Internet of Things (IoT) and Engineering Applications (2017) 2: 1-13
Clausius Scientific Press, Canada

The Structural-Functional Synthesis of IoT Service Delivery Systems by Performance and Availability Criteria

Ivan Demydov[1,a,*], Yulia Klymash[1,b], Mykola Brych[1,c] and Mykhailo Klymash[1,d]

[1]Department of Telecommunications, Lviv Polytechnic National University, Lviv, Ukraine
a. ivan.demydov@gmail.com, b. yulia20122012@ukr.net, c. brych@ua.fm, d. mklimash@lp.edu.ua
**corresponding author*

Keywords: SDP, structural-functional Synthesis, cloud computing, distributed service platforms.

Abstract: This paper is devoted to the quantitative investigation of the availability of cloud service systems. In this paper we calculate the criteria and constraints of a distributed service platform such as availability and system performance index variations by defined set of the main parameters. We analyze the calculation results to enable optimal synthesis of distributed service platforms based on the cloud service-oriented architecture. The method of synthesis has been numerically generalized considering the type of service workload. We used Hurst parameter to statistically evaluate each integrated service that requires implementation within the service delivery platform. The latter is synthesized by structural matching of virtual machines using combination of elementary service components. According to Amdahl's Law the clustering of cloud-networks allows to break the complex dynamic network into separate segments that simplifies access to the resources of virtual machines. This in turn simplifies complex topological structure, enhancing the overall system performance. The proposed approaches and obtained results allow to numerically justify and algorithmically describe the process of structural and functional synthesis of efficient distributed service platforms. These platforms through dynamic configuring and exploitation provide an opportunity to create the dynamic environment in terms of comprehensive services range and significant user workload fluctuation.

Introduction

The structural-functional integrity of modern cloud networking paradigm is very important for building scalable and reliable commercial infrastructures according to Service-Oriented Architecture (SOA). This architectural concept is used for wide set of applications in order to ensure their efficiency in a concurrent world of e-business, e-commerce, personal communications and other activities [1 ,2]. Despite that, the mentioned networking concepts started conquering the market just several years ago. However, such network solutions are commonly characterized by extremely complex design and high commercial value. Thus, we propose analytical method for synthesis of structural and functional parameters of generalized service delivery platform (SDP). The method takes a set of constraints as an input.

Today, cloud-computing services are widely spread across the information technology and telecommunication market. They help to make business workflows more effective and scalable [2]. The key players of these markets are Microsoft (Microsoft Azure), Google (Google Apps Engine), Amazon (Elastic CloudComputing, Simple Storage Service), IBM (Blue Cloud), Nimbus, Oracle etc. Some small companies also have their own cloud computing services. There are multiple free of charge solutions available at the market, e.g., iCloud, Cloudo, FreeZoho, SalesForce etc. All these solutions are different in terms of offered services. Among the typical services are SaaS (Software as a Service), PaaS (Platform as a Service), IaaS (Infrastructure as a Service) and HaaS (Hardware as a Service).

Despite the variety of services, there are quite typical hardware and software facilities used as the basis of most of could systems. They are facilitating an operation of the system that is built according to SOA. Usually such system consists of a set of virtualized service nodes or virtual machines (VM). Scalability and flexibility are achieved through VM replication and migration. Such dynamic environments are commonly very unreliable in terms of failure probability of hardware or software components. There are several principles that allow to minimize this probability and to increase the recovering speed of a cloud system. Most of them are based on distributed data processing (reserving, re-distribution of computing resources etc.) and are invisible for consumers, who perceive system availability as no-faulty operation. The statistics of typical failures in cloud systems is very interesting (see Table 1) [3, 4]. It shows that existing approaches to achieve high-reliability of cloud systems are poor in terms of effectiveness.

Table 1. The Typical Failure Statistics of Cloud-Systems.

N	Service provider name	Services affected	Dates	Unavailability period
1	Google	Gmail, Google Apps Engine	24.02.2009	2,5 hrs
2	Google	Google Search	31.01.2009	40 min
3	Google	Google Gmail	9.03.2009	22 hrs
4	Amazon	Amazon Elastic Cloud	11.06.2009	7 hrs
5	Amazon	Amazon Elastic Cloud	9.12.2009	5 hrs
6	Amazon	Amazon Simple Storage	15.02.2008	2 hrs
7	Amazon	Amazon Elastic Cloud	21.04.2011	27 hrs
8	Microsoft	Microsoft Azure	13-14.03.2008	22 hrs
9	Microsoft	Microsoft Hotmail	12.03.2009	5 hrs
10	Microsoft	Microsoft Sidekick	4.10.2009	144 hrs
11	Flexiant	FlexiScale	31.10.2008	18 hrs

Deeper analysis showed that cloud system unavailability is not the single consequence of a failure. In the case of Microsoft Sidekick failure, all personal data of users was lost [3] and was restored only partially.

Despite the wide set of solutions [5, 6] for ensuring availability of service systems, existing cloud systems are constantly encountering bottlenecks. Thus, ensuring high system availability is the high-priority task.

