



Design of a Multidomain IMS/NGN Service Stratum

Sylwester Kaczmarek  and Maciej Sac  ^(✉)

Faculty of Electronics, Telecommunications and Informatics,
Gdańsk University of Technology, 11/12 Narutowicza Street,
80-233 Gdańsk, Poland
{kasy1, msac}@eti.pg.edu.pl

Abstract. The paper continues our research concerning the Next Generation Network (NGN), which is standardized for delivering multimedia services with strict quality and includes elements of the IP Multimedia Subsystem (IMS). A design algorithm for a multidomain IMS/NGN service stratum is proposed, which calculates the necessary CSCF servers CPU message processing times and link bandwidths with respect to the given maximum values of mean call set-up delays $E(CSD)$ and mean call disengagement delays $E(CDD)$ for all types of successful call scenarios. These delays are standardized call processing performance parameters, which are very important for satisfaction of users and overall success of the IMS/NGN concept. In the paper a block diagram of the algorithm and details regarding its operation are described. Using the implemented algorithm several multidomain IMS/NGN service stratum designs are performed for different data sets and their results are discussed.

Keywords: IMS · Network design · NGN · Service stratum
Teletraffic engineering

1 Introduction

The Next Generation Network (NGN) [1] is a standardized network architecture including elements of the IP Multimedia Subsystem (IMS) [2] (thus the names “IMS-based NGN” and “IMS/NGN” are frequently used) and dedicated to satisfy current and future needs of information society for delivering high quality multimedia services. For a commercial success IMS/NGN should be properly designed [3], to guarantee values of quality parameters, which are satisfactory for users. It particularly concerns standardized call processing performance parameters [4, 5], which include mean call set-up delay $E(CSD)$ and mean call disengagement delay $E(CDD)$.

The paper proposes a design algorithm for resources of a multidomain IMS/NGN service stratum. The algorithm takes into account maximum values of $E(CSD)$ and $E(CDD)$ defined separately for all types of successful call scenarios (intra- and inter-operator) generated in particular domains. Calculations utilize the analytical traffic model of a multidomain IMS/NGN, which was a subject of our previous research [6–8]. Basic information about this model is provided in Sect. 2. Section 3 is dedicated to the proposed algorithm and contains the description of its block diagram and

performed operations. In Sect. 4 results of several multidomain IMS/NGN service stratum designs are presented and discussed. The paper is summarized in Sect. 5.

2 Model of a Multidomain IMS/NGN

In the paper a multidomain IMS/NGN is considered with two domains administered by two operators [6–8]. Each domain consists of the following elements [2, 9]: User Equipments (UEs), Call Session Control Function servers (P-CSCF – Proxy-CSCF, S-CSCF – Serving-CSCF, I-CSCF – Interrogating-CSCF), SUP-FE/SAA-FE (Service User Profile Functional Entity/Service Authentication and Authorization Functional Entity), RACF (Resource and Admission Control Functions). To distinguish between the elements of different domains, numbers 1 and 2 are added to the above mentioned names (e.g. P-CSCF 1, I-CSCF 2). As both operators have their own core and access networks, letters “A” and “C” are additionally used for RACF units (e.g. RACF A1 controls the resources of access network of operator 1).

In the assumed network 16 different service scenarios are performed (registration as well as intra- and inter-operator calls, which can be successful or unsuccessful due to lack of transport resources) [6–8]. In terms of this paper the most important are successful scenarios, for which $E(CSD)$ and $E(CDD)$ are calculated by our analytical traffic model of a multidomain IMS/NGN [6–8]: b1 (intra-operator call, domain 1, the same access area, success), b2 (intra-operator call, domain 2, the same access area, success), d1 (intra-operator call, domain 1, different access areas, success), d2 (intra-operator call, domain 2, different access areas, success), f1 (inter-operator call, originated in domain 1, success), f2 (inter-operator call, originated in domain 2, success). To obtain $E(CSD)$ and $E(CDD)$, mean values of theoretical delays introduced by network elements are calculated and properly summed. These component delays include:

- mean message waiting times in communication queues (buffering messages when links are busy); for calculation of these times mathematical models of queuing systems are used;
- message transmission times (message lengths divided by links bandwidths);
- propagation times (equal to 5 $\mu\text{s}/\text{km}$ – optical links are assumed);
- mean message waiting times in CSCF servers CPU queues, which store incoming messages when CSCF servers CPUs are busy (for calculation of these times mathematical models of queuing systems are used);
- mean message processing times by CSCF servers CPUs as well as RACF and SUP-FE/SAA-FE elements; they are given by the input parameters [6–8].

