KSII TRANSACTIONS ON INTERNET AND INFORMATION SYSTEMS VOL. 9, NO. 6, Jun. 2015

A Negotiation Framework for the Cloud Management System using Similarity and Gale Shapely Stable Matching approach

Rajkumar Rajavel[1] and Mala Thangarathinam[2]
[1] Anna Centenary Research Fellow, College of Engineering Guindy, Anna University
Chennai 600025 - India
[e-mail: rajkumarprt@gmail.com]
[2] Associate Professor, College of Engineering Guindy, Anna University
Chennai 600025 - India
[e-mail: malanehru@annauniv.edu]
*Corresponding author: Rajkumar Rajavel

Received January 27, 2014; revised May 6, 2015; accepted June 1, 2015;
published June 30, 2015

Abstract

One of the major issues in emerging cloud management system needs the efficient service level agreement negotiation framework, with an optimal negotiation strategy. Most researchers focus mainly on the atomic service negotiation model, with the assistance of the Agent Controller in the broker part to reduce the total negotiation time, and communication overhead to some extent. This research focuses mainly on composite service negotiation, to further minimize both the total negotiation time and communication overhead through the pre-request optimization of broker strategy. The main objective of this research work is to introduce an Automated Dynamic Service Level Agreement Negotiation Framework (ADSLANF), which consists of an Intelligent Third-party Broker for composite service negotiation between the consumer and the service provider. A broker consists of an Intelligent Third-party Broker Agent, Agent Controller and Additional Agent Controller for managing and controlling its negotiation strategy. The Intelligent third-party broker agent manages the composite service by assigning its atomic services to multiple Agent Controllers. Using the Additional Agent Controllers, the Agent Controllers manage the concurrent negotiation with multiple service providers. In this process, the total negotiation time value is reduced partially. Further, the negotiation strategy is optimized in two stages, viz., Classified Similarity Matching (CSM) approach, and the Truncated Negotiation Group Gale Shapely Stable Matching (TNGGSSM) approach, to minimize the communication overhead.

Keywords: Negotiation Framework, intelligent broker agent, similarity matching and cloud

---

This research work was supported by the Anna Centenary Research Fellowship grant funded by Anna University under the act of Lr.No.CR/ACRF/Jan.2012/17.

http://dx.doi.org/10.3837/tiis.2015.06.005 **ISSN : 1976-7277**

1. Introduction

In the present decade, cloud computing leverages the Information Technology business, with the on-demand service provisioning and metering mechanism, through the support of the Service Level Agreement (SLA). Here, the SLA is an agreement specifying the set of functional and non-functional properties of the service, which is committed between the Service Consumer (SC) and the Service Provider (SP) [1] [2]. The current state-of-the-art cloud management provides a semi-personalized service access mechanism, with a non-negotiable SLA contract [3]. Such a mechanism may not be adoptable in the future, due to the increased demand for an elastic security service required by the SC, which needs to be personalized through the SLA negotiation approach. In order to meet the future demand, an SLA oriented cloud management system (SLAOCMS) was proposed in the previous research work, which supports fully customized service provisioning, by using the negotiation framework in the service layer of the cloud (present on top of the application layer) [4]. This scenario motivates the study of a many-to-many cloud service negotiation framework, for supporting the negotiable SLA concept, which can satisfy the customized service provisioning mechanism. Moreover, the negotiation framework is considered as one of the major issues in future cloud management [5].

In addition, the cloud based resource provisioning is dynamically configured, and bundled via virtual machines through the negotiation of SLA between the service consumer and service provider, using the intermediate third party broker [6]. The cloud agency can act as a broker, for maintaining the dynamic provisioning, monitoring and reconfiguration of the cloud resources on behalf of the user [7]. Since, the cloud participants (consumer, broker, and provider) are independent bodies with different requirements, policies, and objectives, there is a need for a negotiation framework among the parties to resolve their differences [8]. In the real time e-commerce problem, the broker based negotiation framework using other computing paradigms, like the grid and cluster, was restricted to resource constraints due to its negotiation complexity [9] [10]. Therefore, to overcome the resource constraint, a negotiation framework needs to be presented over the cloud environment, due to its elastic scaling of resources. Further, the web service based negotiation framework in the dynamic and complex environment can coordinate the services, using horizontal and vertical coordination, and may lack in coordination and communication among the negotiation parties [11]. So, an agent based mechanism is needed to automate and improve the performance of the negotiation frameworks in the cloud management system. The Agents facilitate the effective coordination, collaboration, and information exchange among the negotiation parties [12].

