичні основи теорії телекомунікаційних систем / В.В. Поповський, С.О. Сабурова, В.Ф. Олійник, Ю.І. Лосєв, Д.В. Агеєв та ін. За загальною редакцією В.В. Поповського. — Харків: ТОВ «Компанія СМІТ», 2006. — 564 с.

19. *Jensen K.* Coloured Petri nets: basic concepts, analysis methods and practical use. — Berlin: Heidelberg, New York: Springer-Verlag, 1996. — 234 p.

20. *Коваленко Т.Н.* Методика моделирования процессов сетевого взаимодействия в системах с сервис-ориентированной архитектурой // Радиотехника. — Вып.163. — С. 66 – 72.

21. *Коваленко Т.Н.* Исследование влияния пропускной способности телекоммуникационной сети на задержку предоставления сервисов в распределенных системах SOA [Електронний ресурс] / Т.Н. Коваленко, С.А. Макаров // Проблеми телекомунікацій. – 2011. – № 3 (5). – С. 78 – 88. – Режим доступу до журн.: http://pt.journal.kh.ua/2011/3/1/113_kovalenko_soa.pdf.

22. *Kovalenko T.* Optimization of network resources distribution in SOA systems // Journal of EcoAgriTourism. — 2010. — Vol. 6, № 2(19). — P. 164 – 167.