The described analytical traffic model was implemented in the MATLAB environment and thoroughly tested with different queuing system models for communication queues and CSCF servers CPU queues (M/M/1, M/G/1, G/G/1 approximations, PH/PH/1 based on moments and whole experimental histograms) [6–8, 10, 11]. It was also successfully verified with a simulation model [8] implementing the operation of real network elements and standardized call scenarios (the simulated elements process SIP and Diameter messages according to standards), and thus reflecting the phenomena taking place in real IMS/NGN network.

As a result of the above mentioned research, we decided to move from the problem of network analysis to network design. In the next section, based on the described analytical traffic model of a multidomain IMS/NGN, a design algorithm is proposed, which allows calculations of service stratum resources: CSCF servers CPU processing power (message processing times) and link bandwidths.

3 Design Algorithm

The algorithm described in this section calculates the resources of service stratum of a multidomain IMS/NGN with respect to the defined maximum values of $E(CSD)$ and $E(CDD)$ for all types of successful call scenarios (Sect. 2). The proposed method does not assume particular contribution of network domains in $E(CSD)$ and $E(CDD)$ for multidomain calls. It optimizes parameters of all domains to possibly the best fulfill the designer requirements. As the structure of a multidomain IMS/NGN is complicated and the numbers of service scenarios as well as network parameters are large, an iterative approach is used.

The block diagram of the proposed design algorithm is described in Fig. 1. The general idea is to start from the maximum available IMS/NGN service stratum resources (low CSCF servers CPU processing times, high link bandwidths) and in each iteration modify parameters of the least loaded network element (CSCF server CPU or link) to increase its load. If such a modification leads to unacceptable $E(CSD)$ or $E(CDD)$ values, it is cancelled and the selected network element is excluded from further calculations. Computations stop when there are no elements, which parameters can be changed. More details about the performed operations are provided later.

The proposed algorithm was implemented in the MATLAB environment as the `design_IMS_NGN()` function, which returns the following output parameters:

- `TINV`: vector with SIP INVITE message processing times [s] by P-CSCF 1, S-CSCF 1, I-CSCF 1, P-CSCF 2, S-CSCF 2, I-CSCF 2; times of processing other messages by the CSCF servers CPUs are proportional with the a_k factors [6–8];
- `b`: vector with link bandwidths [bit/s]; due to space requirements, its full contents (38 elements) will not be presented;
- `ECSD`: vector with guaranteed mean call set-up delays [s] – for the `b1`, `b2`, `d1`, `d2`, `f1`, `f2` call scenarios (Sect. 2);
- `ECDD`: vector with guaranteed mean call disengagement delays [s] – for the `b1`, `b2`, `d1`, `d2`, `f1`, `f2` call scenarios (Sect. 2);
- `load_CPU`: vector with loads offered to CSCF servers CPUs [Erl] (order the same as in `TINV` vector);
- `load_link`: vector with loads offered to links [Erl] (order the same as in `b` vector).

It can be noticed that two first output parameters (`TINV`, `b`) describe required resources of IMS/NGN service stratum. During iterations of the algorithm these parameters are modified to assure proper values of $E(CSD)$ and $E(CDD)$. Additionally, the information about offered loads in the network is returned (`load_CPU`, `load_link` vectors). For the operation of the algorithm the following input parameters are used:

- ECSD_max, ECDD_max: vectors with maximum values of $E(CSD)$ [s] and $E(CDD)$ [s] (order the same as in ECSD, ECDD vectors);
- TINV_min, TINV_max: minimum and maximum values [s] for all elements of TINV vector;
- b_min, b_max: minimum and maximum values [bit/s] for all elements of b vector;
- TINV_step, b_step: the changes in elements of TINV vector [s] and b vector [bit/s] possible in a single algorithm iteration;
- num_iter_max: maximum number of iterations;
- lambdaR: vector with registration request (SIP REGISTER) intensities [1/s] in domain 1 and 2;
- lambda1d: vector with intra-operator call set-up request (SIP INVITE) intensities [1/s] (domain 1 and 2);
- lambda2d: vector with inter-operator call set-up request (SIP INVITE) intensities [1/s] (requests originated in domain 1 and 2);
- rC: vector with ratios of calls involving multiple access areas to all intra-operator calls (domain 1 and 2);
- pb: vector with probabilities of transport resource unavailability in access 1, core 1, access 2 and core 2;
- EX: vector with mean message processing times [s] by RACF A1, RACF C1, RACF A2, RACF C2;
- EY: vector with mean message processing times [s] by SUP-FE 1/SAA-FE 1, SUP-FE 2/SAA-FE 2;
- d: vector with link lengths [m] (order the same as in b vector);
- queueing: the type of queuing system models used in calculations (Sect. 2); M/M/1 and M/G/1 models are available;
- S: vector with message lengths [bit]; due to space requirements, its full contents (31 elements) will not be presented.

The mentioned input parameters of the algorithm can be divided into four categories: requirements on $E(CSD)$ and $E(CDD)$ (ECSD_max, ECDD_max), constraints on the designed resources (TINV_min, TINV_max, b_min, b_max), parameters of algorithm iterations (TINV_step, b_step, num_iter_max), input parameters of the analytical traffic model of a multidomain IMS/NGN (lambdaR, lambda1d, ..., S; [6–8]).

As already mentioned, at the beginning of the algorithm maximum available IMS/NGN service stratum resources are assumed (Fig. 1): all elements of the TINV vector are set to TINV_min and all elements of the b vector are set to b_max. For such a network configuration the values of the ECSD, ECDD, load_CPU, load_link vectors are calculated using the analytical model of IMS/NGN (Sect. 2; the same model is later used to update the above mentioned values). If any element of the obtained ECSD, ECDD vectors exceeds the given maximum values (ECSD_max, ECDD_max vectors), the design process stops with a proper warning message.

Otherwise, the algorithm starts the first iteration ($i = 1$) of its main “for” loop. The number of iterations can be limited using the iter_max input parameter. When $i > \text{iter_max}$ the values of the ECSD, ECDD, load_CPU, load_link vectors are updated and the algorithm returns its output parameters. The same situation occurs when