The objective of the proposed research work is to develop a novel framework for supporting composite service negotiation with the intermediate Intelligent Third-party Broker (ITB), with the notion of optimal negotiation strategy. This framework considers the cloud based e-commerce with multilateral multi-issue negotiation problem [13]. Currently, very few research works are available in the area of many-to-many negotiation frameworks, using a broker model [14, 15, 16]. First, the Negotiation Framework Architecture (NFA) is proposed as a many-to-many model, using the cloud broker which helps to nominate the best SP to the available SCs [17]. Since this architecture uses the single broker for the negotiation process, it increases the complexity of the broker, for servicing the multiple negotiations among the parties. In order to tackle this situation, a Cloud Negotiation Model (CNM) is proposed with multiple broker agents for parallelizing the negotiation workload on the broker's part [18].

Further, the same research work is extended to the more complex and concurrent negotiation model in the multiple interrelated e-markets [19]. In the context of the agent, the individual broker agent proposed in the above model may not be possible to co-ordinate the concurrent negotiation with more number of SPs. Moreover, this co-ordination may lead to a complex negotiation strategy in the broker agent, which may result in increasing the Total Negotiation Time (TNT) and Communication Overhead (CO) among the negotiating parties.

In addition, the broker agent in the existing Composite Service Provisioning (CSP) model may be compatible for atomic service negotiation, but this may not be applicable for multiple composite service negotiation. This situation may lead to a complex problem in composite service provisioning, with more number of composite service negotiation requests [20]. So, the ITB in the proposed ADSLANF consists of Intelligent Third-party Broker Agents (ITBAs), Agent Controllers (ACs) and Additional Agent Controllers (AACs) for composite service negotiation among the Service Provider Agents (SPAs) and Service Consumer Agents (SCAs). Each ITBA is capable of managing one composite service, by assigning all its atomic services present in the composition to multiple ACs, for the negotiation process. The functionality of the AC is to manage one atomic service at a time and to co-ordinate its concurrent negotiation with multiple SPAs by invoking the AAC. Finally, the AAC does the actual negotiation process with the SPA, which is a subset of the AC. Increasing the additional agents (such as AC and AAC) for composite and concurrent negotiation is again a big issue, since this increases the computational complexity of the ITB negotiation strategy in the multi-agent system. Hence, there is a need for a negotiation framework, which should reduce the overhead associated with the ITB and perform the negotiation in less time. Therefore, the proposed framework needs to optimize the negotiation strategy for reducing the TNT and CO among the negotiation parties.

In this research work, the TNT is optimized through the concurrent negotiation of the ITBA using the AAC, and then the CO is optimized using similarity matching, and the Gale Shapely Matching approach. Currently, the literature reports standard models like the geometric, contrast, and ratio models for similarity matching. The geometric model represents the stimuli (similarity objects) as a set of points in the multi dimensional space, where the Euclidean distance among any two points determines the degree of psychological similarity. However, it does not escape from criticism, like the violation of the metric axiom and the property of lines [21]. Due to the severely restricted number of objects, the geometric model is more applicable to identify the similarity among the images. An alternative set theory based model like the contrast model, overcomes the above criticism by representing the stimuli as a set of features, and expresses their similarity as a linear combination of their common and distinctive features measure [22]. Next, the ratio model generalizes the contrast model of similarity, including other set theoretic models. In addition, it is applicable to the similarity feature set, like letters and strings of symbols. Since the composite services are represented as a set of attributes (features), an appropriate Taversky’s ratio model is extended to identify similar atomic services among the compositions received by the ITB. Instead of negotiating every similar service present among the multiple compositions, this approach will cluster all the similar services, and negotiate on behalf of the cluster. This similarity matching approach would considerably reduce the CO due to the elimination of redundancy in service negotiation. To further reduce the CO, an alternative Gale Shapely Stable Matching [23] approach is extended with negotiation grouping and truncation operation. This approach can optimize the negotiation strategy, from the multiple one-to-many communication to the multiple one-to-one, which reduces the CO among the NPs.