1. The Synthesis Criteria and System Parameters

The simplest network structure (topology) is undirected (oriented) graph G with a set of vertices V and set of edges (arcs) E which corresponds to nodes and lines. The simplest model of structural reliability of an information service system is a random graph $(G; p)$ with $p = \{p(\varepsilon); \varepsilon \in E\}$. It could be characterized by independent removal of edges of G (arcs) $\varepsilon \in E$ with probability $q(\varepsilon) = 1 - p(\varepsilon)$.

In a service system, the availability is tightly coupled with survivability of a set of VM and can characterized by system's ability to quickly and easily recover from a failure the normal operating mode. However, this concept can be described as the ability of the system to reliably operate for a long time with maximal efficiency. The concepts of service availability and service survivability are interrelated in the theory of complex systems (e.g., cloud networks).

The most important component of cloud system reliability is the availability it which describes the ability of the server system to survive continuously in the given conditions and during the given interval of operation. It can be calculated using survivability parameter of a distributed set of VMs. These VMs can form various combinations of ESC.

The properties of ESC combinations are affecting the service availability and overall system performance index.

To assess these parameters in the cloud system we should clearly define the notion of dynamic network topology. In our model, we evaluate the survivability of structures in terms of the probability that two segments will be interconnected in the near future, i.e., there will be at least one edge. This edge is a "key link" to connect these segments. On the other hand, there must be a working ESC, which is not overloaded and is able to process the given flow of requests. These probabilities are affected by the probability of failure of a certain path in the intermediate segments of the server system. For instance, a router between subscribers or VMs. Thus, at any given moment of time, ESC availability depends on the probability of requests blocking in the intermediate segments.

In paper [7], authors understand information systems vitality as ability of a system to perform its basic functions under the impact of outer factors (at least within tolerable loss of quality of service). This definition is similar to the definition given in [8]. In [9] the information systems vitality is defined as ability to perform a given task under the deleterious effects on the entire system or its individual components, keeping operational performance within acceptable limits. These two definitions focus on the following key points. The first, the vitality should be considered as an intrinsic property of the system as it doesn't depend on operating conditions that arise at any given moment of time. It possesses this property all the time and to some extent the property could occur under normal operating conditions, where there are failures that are caused by manufacturing defects, degradation, maintaining etc. The survivability can be observed under the large external influences that are not expected for normal operation mode and can lead to extreme operating conditions. The second, the system supports not all the functions that it should perform during normal operation mode. It supports only the basic functions that sometimes leads to QoS degradation. This means that we should replace the strategy of decreasing the severity of adverse effects.

In the studies about survivability and availability, we identified a number of areas (approaches) where several types of analysis can be used, e.g., game-theoretical [10, 11], probabilistic [8], deterministic [12, 13], graph [14, 15]. Probabilistic and deterministic approaches are the most accommodated for technical purposes. The main ideas of these models were outlined in [15].

The probabilistic methods of survivability investigation are based on assumption that location and time of occurrence of adverse (harmful) effects (HE) can be described using uniform distribution within a single system HE.

The deterministic methods of survivability investigation are based on matching of specific types of harmful factors and resistance of system elements. These approaches cab be divided in static and dynamic. The static approach is based on definitions of object's weak region and on the level of damaging factors. In the next step, the list of items that might be damaged is determined. The level of the system operation quality is determined using the logical functions. A dynamic approach is based on the use of simulation models, including dynamic models: the emergence and development of HE; development of HE factors that affect the state of the elements of the object; object operating in terms of structural and parametric changes induced by damaging factors and by countermeasures to HE.

In turn, the graph models are characterized by simplicity. Traditionally, they are used when investigating the structural survivability that goes along with the concept of "destruction". The system that is represented by a graph can be considered as destroyed, if after removing of the vertices the graph is valid and satisfies one or more of the following conditions:

- graph contains at least two components;
- there are no directed paths for a given set of vertices;
- the number of vertices in the largest component of *G* is less than some given number;
- the shortest way is longer than certain given value.

Accordingly, a system is considered as survivable, and a service system is considered as available in the absence of these conditions. A task of optimal parametric synthesis of cloud SDP can be solved by optimal choice of the designed system parameters for each declared complex service. In [16] we define this as:

$$x_{opt} = (\alpha, n, H) \tag{1}$$

should maximize the following equations and is used to define the synthesis criteria:

$$x_{opt} = \arg\max P_A \left\{ X\left(x_{opt}, t\right) \in D_x, \forall t \in [0, T] \right\}, \tag{2}$$

$$x_{opt} = \underset{x_{opt} \in D_x}{\arg\max}\, S_p\left(x_{opt}\right), \tag{3}$$

where: $X\left(x_{opt}, t\right)$ is a probabilistic process of parameters x_{opt} changing, $P_A(\bullet)$ is a service availability stochastic function and $S_p(x) = \left(\alpha + (1-\alpha)/n\right)^{-1}$, is a structural performance function. Here D_x is a tolerance region for x_{opt} parameters, T is an exploration time for current SDP realization.

Let us define the tolerance region:

$$D_x = \left\{ x \in R^3 : P_{A\min} \le P_A(x) \le 1 \right\}, \tag{4}$$

where $P_{A\min}$ is the minimum acceptable service availability within designed SDP.