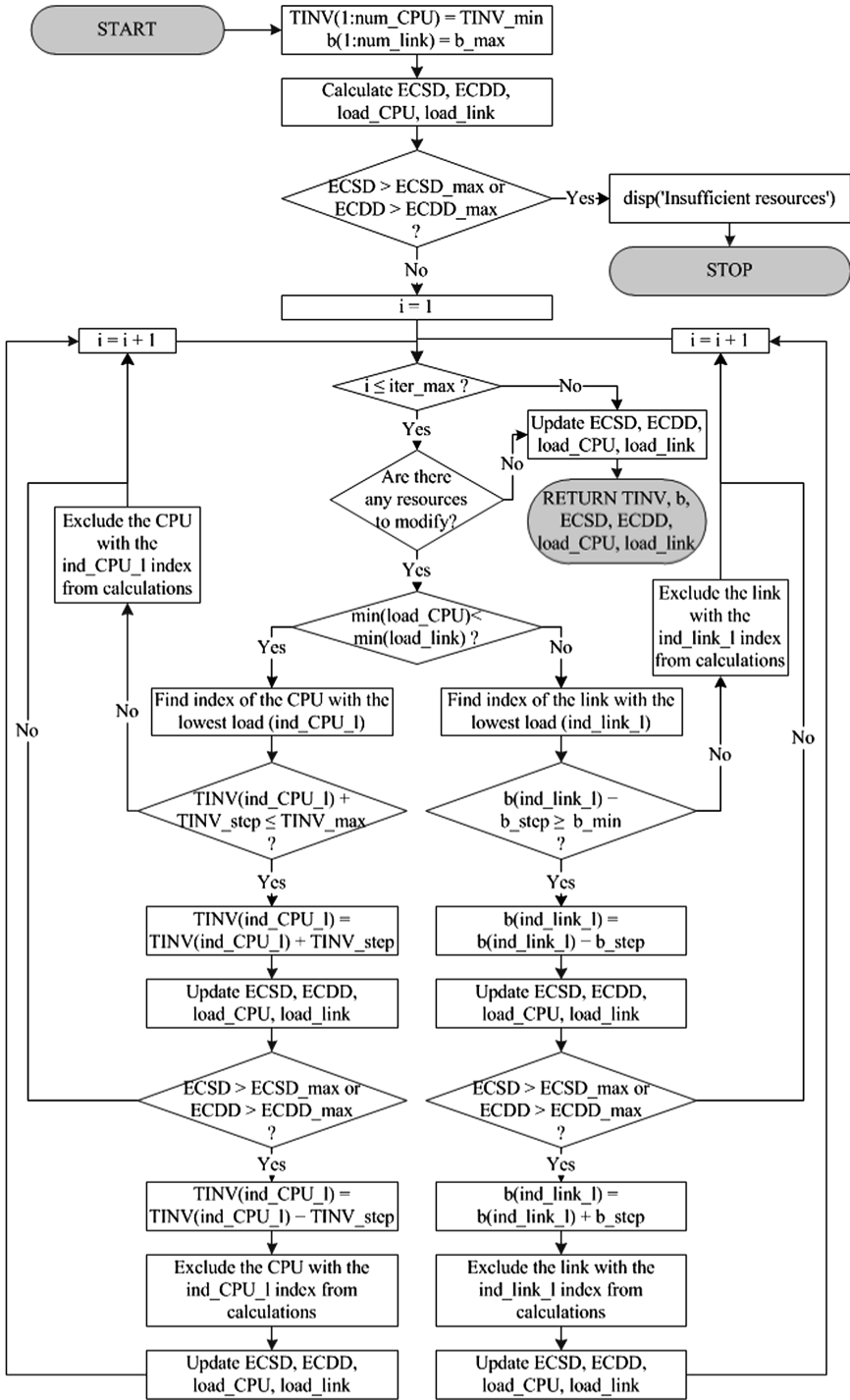


Fig. 1. Block diagram of the proposed design algorithm for IMS/NGN.

parameters of all CSCF servers CPUs and links are fixed and there are no resources to modify (all CPUs and links are excluded from calculations – more details later).

The next operation in each iteration is to determine the type of the least loaded network element (CSCF server CPU or link). It is performed by comparing minima of the $load_CPU$ and $load_link$ vectors. Subsequently, the index of the least loaded element is found (ind_CPU_1 for CSCF servers CPUs or ind_link_1 for links). To increase the load offered to this element, its resources are modified (SIP INVITE message processing time of the CSCF server CPU is increased by $TINV_step$, bandwidth of the link is decreased by b_step). Before performing this operation the constraints on $TINV$ and b are checked ($TINV_max$, b_min input parameters). When they are violated, no resource modifications are performed, the selected network element is excluded from further calculations and the next algorithm iteration starts.

After modifying resources of the least loaded network element, the values of the $ECSD$, $ECDD$, $load_CPU$, $load_link$ vectors are updated. If no element of the updated $ECSD$, $ECDD$ vectors exceeds the given maximum values ($ECSD_max$, $ECDD_max$ vectors), the algorithm continues its operation in new iteration. Otherwise, the performed resource modification is cancelled (SIP INVITE message processing time of the CSCF server CPU is decreased by $TINV_step$, bandwidth of the link is increased by b_step), the selected network element is excluded from further calculations, the values of the $ECSD$, $ECDD$, $load_CPU$, $load_link$ vectors are updated and then the next algorithm iteration starts.

4 Results

We present several multidomain IMS/NGN service stratum designs using the proposed algorithm. In experiments the following input parameters of the algorithm were used: $TINV_min = 0.05$ ms, $TINV_max = 10$ ms, $b_min = 1$ bit/s, $b_max = 50$ Mbit/s, $num_iter_max = 100000$, $lambdaR = [50, 50]$ 1/s, $lambda1d = [50, 50]$ 1/s, $lambda2d = [50, 50]$ 1/s, $rC = [0.5, 0.5]$, $pb = [0, 0, 0, 0]$, $EX = [10, 10, 10, 10]$ ms, $EY = [10, 10]$ ms, $d = [200, 200, \dots, 200]$ km, M/M/1 queuing system models.

