

A Multi-Objective Optimization of Cloud Based SLA-Violation Prediction and Adaptation

Vivek Gaur

Birla Institute of Technology, Computer Science Department, Jaipur-302017, India
E-mail: vivekgaur@bitmesra.ac.in

P. Dhyani

Banasthali Vidyapith, Computer Science Department, Jaipur-302019, India
E-mail: dhyani_p@yahoo.com

O.P. Rishi

Kota University, Computer Science Department, Kota-324010, India
E-mail: omprakashrishi@yahoo.com

Abstract—Monitoring of Cloud services is vital for both service providing organizations and consumers. The service providers need to maintain the quality of service to comply their services with the QoS parameters defined in SLA's such as response time, throughput, delay through continuous monitoring of services. The dynamic monitoring involves prediction of SLA violations and subsequent adaptation of the service compositions. The task of adaptation is in fact the task of discovering another plausible composition in the face of services recorded to have generated QoS violations. QoS- Driven Utility based service composition approach considers the individual user's priorities for QoS parameters and determines the overall utility measure of the service composition for the end user. In this work we present the problem of service composition adaptation as a multi-objective assignment optimization problem, which in turn is a NP-hard problem. The evolutionary algorithm GA with Tabu has been formulated as a Memetic and Pareto optimal approach for the adaptation problem and analyzed for efficiency in solving the problem.

Index Terms—Quality of service, service level agreement, service composition, service utility.

I. INTRODUCTION

Cloud Computing provides the business-critical applications to serve its customers and employees in the organization. The user and business units need performance transparency and visible resource usage for valuable utilization of the cloud based services [1]. The monitoring of the cloud services and maintaining optimal performance has become an important task for the organizations. An effective performance monitoring of the cloud services requires an autonomous tool for tracking and validating the quality parameters. To meet the quality of service (QoS) parameters as per SLA is an important task also due to the unique QoS requirements of an individual user [2]. The service composition

selected for the user may not necessarily continue to maintain the negotiated QoS values. In a dynamic cloud environment the QoS values of services tend to vary with time and new services may emerge replacing the original services. The changes may result into violation of the service-level agreement (SLA) as eventually, one or more service components might violate the QoS-constraints. In such cases a suitable substitution of appropriate service component is required. Thus a fully functional model is needed for dynamic monitoring against SLA violations, thereby providing optimal adaptation and exploits the real essence of cloud computing [3,4]. The QoS aware service combination selection aims to determine the best combination of cloud services candidates that satisfies a set of user constraints as per given SLA [5,6]. The selected service composition is the one which maximizes the user related utility function. The use of Utility function as a normalized function enables to determine user preferred QoS values. The aggregated values of the utility functions gives the overall service composition utility resulting as the selection of the best composition. A multi-objective ranking problem can be addressed, with the user priorities determined as a weighted sum of the related utility functions [5]. The problem of web service adaptations performed on detection of SLA violation as part of continuous web service monitoring is seen as an optimal assignment problem. The problem of service adaptation can be observed as a task of matching the requirements of multiple service consumers and the offering from service providers efficiently, while optimizing multiple objectives (QoS parameters). Multiple clients requesting similar services should be satisfied, and secondly, the assignment process of the service consumers and the service providers should be optimized. Since we have several service consumers and equally numbered service providers, the aim is to match the consumer-provider pairs as closely as possible. In this paper we aimed to develop a tool which enables the individual user to find the optimal service composition available with highest utility value with minimum

violation and adaptation cost. In this work we applied Memetic and Pareto-optimal approaches to determine the violation and adaptation cost and analyzed the results for their efficacy. The use of Tabu search for local optimization will improve the results as it helped in maintaining the dynamic sample space for selection of service composition and enable to overcome the limitation of basic GA by preserving the better results in intermediate iteration of GA execution [6,7].

The rest of the paper is organized as follows: The Section II introduces the related work of the study. The proposed framework and algorithm is presented in section III. Section IV presents simulation results and analysis and Section V concludes the paper.