The solution for task (1) is based on the analysis of the interrelations of the service availability and ESC parameters, as well as workload traffic statistical parameters (Hurst parameter). 1st and 2nd statistical moments for workload traffic served by respective service are also needed to examine all necessary stochastic characteristics. These parameters could be easily obtained after statistical simulation of the workflow intensity with necessary *H* parameter [17]. A set of internal parameters

$x_{opt} = (\alpha, n, H)$ could be represented as a point inside R^3 cube, and the space of allowed x_{opt} parameters is limited by D_x tolerance region.

2. Criteria Calculation

Given internal SDP parameters we could define [15]:

$$P_A(\bullet) = \frac{1}{N_0} \sum_{i=1}^{N_0} P_i(x,t)(N_0 - i), \tag{5}$$

where $N_0 = N(N-1)/2$, N is a number of SDP service nodes with organized VM, that aggregating respective ESCs. There is no trivial solution was found and the common task was splitted to the parts using additive survivability definition [15]:

$$P_i(x,t) = \sum_{i=0}^{\alpha} G_i^{E2}(x,t) + \sum_{i=\alpha+1}^{n} G_i^{E1}(x,t) + \sum_{i=n+1}^{N} G_i^{P}(x,t), \tag{6}$$

respectively using [18] and Erlang process of *i*-th order definition:

$$G_i^{E1}(x,t) = \frac{\left(i\Lambda_i^{E1}(x)\right)^i}{(i-1)!} t^{i-1} e^{-i\Lambda_i^{E1}(x)t}. \tag{7}$$

Thus,

$$\Lambda_i^{E1}(x) = \frac{h_1(x)\pi_1}{i(n-\alpha)}, \tag{8}$$

and $h_1 = \lambda p_1(x)$, hereinafter π_1 is an average physical availability of VM at each parallelized by ESC combination SDP service node. According to the transformation of Norros equation in [17]:

$$p_1(x) = e^{-\left(\frac{(1-H)^{2H} H^{-2H}\left(2^{-\frac{1}{2H}}\left(c_v(n-\alpha)^{\frac{1}{2}}\right)^{-H/2} B^{\frac{1}{H}-1}\left(C-\lambda/(n-\alpha)\right)\left(\frac{\lambda}{n-\alpha}\right)^{-H/2}\right)^{2H}}{(H-1)^2}\right)}, \tag{9}$$

where *B* is a buffer utilization ratio at the moment of time *t*, *H* is the Hurst parameter of respective traffic type, that applicable for examined complex service, *C* is the average throughput capacity of the ESC, λ is a traffic intensity, c_v is a variation coefficient of the incoming traffic workload, both indexes are for examined complex service. Consequently, using Erlang process of *i*-th order definition:

$$G_i^{E2}(x,t) = \frac{\left(i\Lambda_i^{E2}(x)\right)^i}{(i-1)!} t^{i-1} e^{-i\Lambda_i^{E2}(x)t}, \tag{10}$$

where

$$\Lambda_i^{E2}(x) = \frac{h_2(x)\pi_2}{i}, \tag{11}$$

and $h_2 = \lambda p_2(x)$, hereinafter π_2 is an average physical availability of VM at each sequential by ESC combination SDP service node. According to the transformation of Norros equation [17]:

$$p_2\left(x\right)=e^{-\left(\frac{(1-H)^{2H}H^{-2H}\left(2^{-\frac{1}{2H}}c_v{}^{-H/2}B^{\frac{1}{H}-1}(C-\lambda)\lambda^{-H/2}\right)^{2H}}{(H-1)^2}\right)}, \tag{12}$$

and respectively using Poisson process definition let's define [18]:

$$G_i^P\left(x,t\right)=\frac{\left(\pi_3\lambda p_2\left(x\right)\right)^i t^i}{i!}e^{-\pi_3\lambda p_2(x)t}, \tag{13}$$

where π_3 is an average physical availability of VM at each transport SDP service node.

Using (5-13) we define the statistical distribution for ESC combination and respectively service availability criterion (2) calculation on the each synthesized VM structure of cloud SDP. Using the same set of parameters (1) that was used in (2-3), we simultaneously define structural performance by the second synthesis criterion (3) after Amdahl's Law (Fig.1) [19].

3. Conceptual Study of Targeted Technologies

Implementation of modern broadband network services, and IoT (Internet of Things) concept in particular drastically changes the view on services and service network systems' infrastructure. In these systems, all network addressing functions and stream management are delegated to cloud environment, removing the need for local networking equipment if it is not either software configured (SDN) or doesn't provide a direct connection to communication networks. The given category of information and communication systems (CBN) were developed at Petrino and Aryaka companies' \cite{20demydov2016method}. Network-as-a-Service (NaaS) was proposed for generalization of mentioned CBN properties [4]. Approaches to CEN system deployment are investigated in papers [21-24], particularly the optimization of corresponding network equipment performance. SDN concepts of network functions virtualization, as well as service network infrastructure elements, are a basis for description and research of CBN service systems [25]. Structural and parametric researches of cloud service systems are performed in papers [1, 22, 23]. The authors of the researches obtained the analytical dependencies of the main functional parameters, alongside with quality of service dependencies (jitter, packet delay, system throughput, packet loss ratio). To improve quality of service by increasing the availability indicators of cloud platform telecommunication nodes based on [26], we propose the method of dynamic routing metrics correction. We simulated some service network systems with scaling using breakthrough structural tracing. The numerical availability of cloud service systems must be assessed in order to perform an effective synthesis of service delivery system using the service availability criteria (which includes reliability, robustness and QoS). Results and functional dependencies (1-13), obtained in [27] were used for the modelling of the above-mentioned characteristics in this paper. Overlapping of the results obtained in this paper with the results mentioned earlier, allows us to assess the alternative cloud service system segments deployment approaches with the required parameters for further effective processing of their development and modification strategies.