The results of the performed investigations are presented in Fig. 2 (data set 1) and Fig. 3 (data set 2). Each of them includes six subfigures with mean call set-up delays for particular call scenarios ($b1$, $b2$, $d1$, $d2$, $f1$, $f2$ – Sect. 2; e.g. $E(CSD)_{b1}$ concerns the $b1$ scenario). Due to space limitations analogical subfigures for mean call disengagement delays are not demonstrated in the paper. All subfigures of Figs. 2, and 3 include three charts: maximum values of $E(CSD)$ given at the input of the design algorithm (elements of the $ECSD_max$ vector) – blue circles, values obtained for the network designed using the algorithm with $TINV_step = 0.5$ ms, $b_step = 0.5$ Mbit/s (red “x” signs) as well as $TINV_step = 0.1$ ms, $b_step = 0.1$ Mbit/s (green squares).

In experiments with data set 1 the elements of the $ECSD_max$ and $ECDD_max$ vectors (blue circles in Fig. 2) increase linearly ($ECSD_max = k \cdot [45, 45, 45, 45, 90, 90]$ ms; $ECSD_max = k \cdot [25, 25, 25, 25, 40, 40]$ ms; $k = 1, 2, \dots, 10$). It can be noticed that for different $TINV_step$, b_step values (0.5 ms, 0.5 Mbit/s – red “x” signs; 0.1 ms, 0.1 Mbit/s – green squares) the designed networks fulfill the condition of $ECSD \leq ECSD_max$. For small k values (1–3) the obtained $E(CSD)$ are very close to

the maxima given at the input of the design algorithm – for all call scenarios. However, when $k > 3$ a similar situation occurs only for the f1 and f2 scenarios and for the other scenarios there are visible differences between the assumed maximum and the obtained call set-up delays. These differences are smaller when smaller TINV_step, b_step values are used, which makes network design process more time consuming.

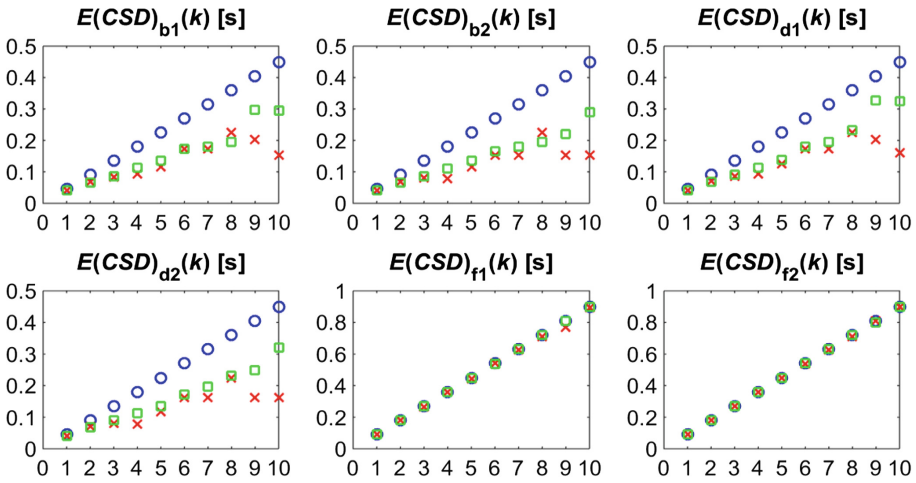


Fig. 2. Mean call set-up delays for the b1, b2, d1, d2, f1, f2 call scenarios (data set 1).