II. RELATED WORK

There are several approaches suggested to address the problem of multi-objective optimization of Cloud based prediction of violation and adaptation. Zeng et al. [8] presented a method similar to Cardoso [9] approach to compute the overall QoS in composition except for loop used in the method. The work performs a Genetic operation on the problem encoded into Genome. They use the Fitness function to maximize and minimize the respective QoS attributes and also impose a penalty as a adaptation cost for the violation of the SLA constraints. Gerardo et al. [10] formulated the optimization of multiple QoS criteria as aggregated single optimization problem using weighted summation. The work presented the use of penalty based method to deal with the QoS constraints. Canfora et al. [11] formulated the multiple QoS criteria based composition of services to be selected from multiple candidate services as combinatorial optimization problem that is known to be a NP-Hard. In this work authors formulated a fitness function using Genetic algorithm to compare the candidate service compositions. Jiuyun et al. [12] presented an Artificial Immune System algorithm based approach to address the service composition problem. In Clonal operation, the compositions (antibodies) with high fitness values are used as heuristic information for generation of service compositions. The work considered various QoS attributes in fitness function such as service response time, service cost, service reliability and service availability and compositions (antibodies) with best fitness are considered as candidate solution for the composition in the algorithm. Ludwig et al. [13] applied memetic algorithm as an improved approach by combined use of swarm intelligence algorithm or any other evolutionary method with a local search technique, to achieve better results. The work used the algorithm called Munkres as local search technique combined with GA as well as a PSO approach as a memetic algorithm. A multi-objective genetic algorithm optimization approach is followed to generate a set of optimal Pareto service composition by optimizing the different objective functions simultaneously [14]. However, the execution

efficiency of the algorithm still needs to be improved further. Yoo et al. [15] presented a multi objective optimization algorithm to order the Pareto-optimal results has been proposed. The presented algorithm ranks the results on cost of service failure with respect to the user's priorities for the services. However, the proposed approach is not capable to compare and present the better Pareto-optimal solution.

The existing approaches lack the consideration of utility of the composition for individual user based on the violation and adaptation cost, an essential requirement for a user-centric approach.

III. PROPOSED FRAMEWORK AND ALGORITHM

In our work, we propose a GA-Tabu based memetic approach for optimal cloud service composition selection as shown in fig.1. Selection of service composition is performed based on the composition utility value considering the SLA violation and adaptation cost for the requested QoS parameters.

A. Framework of SLA Violation predictor based monitoring service

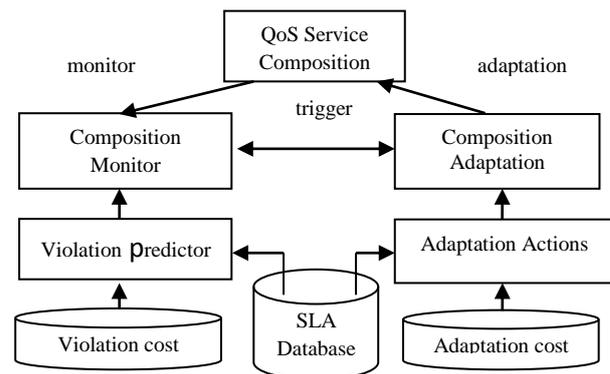


Fig.1. Proposed Framework

- *QoS Service Composition* - composition selected as a candidate solution by the algorithm.
- *Composition Monitor* - monitors the selected service composition for the violations occurred.
- *Violation predictor* - predicts the level of violation against the service level objectives (SLO) and also determines the violation cost.
- *SLA Database* - service level agreement containing the details of the quality that customer demands.
- *Composition Adaptations* - determines the required adaptations for the selected QoS Composition.
- *Adaptation Actions* - determines the possible adaptations to meet the SLA with minimum adaptation cost.
- *Adaption cost* -cost incurred due to the incorporation of the adaptations to meet the SLA requirements.

B. Proposed Algorithm

(1) Fitness Function

Fitness function used in proposed algorithm is as follows:

$$U = UL - TC \quad (1)$$

Where, the utility of the composition (UL) is determined using (2) adopted from the work by T. Yu et al. [5] and the minimum and maximum aggregated values of the i^{th} QoS parameter of service composition C are computed as in (3) and (4).

$$F(C) = \sum_{j=1}^r \frac{Q_j^{\max} - q_j(C_i)}{\max(1, (Q_j^{\max} - Q_j^{\min}))} \quad (2)$$

$$Q_j^{\max} = \sum_{i=1}^n Q_{i,j}^{\max}, Q_{i,j}^{\max} = W_{j,i} \max_{s_{ik} \in S_i} q_j(c_{ik}) \quad (3)$$

$$Q_j^{\min} = \sum_{i=1}^n Q_{i,j}^{\min}, Q_{i,j}^{\min} = W_{j,i} \min_{s_{ik} \in S_i} q_j(c_{ik}) \quad (4)$$