Cloud service systems had obviously become very popular around the market nowadays, making electronic business more effective and scalable [2]. The most famous are the solutions from Microsoft (Microsoft Azure), Google (Google Apps Engine), Amazon (Elastic Cloud Computing, Simple Storage Service), IBM (Blue Cloud), Nimbus, Oracle and so on. Relatively small operators provide cloud computing services alongside large enterprise cloud systems. There also exist solutions which free of charge such as iCloud, Cloudo, FreeZoho, SalesForce etc. They vary in the list of services that are offered, as well as the type of service they deliver: SaaS (Software-as-a-

Service), PaaS (Platform-as-a-Service), and IaaS (Infrastructure-as-a-Service). For the case of finished infrastructure transformation from CEN to CBN paradigm HaaS (Hardware-as-a-Service) is also worth mentioning. Despite the service variety (which are generally called XaaS), there exist a few typical software and hardware tools, which are often used as basis for cloud systems deployment. They provide system functionality, which is based on SOA (Service-Oriented Architecture). They are implemented as a set of virtualized service nodes or virtual machines that are replicated in order to sustain scalability and support some set of electronic services with flexibility and according to consumers' needs. Hardware and software tools of cloud computing platform may sometimes operate unstable or unreliable due to it's imperfection or degradation in some statistical indicators. In order to minimize such possibility and decrease the recovery time in cloud computing environment it is necessary to apply special principles, the majority of which are specific for distributed data processing (redundancy, parallelization, calculation resources reallocation etc.).

Described approaches are aimed at partial concealment of the real system availability for the purpose of creating the illusion of trouble-free operation of a distributed service platform. Despite that, the statistics of cloud service platforms failures are available [3, 27]. It shows that in some situations solutions applied in SDP for obtaining high service availability become ineffective. Analyzing deeper, it is worth mentioning that system inaccessibility may not be the only consequence of cloud systems' failures. In case of Microsoft Sidekick failure, users' personal data was lost and was not fully recovered [3, 27]. Despite the high level of well-known methods for increasing availability of service systems, cloud systems are still a subject for bottleneck analysis. The main purpose is to increase their operation robustness, service availability and overall performance [6, 28]. The relevance of research on the subject is rather high.

4. Service Availability Modelling in Scalable Cloud Service Networks

Structural and functional integrity of modern cloud computing paradigm is important for scalable and reliable service platform deployment using SOA. Many applications involve this architectural concept to become more effective in a world, where most business processes are parallel: electronic business, e-commerce, personal communications and so on [1, 2] and such network concepts become more and more widespread by the year. Due to the high complexity of designing and high commercial value of network solutions we developed and analytical synthesis method [27] for the purpose of optimization structural and functional parameters with given constraints for typical cloud service delivery platforms (SDP).

To perform a simulation of service availability in scalable cloud service networks, we define "structural parameters" as quantitative indicators for basic service components, which are somehow configured in virtual machines structure on cloud telecomm platform nodes. The structures of network connections in physical network topology are defined unclearly, which is inherent for cloud systems, separating them from traditional network architectures. In general, topological configuration of virtual machines is dynamic, same as the configuration of offered and required service array. Virtual machines migrate and replicate elementary services, as the components of complex applications according to subscribers' demands distribution. That is, service-oriented architecture owns a totality of migrating resources inside a cloud system, which is extremely complex distributed object, forming a specific implementation of SDP.