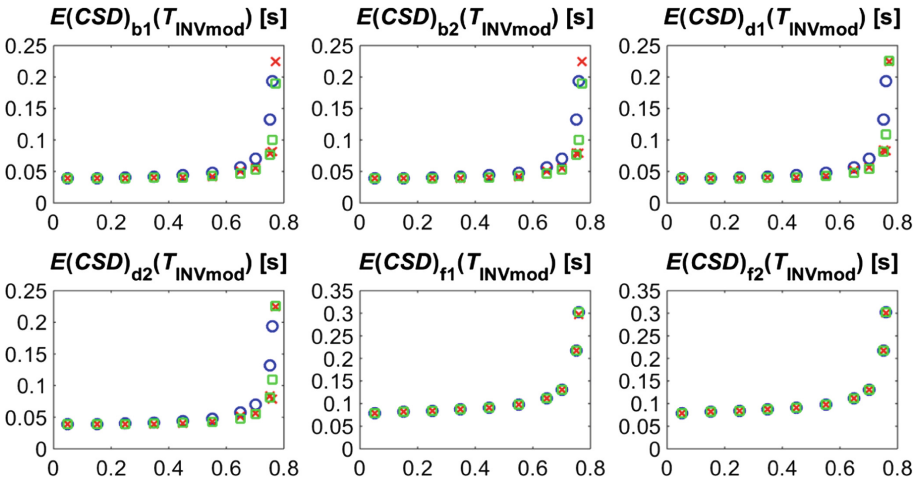


Fig. 3. Mean call set-up delays for the b1, b2, d1, d2, f1, f2 call scenarios (data set 2). The values of T_{INVmod} are given in milliseconds.

The reason for the above mentioned differences is the complexity of a multidomain IMS/NGN. This results in the fact that there are certain relations between call set-up and disengagement delays for particular call scenarios, which additionally change with

network load. The relations assumed in data set 1 (Fig. 2) are fixed (all maximum $E(CSD)$ values grow linearly) and only for $k < 3$ the algorithm can find network parameters, which give satisfactory mean call set-up delays for all call scenarios.

To further investigate this matter, the experiments with data set 2 were performed (Fig. 3), where the ECSD_max and ECDD_max vectors were taken from the analytical model of a multidomain IMS/NGN (Sect. 2) with $b_{mod} = 50$ Mbit/s and $TINV_{mod} = 0.05 \dots 0.8$ ms. b_{mod} represents link bandwidth, which was constant and the same for all links. $TINV_{mod}$ is SIP INVITE message processing time by CSCF servers CPUs – it was changed during consecutive analytical model runs but always the same values were applied for all CPUs.

Using the approach presented in Fig. 3 proper relations between call set-up and disengagement delays for particular call scenarios are preserved. This way IMS/NGN service stratum designs using the proposed algorithm lead to $E(CSD)$ and $E(CDD)$ results very close to the assumed maxima. Despite such a good conformity, the designed networks have different resources than the parameters used to calculate the ECSD_max and ECDD_max vectors with the analytical model (Table 1). For example for $TINV_{mod} = 0.25$ ms, $TINV_{step} = 0.1$ ms, $b_{step} 0.1$ Mbit/s the algorithm returns $TINV = [0.05, 0.15, 0.25, 0.05, 0.15, 0.25]$ ms and $b = [26.3, 50, 12.3, 26.3, 50, 32.8, \dots]$ Mbit/s, while during calculation of ECSD_max and ECDD_max the values of $TINV_{mod} = 0.25$ ms were used for all CSCF servers CPUs and the values of $b_{mod} = 50$ Mbit/s were used for all links. This demonstrates that there are different sets of IMS/NGN parameters, which give very similar $E(CSD)$ and $E(CDD)$ results. It is also very helpful for the designer to know the operation of the network and the relations between call set-up and disengagement delays for particular call scenarios.

Table 1. Selected IMS/NGN service stratum design results for data sets 1–2 (Figs. 2 and 3).

data set	TINV_step [ms]	b_step [Mbit/s]	TINV [ms]	b [Mbit/s], first 6 elements
1, $k = 2$	0.5	0.5	[0.55, 0.55, 1.55, 0.55, 0.55, 1.55]	[4, 11.5, 2, 4, 8.5, 5, ...]
1, $k = 2$	0.1	0.1	[0.45, 0.55, 1.25, 0.45, 0.55, 1.25]	[3.7, 11.9, 1.7, 3.7, 8.8, 4.6, ...]
1, $k = 3$	0.5	0.5	[0.55, 0.55, 1.55, 0.55, 0.55, 1.55]	[2.5, 10, 1.5, 2.5, 7.5, 3, ...]
1, $k = 3$	0.1	0.1	[0.55, 0.75, 1.65, 0.55, 0.75, 1.65]	[2.9, 9.3, 1.3, 2.8, 6.9, 3.5, ...]
2, $TINV_{mod} = 0.15$ ms	0.5	0.5	[0.05, 0.05, 0.05, 0.05, 0.05, 0.05]	[30.5, 50, 14, 30.5, 50, 38, ...]
2, $TINV_{mod} = 0.15$ ms	0.1	0.1	[0.05, 0.05, 0.15, 0.05, 0.05, 0.15]	[33.5, 50, 15.7, 33.6, 50, 42, ...]
2, $TINV_{mod} = 0.25$ ms	0.5	0.5	[0.05, 0.05, 0.55, 0.05, 0.05, 0.55]	[31, 50, 14.5, 31, 50, 38.5, ...]
2, $TINV_{mod} = 0.25$ ms	0.1	0.1	[0.05, 0.15, 0.25, 0.05, 0.15, 0.25]	[26.3, 50, 12.3, 26.3, 50, 32.8, ...]