Where, $i = 1$ to n and $j = 1$ to r represents different service types (service1, service2, service3 etc.) and the QoS parameters (price, response time, reliability and throughput) respectively. W_{ji} is the weight (0 to 1) assigned by the end user for j^{th} QoS of i^{th} service. $Q_{i,j}^{\max}$ is the maximum value of the j^{th} parameter in all service candidates/ chromosomes of a i^{th} service category, similarly $Q_{i,j}^{\min}$ is the minimum value. $Q_{i,j}^{\max}$ is the maximum value of the j^{th} parameter of a composite composition C and similarly $Q_{i,j}^{\min}$ is the minimum value. $q_j(c_{ik})$ is the j^{th} QoS parameter values for k^{th} candidate of i^{th} service type. The total cost (TC) is determined using (5).

$$TC = \text{Total Cost} = AC + CC + VC \quad (5)$$

AC = Cost of adaptation from service i to service j , as system parameters.

CC = Combination constraints as specified during registration of the services.

VC = Violation Cost using (6).

$$V(C) = \sum_{s=1}^{\text{Services}} \sum_{q=1}^{\text{QoS}} (W_{sq} * (QoSval(C_{sq}) \sim T_{sq}) / T_{sq}) \quad (6)$$

Where, T_{sq} = Threshold over service s for QoS parameter q .

(2) Memetic Approach

Memetic approach based solutions employs Tabu-search based local optimal search strategy while

performing reproduction pool generation during the general steps of basic GA. The intermediate results are stored and compared and finally a composition with higher utility value is selected.

Step1. Generate population of chromosomes P =

{Set of service compositions}

Step2. Compute Utility Value of each chromosome using 1.

Step3. Generate Reproduction Pool:

i. Compute the Violation Matrix.

ii. Perform local optimization search (using Tabu-out the most inferior chromosomes from the reproduction pool and replace with the mutated chromosome, if better) over the current population.

iii. Repeat step 3.ii until pool is full.

Step4. Perform Crossover

Step5. Perform Mutation with given mutation probability.

Step6. Repeat from Step2 until given no. of iterations are over.

(3) Pareto-Ordering Approach

Pareto-ordering approach considers the comparison of individual QoS parameter of each service type of a candidate (chromosome) composition which effectively realizes the weight or priority requested by the customer as per SLA.

Step1. Generate population of chromosomes P =

{Set of service compositions}

Step2. Compute Utility Value of each chromosome using (1).

Step3. Generate Reproduction Pool:

i. Compute the Violation Matrix.

ii. Apply *Pareto Ordering* of the chromosomes in terms of Violation costs

iii. Store Utility Value of the first chromosomes.

Step4. Perform Crossover

Step5. Perform Mutation with given mutation probability.

Step6. Repeat from Step 2 to step 5 until given no. of iterations are over.

Pareto ordering fitness function

- Def 1: Pareto Ordering of Chromosomes (Service Compositions)- Descending
- $X \succ Y$ iff $x_{1..i} = y_{1..i}, x_{i+1} > y_{i+1}, i \in [1..N]$
- Def 2: Violation Cost using (6).

IV. SIMULATION AND RESULTS

This section presents the simulation setup and the analysis of the simulation results. The simulation setup details the input data used for the simulation purpose and compares the performance of the two approaches in results analysis section.

A. Simulation Setup

Four QoS parameters were considered for simulation purpose, namely, price, response time, throughput, and reliability. A random dataset was generated for the four QoS parameters. During initial simulation, five different service types were considered namely, storage service, software service, processing service, network service and platform service; each with variable number of candidate service providers like $S_1, S_2, S_3, \dots, S_n$, as shown in tables 1, 2, 3, 4, and 5 respectively.

The adaptation and combination constraints costs were also generated using random function for the simulation purpose. As defined earlier in section III, the adaptation cost (AC) is the overhead incurred due to substitution of one service provider with another; diagonals were set to zeros since cost of substituting with itself is zero or no meaning.

The combination constraint cost (CC) is set to 0 (zero) in case of no constraint else set 1 (one) if a service cannot be replaced with other and it generates a heavy penalty say (999) for the simulation purpose.

The QoS violation thresholds (VT) for each of the four QoS parameters were set uniform for all services, and in the reported experiment, the vector VT was assumed as, $VT = [9000, 4.8, .8, .5]$.