More recently, cloud service brokerage has been analyzed as a key concern, for not only managing the performance and delivery of cloud services, but also for negotiating the relationship between the cloud consumer and provider. In the context of the cloud broker architecture and programming model, the brokerage solutions such as aggregation, customization, and integration of services, were suggested for future cloud development and research [24]. The proposed research context of introducing cloud broker based negotiation framework, involves the complex interactions among the cloud participants. This research focus needs to specify the importance of automation and negotiation strategy optimization, which works on top of the existing cloud broker architecture. Here, the automated negotiation is incorporated, using an agent based technology, and the optimization of broker negotiation strategy can be achieved through the elimination of redundant and multipoint characteristic negotiation using the proposed CSM and TNGGSSM approach.

This research work is organized into six sections. The next section presents the related work of the negotiation framework. In section 3, the problem formulation and its measurable performance metrics are represented. Section 4 introduces the proposed architecture of ADSLANF. The negotiation strategy optimization using the CSM approach is presented in section 5. Section 6 deals with the negotiation strategy optimization using the TNGGSSM approach. In section 7, the empirical and experimental evaluation of the proposed ADSLANF compared with the existing frameworks is given. The final section deals with the conclusion and future scope of this research.

2. Related Work

A generic negotiation framework, categorises the taxonomy of the negotiation requirement into higher level parameters, such as the negotiation process, context, issues, offer submission, and negotiation information processing [25]. A process view of the NegoFAST-Bargaining architecture, classified its design details into coordination, policies, negotiation, and generation processes [26]. The negotiation process can be further classified into four stages, such as Information collection, Search, Negotiation, and Evaluation stage [27]. In general, there are several phases involved in the implementation of the SLA negotiation framework lifecycle, such as development, discovery & negotiation, provisioning & deployment, execution, assessment & corrective actions, and termination & decommission of service [28]. The design and implementation of the complete negotiation framework with optimal solution is a well known NP-complete problem [29] [30]. For the sake of simplicity and clear understanding of the negotiation concepts, this research work concentrates only on the service requirement portion of the negotiation phase, and the remaining provisioning and workflow management of composite service orchestration are kept for future work.

The NEGOtiations for achieving and maintaining a Sustainable Enterprise Interoperability (NEGOSEIO) framework, defines the structure of the negotiation process or strategy into five parts; i.e., Initialization, Choosing tactics, Choosing partners, Negotiation, and Contract Adoption [31]. The problem of optimizing the negotiation strategy can be solved in two ways, using long-term and pre-request optimization [32]. Here, the long term optimization can be applicable during the negotiation part using the reinforcement and distributed learning model. A pre-request optimization can be applied during the initialization part of the negotiation strategy using the optimization function, under several quality of service parameter constraints. Such an optimization problem with respect to several constraints, is known to be NP-hard. Therefore, this research work has restricted the focus to pre-request optimization of negotiation strategies, with respect to few parameter constraints, like TNT and CO. So that, the existing negotiation strategies present in the broker based cloud negotiation frameworks are classified in the context of pre-request optimization, using the utility function (with respect to price, negotiation speed and time slot), redundant, and multi-point characteristic function of negotiation. Similarly, the cloud negotiation frameworks are classified in the context of long-term optimization, using the reward function, learning rate, negotiation state, and discount or concession function.