5 Conclusions

The paper continues our research concerning call processing performance parameters (mean call set-up delay $E(CSD)$ and mean call disengagement delay $E(CDD)$) in a multidomain IMS/NGN, which are very important for satisfaction of users and overall success of this concept. We move from the problem of network analysis, investigated earlier, to the aspect of network design. Based on our analytical traffic model, a design algorithm for a multidomain IMS/NGN service stratum is proposed, which returns CSCF servers CPU message processing times and link bandwidths for given maximum values of $E(CSD)$ and $E(CDD)$ for all types of successful call scenarios.

A block diagram and details regarding the operation of the algorithm are described. It is also implemented and tested for several data sets. The performed research demonstrate that the algorithm works properly – the designed networks are close to the assumed maxima of $E(CSD)$ and $E(CDD)$, but do not exceed them. However, for some sets of input parameters it may be not possible to get very near to these maxima for all call scenarios, as for particular scenarios there are certain relations between call set-up and disengagement delays, which additionally change with network load. Therefore, it is advisable for a designer to know some details about the network operation and learn the above mentioned relations.

References

1. General overview of NGN, ITU-T Recommendation Y. 2001, Dec 2004
2. IP Multimedia Subsystem (IMS); Stage 2 (Release 14), 3GPP TS 23.228 v14.3.0, Mar 2017
3. Hadžialić, M., et al.: Problem of IMS modeling – solving approaches. In: CTRQ 2012 : The Fifth International Conference on Communication Theory, Reliability, and Quality of Service. Chamonix/Mont Blanc, France (2012)
4. Call processing performance for voice service in hybrid IP networks, ITU-T Recommendation Y. 1530, Nov 2007
5. SIP-based call processing performance, ITU-T Recommendation Y. 1531, Nov 2007
6. Kaczmarek, S., Sac, M.: Traffic model of a multidomain IMS/NGN. Telecommun. Rev. Telecommun. News (8–9), 1030–1038 (2014)
7. Kaczmarek, S., Sac, M.: Call processing performance in a multidomain IMS/NGN with asymmetric traffic. In: Grzech, A., et al. (eds.) Information Systems Architecture and Technology: Selected Aspects of Communication and Computational Systems, pp. 11–28. Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław (2014)
8. Kaczmarek, S., Sac, M.: Verification of the analytical traffic model of a multidomain IMS/NGN using the simulation model. In: Grzech, A., Borzemski, L., Świątek, J., Wilimowska, Z. (eds.) Information Systems Architecture and Technology: Proceedings of 36th International Conference on Information Systems Architecture and Technology – ISAT 2015 – Part II. AISC, vol. 430, pp. 109–130. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-28561-0_9
9. Functional requirements and architecture of next generation networks, ITU-T Recommendation Y.2012, Apr 2010

10. Kaczmarek, S., Sac, M.: Analysis of IMS/NGN call processing performance using G/G/1 queuing systems approximations. *Telecommun. Rev. Telecommun. News* (8–9), 702–710 (2012)
11. Kaczmarek, S., Sac, M.: Analysis of IMS/NGN call processing performance using phase-type distributions based on experimental histograms. In: Świątek, J., Borzemski, L., Wilimowska, Z. (eds.) *Information Systems Architecture and Technology: Proceedings of 36th International Conference on Information Systems Architecture and Technology – ISAT 2017*. AISC, vol. 656, pp. 138–155. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-67229-8_13