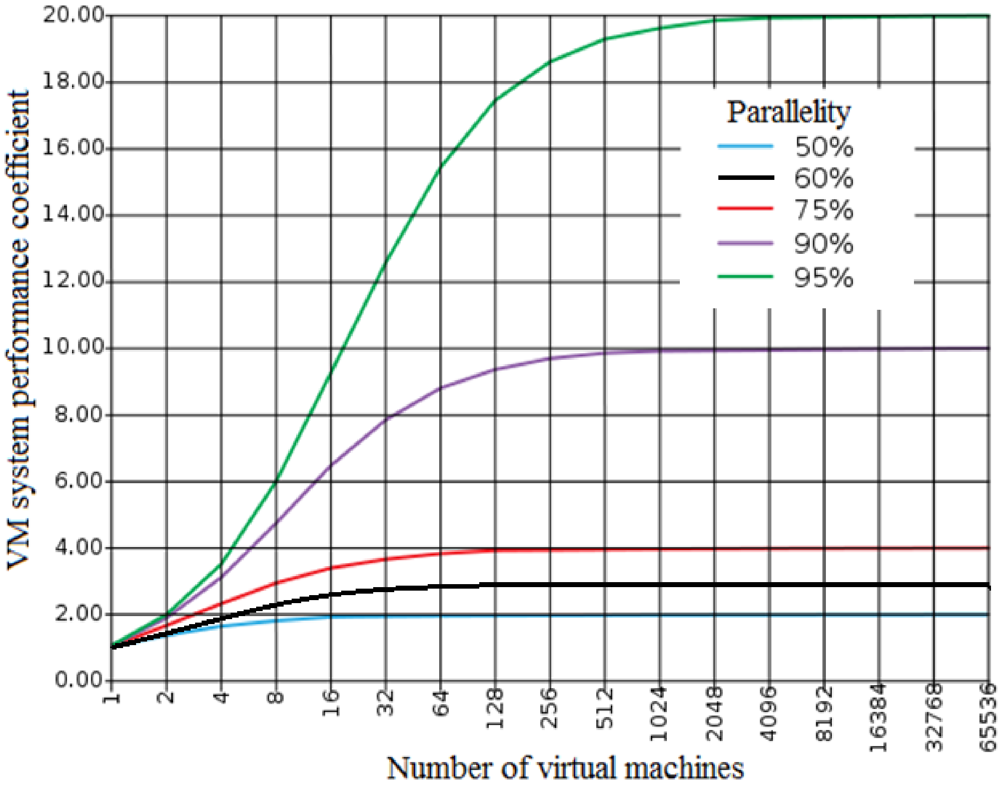

Figure 1. VM systems' performance coefficient after Amdahl's Law [19].

We can outline the specific service components groups, which are used in the orchestration process, by which complex application may be used by SDP users. Service components classification, as threads or streams, which are performed by virtual machines may be adopted in accordance with the definitions of Amdahl's Law [4, 19] (Fig. 1). Thus, for our integral model we can outline hypervisors and other successive elementary service components (ESC) of service applications set α into one group, and also ESCs which serve queues in parallel – into another $\eta=n-\alpha$ (see Fig. 2). Here n depicts a total number of ESCs [27], as was mentioned in previous chapter.

This way, taking unclearly (fuzzy) specified and dynamic cloud network system structure into account, a task of optimal structural and functional synthesis can be reduced to selection of optimal service components quantities, which belong to different designated groups within their combined mix during the formation and building up of complex applications in a service-oriented architecture. Unfortunately, the main difficulties associated with this task come with a lack of research about stochastic traffic serving processes by distributed service applications in system aspect under the conditions of SDP load variation, which is generated by user requests to different service types [5, 17]. Service functional properties should be considered in terms of stochastic processes, given their direct dependence on the statistical properties of the load served. To characterize the statistical features of the traffic load on the cloud system self-similarity Hurst parameter can be used. Accordingly, in [16] we defined the specific features for the following types of traffic: VoIP, VoD, IPTV Multicast, Web-data etc.

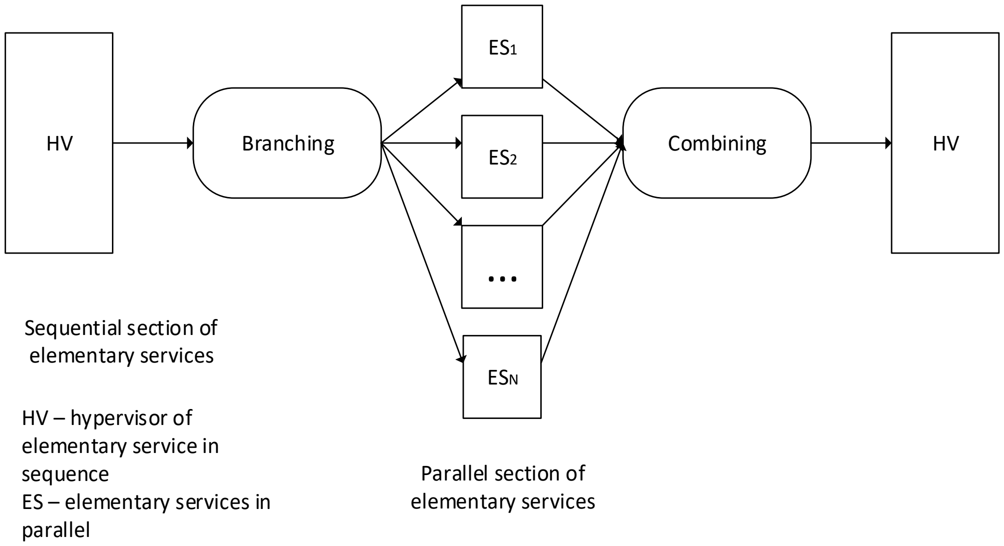

Figure 2. Parallel and sequential sections of elementary services of a virtual structure in cloud SOA [27].

Let us define the concept of "functional parameter" for a corresponding service type, which should be served in SOA-based cloud platform, as statistically predefined and calculated Hurst parameter, which corresponds to the specific type of traffic for this service. Thus, for each synthesized instance of cloud architecture the corresponding indicator of generalized service availability can be presented and calculated for each functional service type, which SDP can offer, same as relative performance indicator (Fig. 1) [27] for a given combination of ESCs. We chose both indicators as criteria for optimal structural and functional synthesis of service delivery platforms [27].

Next, we present the results of service availability modelling in scalable cloud service network, which was performed using MATLAB and Mathcad systems based on numerical calculations and analytical dependencies approximation, which were presented in [4, 29] (Figs. 3, 4). We assume, that for the modelling results depicted in Fig. 3 a, cloud platform service layer implementation is performed without optimal distribution of service flows (virtual machines migration).