Table 1. Comparison of different frameworks in the service negotiation phase

Negotiation Framework / Model	**Negotiation Approach**	**Performance Measurement Attribute**											**Optimization Context**						
													Pre-request				**Long-term**		
		Utility Value/ Satisfaction Degree	**Cost Value**	**Cloud/Test-bed Load (Utilization)**	**Negotiation Speed**	**No. of Job Failure**	**Average Waiting Time**	**Success/Acceptance Rate**	**Fitness Value**	**No. of Issues (Multi-issues)**	**Total Negotiation Time**	**Communication Overhead**	**Utility/Pay-off Function**	**Cost Function**	**Negotiation Redundancy**	**Multi-point Characteristic**	**Reward Function**	**Negotiation State/Rounds**	**Discount/Concession Function**
CNM	Market driven / oriented	√			√			√					√						
ACT	Middle, random, and heuristic selection	√		√	√								√						
NFA	Alternative					√	√				√		√						
LoM2HiS	Cost function		√											√					
DSNF	Reinforcement learning	√									√		√						
EBM	Evolutionary	√						√	√	√			√					√	
CSP	Practical approach	√						√											√
ISN	Computational or Modeling										√						√		
Proposed ADSLANF	Similarity and Gale Shapely	√									√	√	√		√	√			

The cloud broker based NFA was proposed for nominating the best service provider, using the broker negotiation strategy. In the context of pre-request optimization, an intermediate multi-broker based CNM is proposed, to automate the cloud service negotiation using the market-driven, bargaining position estimation and time dependent concession strategies. In the similar way, an Agent-based Cloud Testbed (ACT) model was designed with the price and time-slot utility function for making concurrent proposals during the negotiation process, using trade-off, linear, conciliatory, and conservative concession strategies [33]. A novel Low-level Metrics to High-level SLA (LoM2HiS) framework is proposed, for managing the cloud application scheduling and provisioning, using the cost based approach [34]. In the Distributed SLA Negotiation Framework (DSNF) model, a reinforcement learning approach is used to provide optimization solution for the managemnt of the inter-network service provider SLA negotiation problem [35]. The above frameworks have the advantage of generating multiple concurrent proposals, and the limitation of concurrent negotiation from each negotiating agent to all other opponent agents, without delay in the pre-request optimization context.

In the context of long-term optimization, a validation of multi-issue negotiation process is carried out in the Evolutionary Bargaining Model (EBM) using the subgame perfect equilibrium strategies. Later, it is extended to the competitive cloud market scenario for getting a fair payoff (utility value) and bargaining opportunity to the opponents [36]. The CSP model is proposed for Quality of Service negotiation between the consumer and the composite service provider. Here, the intermediate composite service provider concurrently negotiates with multiple providers, for selecting each atomic service with higher utility value. An Agent based cloud service composition architecture is proposed, using the broker agent, to provide a single virtualized service to the consumer, which involves composing services such as persistent virtualized service, and vertical (integrating homogeneous service) and horizontal services (integrating heterogeneous service) [37]. In order to provide a long-term benefit to the negotiation party, an Inter-carrier SLA Negotiation (ISN) model is proposed, using the reinforcement learning strategy for optimizing the number of negotiation rounds [38]. The above frameworks in the long-term context, can optimize the negotiation strategy with respect to the number of negotiation rounds, which can reduce the TNT and CO to some extent. These strategies cannot optimize the redundant and multi-point negotiation characteristic, associated with each broker agent.

The detailed information about all the above negotiation framewroks is classified, with respect to the negotiation approach, performance measure attribute, and the optimization context as shown in **Table 1**. In spite of these recent advancements in concession, market driven agent, and other negotiation strategies, the existing frameworks can only optimize the negotiation strategy to some extent, with less TNT and CO. In the literature, there is a lack in optimizing the redundant and multi-point negotiation characteristic associated in the broker negotiation strategy, as shown in the framework comparison table. In order to overcome the above limitations, an ADSLANF is proposed to formulate the research problem, using the CSM and TNGGSSM approaches which can optimize the redundant and multi-point characteristic negotiation, and reduce the TNT and CO to much lesser values. The basic notations used for the formulation of the proposed negotiation model are given in **Table 2**.