Table 1. Storage Service

Service\ QoS	Price (dollar)	Response Time (seconds)	Throughput (0-1)	Reliability (0-1)
S1	2175.633	3.533576	0.073995	0.758112
S2	2510.418	2.788945	0.684096	0.871111
S3	8929.224	1.567145	0.402388	0.350777
S4	7032.232	0.831018	0.982835	0.685536
S5	5557.379	3.112486	0.402184	0.294149
S6	1844.337	4.939674	0.620672	0.530629
S7	2120.308	0.85216	0.15437	0.832423
S8	773.4681	1.288961	0.381345	0.59749
S9	9138.004	1.983997	0.161134	0.335311

Table 2. Software Service

Service\ QoS	Price (dollars)	Response Time (seconds)	Throughput (0-1)	Reliability (0-1)
S1	3242.199	0.732574	0.99685	0.491806
S2	3017.268	3.155706	0.553542	0.071037
S3	116.8099	4.296602	0.515458	0.887739
S4	5399.051	4.871108	0.330682	0.064634
S5	953.7269	2.854192	0.430002	0.436185
S6	3217.356	3.214326	0.553542	0.056342
S7	217.5147	4.196602	0.324412	0.822331

Table 3. Processing Service

Service\ QoS	Price (dollars)	Response Time (seconds)	Throughput (0-1)	Reliability (0-1)
S1	1776.025	3.240992	0.662808	0.414523
S2	3985.895	0.126141	0.330829	0.46484
S3	1339.313	4.211033	0.898486	0.763957
S4	308.8955	2.795163	0.118155	0.818204
S5	9391.417	4.270511	0.988418	0.100222
S6	3013.061	1.739396	0.539982	0.178117
S7	2955.338	2.230133	0.706917	0.359635
S8	3329.363	0.271197	0.999492	0.056705
S9	4670.682	0.885538	0.287849	0.521886
S10	2301.231	2.453461	0.679873	0.344637

Table 4. Network Service

Service\ QoS	Price (dollars)	Response Time (seconds)	Throughput (0-1)	Reliability (0-1)
S1	5517.925	1.732244	0.782872	0.70434
S2	5835.706	4.432719	0.693788	0.729513
S3	5118.199	2.273474	0.009802	0.224277
S4	825.9273	2.067136	0.843213	0.269055
S5	7195.701	1.08866	0.922332	0.673031
S6	9961.561	0.628273	0.770954	0.477492
S7	3545.343	1.544573	0.04266	0.623716
S8	9712.588	3.630522	0.378186	0.236445

Table 5. Platform Service

Service\ QoS	Price (dollars)	Response Time (seconds)	Throughput (0-1)	Reliability (0-1)
S1	7217.533	2.718316	0.269092	0.362459
S2	8777.991	4.923881	0.749018	0.788113
S3	5824.33	3.57839	0.503888	0.780296
S4	706.8434	4.194848	0.64681	0.668512
S5	9227.446	2.166303	0.307746	0.133504
S6	8003.721	2.353124	0.138725	0.021556
S7	2859.469	2.803567	0.475573	0.559841

B. Result Analysis

Simulations were conducted to compare the results of the Memetic Algorithm with those of the Pareto Optimal Genetic Algorithm. The violation costs incurred on applying Memetic and Pareto approaches are presented in fig. 2 and the former one outperforms the later, as the violation cost generated by Pareto-ordering approach is higher for all different number of iterations. The adaptation cost generated by Pareto-ordering method

reaches at maximum to (2.5) as shown in fig. 3, which is considerable higher than Memetic approach which incurs (1.5) at maximum. Fig. 4 depicts the utility values generated for Memetic and Pareto-ordering approaches. The memetic method yields better utility value of 4.25 as compared to 2.6 by Pareto-ordering method.

Pareto Optimal GA was formulated with the expectation of better representation of client's preference to respective QoS-parameters. However observing considerable differences in the compositions' utility values, violation costs, and adaptation costs corresponding to application of the Memetic Algorithm with those of the Pareto approach, the former proves to be better choice for QoS composition selection.

Therefore, we can conclude from the above results that Memetic algorithm outperforms both the basic GA and the Pareto-ordering approaches. Applying MA approach yields higher composition utility value and smaller violation and adaptation cost as compared to the other approaches.