The case, where service flows may be adaptively corrected on demand by virtual machines migration [30] is depicted in Fig. 3 b. In case of non-optimal cloud network platform resources configuration the necessary resources exceed the available resources of virtual machines, while in case of adaptive cloud network platform resources configuration, the comprehensive resource roughly equivalents the available virtual machine requirements. Figs. 2, 3 (a, b) list various load parameters Λ and the values of Hurst parameter H for aggregated inbound traffic. P_a is a generalized service availability of cloud network platform; η is a fraction of ESCs, which process the queues in parallel (%); N_a is a number of service nodes in cloud network platform.

Conclusion

In this paper, we investigated service availability of distributed service platform. Based on the obtained results, we performed an effective synthesis of configuration of scalable cloud service system and considered load type and volume. The workload requires processing in terms of corresponding service development using different combinations of parallel and successive ESCs in order to achieve the best indices of performance and system efficiency. Based on the obtained results, we draw a conclusion about gradual reduction of service availability in cloud network system during the load increase. Moreover, the increasing rate is slower in system instances, where

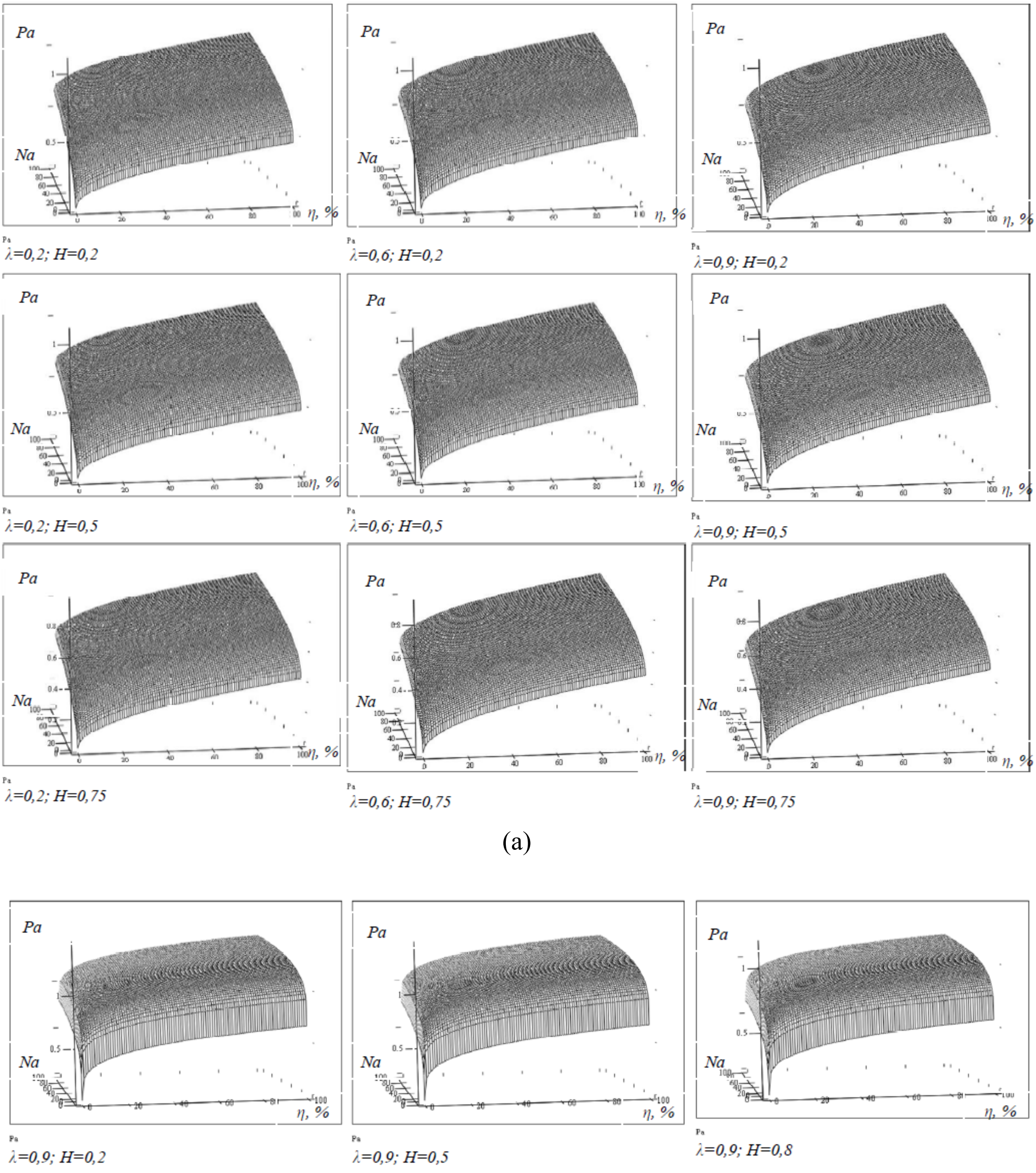

Figure 3. Service availability modelling results in scalable cloud service network for non-optimal case - a, and for adaptive case - b.