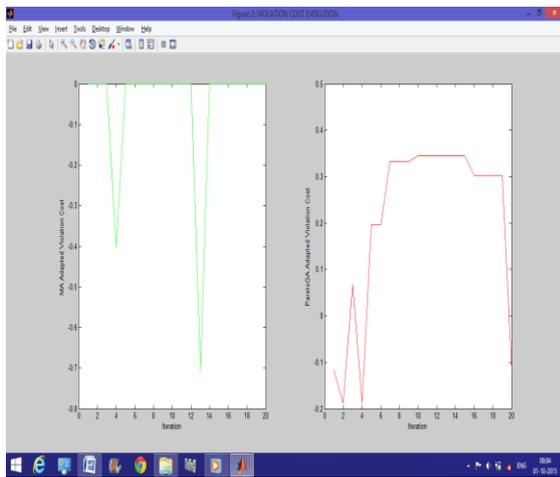


Fig.2. MA-Pareto Violation Cost Plot

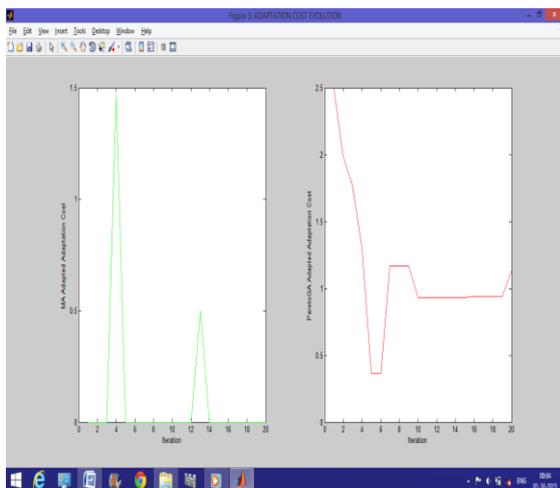


Fig.3. MA-Pareto Adaptation Cost Plot

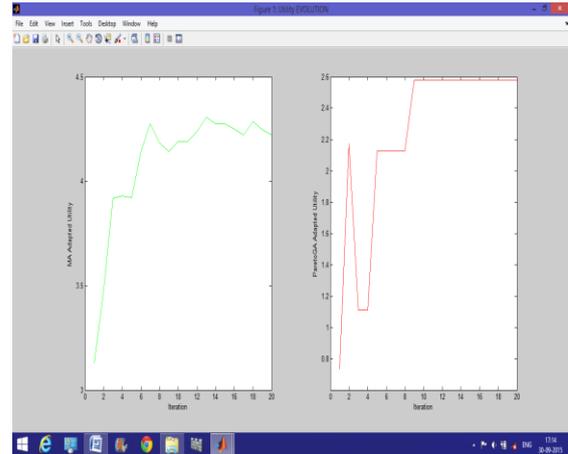


Fig.4. MA - Pareto Adapted Service Utility Value

V. CONCLUSION

The basic Genetic Algorithms are observed to have a limitation of presenting only the best solution found in the final iteration, while intermediary iterations still could produce better outcome. This limitation has been addressed by the development of a Memetic algorithm (MA) and Pareto-ordering solutions that applied Tabu-search based local optimal search strategy while performing reproduction pool generation during the general steps of basic GA. Although Pareto approach considers the comparison of individual QoS parameter of each service type of a candidate (chromosome) composition which effectively realizes the weight or priority requested by the customer as per SLA. Observing considerable differences in the violation and adaptation cost of Memetic with Pareto approach, the former proves to be better choice for QoS composition selection. Therefore, Memetic algorithm approach is concluded as suitable algorithmic solution which can be implemented for the cloud based service composition adaptation in the face of SLA violation.

REFERENCES

- [1] Farnaz Sharifi Milani, Ahmad Habibzad Navin, "Multi-Objective Task Scheduling in the Cloud Computing based on the Patrice Swarm Optimization", *IJITCS*, vol.7, no.5, pp.61-66, 2015. DOI: 10.5815/ijitcs.2015.05.09.
- [2] P. Leitner, A. Michlmayr, F. Rosenberg, and S. Dustdar, "Monitoring, Prediction and Prevention of SLA Violations in Composite Services," in *Proceedings of the IEEE International Conference on Web Services (ICWS'10)*. Los Alamitos, CA, USA: IEEE Computer Society, 2010, pp 369-376.
- [3] Khosro Mongouie, Mostafa Ghobaei Arani, Mahboubeh Shamsi, "A Novel Approach for Optimization Auto-Scaling in Cloud Computing Environment", *IJCNIS*, vol.7, no.11, pp.46-53, 2015. DOI: 10.5815/ijcnis.2015.11.05.