fraction of the parallel ESCs, which serve the queues, prevails the successive fraction. With the bigger number of service nodes in a cloud platform and parallel growth of service components, the service availability increases faster. According to the indexes, obtained during the simulation, we assume that an optimal η/α ratio would be *60/40*. The impact score of Hurst parameter to service availability manifested in its high values, showing that self-similar traffic significantly reduces the latter. Therefore, we can see that in the case of resources shortage as for sub-optimal cloud service platform configuration, for the self-similar traffic the reduction of availability index is possible by an average of *15%* for $\lambda=0.2$ and up to *30%* for $\lambda=0.9$. In the context of adaptive cloud platform resources configuration, the use of virtual machine migration will lead to the equal reduction of service availability and will not exceed *5-10%*, and with the service nodes count above *40* it becomes insignificant. Note, that high levels of self-similarity are intrinsic to the internet traffic generation services – H~0.685, IP-telephony – H~0.981, Video-on-Demand (VoD) – H~0.608, service data transmission – H~0.719 [28].

Taking Amdahl’s law into account, we claim that cloud system clustering allows to achieve the maximal effectiveness for cluster size not less than 100 nodes, which comes out from the results of service availability simulation. However, increase of the size of cloud platform cluster will provoke a necessity to modify thread management methods, in particular, the use of modified methods for improving the availability of telecommunications nodes.

In overall the contributions of the paper are following:

a. New structure of the software code for SDP handlers with mutual optimization of the componential (modular) operational availability and performance gain of the parallelized program components at the SaaS were proposed. They benefit by balancing the consequent and concurrent logic of data workload handling processes over the under layered platform, as a service.

b. PaaS system configuration based on optimization of its structural and parametric indexes by the criteria of system productivity and service availability. This configuration is achieving after clustering of concurrently operating virtual machines set undergoing their migration process into the service network platform.

c. IaaS network architecture principles by the criteria of the network interfaces availability of the infrastructural telecommunication systems and their performance index. It means composition of concurrent (ex. after virtualization) elements of an active network equipment and consequent passive network elements, that’s appropriate for traffic streams handling of different priorities and types.

Considering, that for service network systems based on IaaS architecture the optimal ratio between concurrent and consequent service components is approaching to the “golden section ratio” *(60/40%)*, it could be concluded that servicing of a network workload as flow of packets should be performed at network infrastructure within up to *40%* of consequent (passive) network equipment application. It could be, for example, switching equipment (in some CEN realization), that serving the aggregated workload without any differentiation. The other part of network equipment should be concurrent by the its character. It should differentiate the workload to serve separate streams in the dedicated components, that operate concurrently or in quasi-parallel mode. The examples of such workload are high-priority data flows. We proved that separation of high-priority data flow processing and worsening of non-real-time parameters of data flows leads to significant improvement of the quality of high-priority data flow processing. To avoid mutual impacts among concurrent (parallelized) components when synthesizing the service system, the respective software and hardware resources should be dedicated and reorganized as virtualized components. This increases system’s operational resilience and is based on the improvement of processes at network-dependent levels of the ISO/OSI model. Keeping real the ratio *(60/40%)* between concurrent and consequent service components, we could see that system’s availability is increased. The redistribution of the memory (buffering) and calculation of system resources are performed. It falls to the synthesized network architecture and the weighted delay of the network traffic transportation is decreased. Thus, the QoS for critical applications with special real-time data flow should be improved. Aggregation of the data flow on the consequent components of the service network system should not pay into global QoS worsening. These fluctuations of QoS parameters are local by their character on the separated network interfaces (for ex. at IaaS). In overall due to effects of synergism and system emergency, an improvement of the QoS in service network systems (considering data flows timing parameters) is observed, concerning specified traffic priorities.

On the basis of solutions (1-3) the consideration was done upon the tolerance region (4) using the proposed model, which describes application of the Amdahl's Law and, respectively, the performance of concurrent handlers, in circumstances of availability changes of their components after the Dodonov-Lande model for a complex systems resilience. It allows to improve the effectiveness of the IaaS, PaaS and SaaS implementations after criteria of the system performance index, system availability and resilience using given set of initial parameters, including service network system structure, and parameters of the traffic, considering modern NFV approach. Additionally, the models and approaches of the aggregated traffic processing using more precise structural, operational and statistical system parameters were considered in this paper.

Acknowledgments

This research is supported by the project entitled "Methods for designing the heterogeneous information and communication system to deploy the software defined multi-purpose 5G networks", funded by Ukrainian government.

References

[1] N. Kryvinska and C. Strauss. Conceptual model of business services availability vs interoperability on collaborative IOT-enabled e-business platforms. In Internet of Things and Intercooperative Computational Technologies for Collective Intelligence, pages 167–187. Springer, 2013.

[2] N. Kryvinska and C. Strauss. Next Generation Networks: Service Delivery and Management. Peter Lang, 2011.

[3] W. F. Witt III. Keep your feet on the ground when moving software into the cloud. International Journal of Digital Content Technology and its Applications, 4(2):10–17, 2010.

[4] I. Demydov, M. Klymash, Z. Kharkhalis, and B. Strykhalyuk. The research of the availability at cloud service systems. In International Conference on Radio Electron- ICS & Info Communications (UkrMiCo), pages 438–442. IEEE, 2016.

[5] I. Demydov, M. Klymash, N. Kryvinska, and C. Strauss. Enterprise distributed service platform–network architecture and topology optimization. International Journal of Space-Based and Situated Computing, 2(1):23–30, 2012.

[6] I. Demydov, O. Lavriv, Yu. Dobush, B. Buhyl, and M.Klymash. Analysis of service workflows distribution and service delivery platform parameters. International Journal of Services, Economics and Management, 5(4):280–290, 2013.

[7] I. A. Ryabinin. The theoretical basics of the marines ships EES design. Naval Academy of USSR, 1964.

[8] B. G. Volik and I. A. Ryabinin. The effectiveness, reliability and robustness of the controlling systems. Automatics and telemechanics, 12:151–160, 1984.

[9] V. F. Krapivin. To the theory of complex systems robustness. Moscow: Nauka, 1978.

[10] V. F. Krapivin. Theoretical and gaming methods of complex systems' synthesis in the conflict situations. Moscow: Sovietskoye radio, 1972.

[11] I. A. Ryabinin and Cherkesov G. N. Logical and probability methods of the reliability investigations in the structureally complexive systems. Moscow: Radio and Communications, 1981.

[12] B. G. Volik. Reliability, effectiveness and resillance of the controlling systems. Control systems and their application, Moscow: Institute of control sciences of USSR Academy of Science:23–31, 1985.

[13] V. A. Guliaev, A. G. Dodonov, and S. P. Peehov. The ressilent organization structures calculation. Kyiv: Naukova dumka, 1982.

[14] A. G. Dodonov, M. G. Kuznetsova, and E. S. Gorbachyk. An introduction to the system ressilence calculation theory. Kyiv: Naukova dumka, 1990.

[15] A. G. Dodonov and D. V. Lande. The ressilence of infor- mational systems. Kyiv: Naukova dumka, 2011.

[16] B. Strykhalyuk, I. Demydov, V. Romanchuk, and M.Beshley. A statistical parameters and characteristics study of informational flows in heterogeneous networks. Scientific Proc. of Ukrainian Research Institute of Commun, (6):82–92, 2014.

[17] M. Klymash, O. Lavriv, B. Bugil, and R. Bak. The simulation and provision of the buffer's utilization in the multi service traffic distribution system using auto stopping technique. Problems of Telecommunications, 2(4):108–117, 2011.

[18] P. Deshchyskyi, B. Strykhalyuk, and M. Kaidan. Investigation of the network overloading impact to the resilience of the peering system. Herald of Lviv Polytechnic National University, (738): 231–235, 2012.

[19] G. Amdahl. Validity of the single processor approach to achieving large scale computing capabilities. In Proceedings of the ACM spring joint computer conference, pages 483–485. ACM, 1967.

[20] I. Demydov, O. Shpur, R. Lutfor, and Yu. Deshchynskyi. The method of improving the availability at cloud service systems. In International Conference Radio Electronics & Info Communications (UkrMiCo), 2016, pages 1–3. IEEE, 2016.

[21] P. Deshchyskyi, B. Strykhalyuk, and M. Kaidan. Enhancement of the servicing quality in the converged mobile systems on the basis of UMA-A platform. Problems of Telecommunications, 13(1): 3–19, 2014.

[22] M. Klymash, B. Strykhalyuk, O. Shpur, and M. Beshley. The model of services providing on the basis of cloud-system logical structure adaptation method. Scientific Proceedings of Ukrainian Communication Research Institute, (5): 3, 2014.

[23] M. Beshley, M. Klymash, B. Strykhalyuk, O. Shpur, B.Bugil, and I. Kagalo. Soa quality management subsystem on the basis of load balancing method using fuzzy sets. International Journal of Computer Science and Software Engineering (IJCSSE), 4(1):10–21, 2015.

[24] M. Klymash and B. Buhyl. The common optimization technique for telecommunication network structures by the criterion of its resources distribution effectiveness. Information processing systems, (7): 72–78, 2013.

[25] P. Barham, B Dragovic, K. Fraser, S. Hand, T Harris, A.Ho, R Neugebauer, I. Pratt, and A. Warfield. Xen and the art of virtualization. In ACM SIGOPS operating systems review, volume 37, pages 164–177. ACM, 2003.

[26] I. Demydov, M. El Hatri, and O. Shpur. Dynamic correction of routing metrics by pervasive structural routing in the scalable distributed service networks. In Second International Scientific-Practical Conference Problems of Infocommunications Science and Technology (PIC S&T 2015), pages 164–166. IEEE, 2015.

[27] I. Demydov, B. Strykhalyuk, O. Shpur, M. El Hatri, and Yu. Klymash. The structural-functional synthesis of cloud service delivery platform after service availability and performance criteria. Information processing systems, (1):144–159, 2015.

[28] I. Demydov, B. Strykhalyuk, and M. Klymash. The holographic network analysis. In IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom'2014), pages 66–68. IEEE, 2014.

[29] M. Klymash, I. Demydov, M. El Hatri, and Yu. Deshchynskiy. The investigation of the availability at cloud service systems. Herald of Lviv Polytechnic National University, (849): 218–228, 2016.

[30] B. Strykhalyuk, Z. Kharkhalis, and I. Demydov. The structural and functional optimization of virtual machines migration processes at distributed datacenters. Computer Technologies of Printing: Scientific Proceedings, (32): 69–81, 2014.