- [4] P. Leitner, B. Wetzstein, D. Karastoyanova, W. Hummer, S. Dustdar, and F. Leymann, "Preventing SLA Violations in Service Compositions Using Aspect-Based Fragment Substitution", in Proceedings of the International Conference on Service -Oriented Computing (ICSOC'10),2010.
- [5] T. Yu, Y. Zhang, and K.J. Lin, "Efficient algorithm for web service selection with end-to-end QoS constraints," ACM Transactions on the web, Vol. 1, pp. 1-26, 2007.
- [6] Vivek Gaur, Praveen Dhyani, O. P. Rishi, "A GA-Tabu Based User Centric Approach for Discovering Optimal QoS Composition," IJMECS, vol.7, no.2, pp.56-62, 2015. DOI: 10.5815/ijmecs.2015.02.08.
- [7] L. Wang, J. Shen and J. Yung, "A Survey on Bio-inspired Algorithms for Web Service Composition," Proc. of the 16th International Conference on Computer Supported Cooperative Work in Design, pp.569-574, 2012.
- [8] Wenying Zeng, Yuelong Zhao and Junwei Zeng, "Cloud Service and Service Algorithm Research," ACM, June 12-14, 2009.
- [9] J. Cardoso, Quality of Service and Semantic Composition of Workflows, PhD thesis, Univ. of Georgia, 2002.
- [10] Gerardo Canfora, Massimiliano Di Penta, Raffaele Esposito, and Maria Luisa Villani. An approach for QoS aware service composition based on Genetic Algorithms. In GECCO '05: Proceedings of the 2005 conference on Genetic and evolutionary computation, pages 1069 - 1075, New York, NY, USA, 2005.
- [11] G. Canfora, M. D. Penta, R. Esposito, and M. L. Villani. An Approach for QoS-aware Service Composition based on Genetic Algorithms. In Genetic and Evolutionary Computation Conference, June 2005.
- [12] J. Xu, S. Reiff-Marganiec, "Towards Heuristic Web Services Composition Using Immune Algorithm," Web Services ICWS '08, IEEE International, pp. 238 – 245, November 2008.
- [13] Simone A. Ludwig, "Memetic Algorithms applied to the Optimization of Workflow Compositions", Journal of Swarm and Evolutionary, October 2012.
- [14] H. Wada, P. Champrasert, J. Suzuki, and K. Oba, "Multi-objective Optimization of SLA - aware Service Composition," in IEEE Workshop on Methodologies for Non- functional Properties in Services Computing, July 2008.
- [15] Yoo, J.J.-W.; Kumara, S.; Dongwon Lee; Seog-Chan Oh;, "A Web Service Composition Framework Using Integer Programming with Non-functional Objectives and Constraints, " E- Commerce Technology and the Fifth IEEE Conference on Enterprise Computing", E-Commerce and E-Services, 2008 10th IEEE Conference on , vol., no., pp.347-350, 21-24 July2008.

Authors' Profiles



Vivek Gaur received a M.E. degree from Birla Institute of Technology and Science (BITS), Pilani, India in December. Currently, he is an Assistant Professor of Computer Science Department at the BIT, Mesra, Ranchi, India. Mr. Gaur's major research interest lies in distributed systems, social networking, and data

mining technologies.



Praveen Dhyani received a Ph.D. degree from Birla Institute of Technology and Science (BITS), Pilani, India. Currently he is a Professor of Computer Science and Executive Director at Banasthali University Jaipur Campus. Previously Dr. Dhyani established and headed national and international centers of BIT MESRA at Jaipur, Bahrain, Muscat, and RAK (UAE). His R&D accomplishments include electronic devices to aid foot drop patients and development of voice operated wheelchair. He is also a member of the Programme Execution Committee (PEC), UIDAI Biometric Centre of Competence (UBCC), Unique Identification Authority of India (UIDAI), Planning Commission, Government of India.



O.P. Rishi received a M.Tech. degree from Birla Institute of Technology (BIT), Mesra, Ranchi, India in May 1998 and a Ph.D. degree from the Banasthali University, Jaipur, India. Currently, he is an Associate Professor of Computer Science Department and Director Research at the Kota University, Kota, India. Dr. Rishi's major research interest lies in service-oriented architecture, distributed systems, web and data mining technologies.

How to cite this paper: Vivek Gaur, P. Dhyani, O.P. Rishi, "A Multi-Objective Optimization of Cloud Based SLA-Violation Prediction and Adaptation", International Journal of Information Technology and Computer Science(IJITCS), Vol.8, No.6, pp.60-65, 2016. DOI: 10.5815/ijitcs.2016.06.08