Table 2. Basic notations used in the negotiation model

Notation	Description
C_X	Composite service $C_{X\in(1,n)}$ is the combination of multiple atomic service $A_1, A_2, ..., A_k$
$SCA_{X\in(1,n)}$	Service Consumer Agent
$ITBA_{X\in(1,n)}$	Intelligent Third-party Broker Agent
$SPA_{Z\in(1,m)}$	Service Provider Agent
$AC_{Y\in(1,k)}$	Agent Controller

$AAC_{Z\in(1,m)}$	Additional Agent Controller
$TNT(C_X)$	Total Negotiation Time during the negotiation of C_X
$CO(C_X)$	Communication Overhead during the negotiation of C_X
Φ	Expected strategy delay time of negotiation party
λ	Expected communication time between negotiation parties
$\Psi(ITBA_X)$	Expected service lookup time of the broker agent
I_{CB}	Number of interactions occur between the SCA_X and $ITBA_X$
$I_{BP}(ITBA_X, SPA_Y)$	Summation of number of initial $I_{BP}^{Initial}$, actual I_{BP}^{Actual} and final I_{BP}^{Final} interactions occur between the $ITBA_X$ and SPA_Y
A_{tot}	Total number of atomic services involved in the negotiation process
A_{FN}	Number of atomic services optimized for final negotiation
A_{SGroup}	Number of similar service groups involved in A_{tot}
$A_{NSGroup}$	Number of non-similar atomic service groups involved in A_{tot}
ρ_{PorCP}^{T}	Set of negotiable attributes $X_1^{\tau}, X_2^{\tau}, ..., X_n^{\tau}$ of the $\rho_P or \rho_{CP}$ with threshold limit $(\tau_{\min}, \tau_{\max})$
$PL(SCA_{X\in(1,n)})$	Preference List in SCA perspective is the set of opponent $PL(SPA_1), PL(SPA_2), ..., PL(SPA_m)$ with rank preference
$PL(SPA_{Z\in(1,m)})$	Preference List in SPA perspective is the set of opponent $PL(SCA_1), PL(SCA_2), ..., PL(SCA_n)$ with rank preference
$\underset{\bullet}{\bigcup}{}^{t}_{NP}(\rho)$	Total utility function of the $\rho_P or \rho_{CP}$
$\bigcup_{NP}(X_i)$	Utility value of attribute X_i with respect to negotiation party
$W(X_i)$	Weight of attribute X_i with respect to negotiation party
$Sim(A^a, B^a)$	Similarity between the attribute $a \in \{SNA, SRA, SNPA\}$ of composite service A and B
W_a	Probability weight of the attribute $a \in \{SNA, SRA, SNPA\}$
$f(A^a \cap B^a)$	Similarity function between the objects based on common features
$f(A^a - B^a)$	Similarity function between the objects based on distinctive features
$SGroup$	Similar atomic service Group
$NSGroup$	Non Similar atomic service Group
$IGroup$	Individual atomic service Group
NG	Negotiation Group
CNG_X	Concurrent Negotiation Group

3. Problem Formulation and Measurable Performance Metrics

In general, a composite service negotiation using the broker strategy involves the decomposition and management of complex atomic services. There is a need to optimize the negotiation strategy for reducing the TNT and CO among the negotiating parties (SCAs, ITBAs and SPAs). The composite service requests from the $SCAs = \{SCA_1, SCA_2, ..., SCA_n\}$ are represented as $C = \{C_1, C_2, ..., C_n\}$, and each composite service is the combination of multiple atomic services represented as $C_X = \{A_1, A_2, ..., A_k\}$ where $X \in (1, n)$. These services undergo concurrent negotiation with the set of $SPAs = \{SPA_1, SPA_2, ..., SPA_m\}$ through the intermediate $ITBAs = \{ITBA_1, ITBA_2, ..., ITBA_n\}$, which negotiate the service on behalf of the SCAs. Here, the $ITBA_X$ negotiates the C_X as a multiple atomic service A_Y, through an Agent Controller AC_Y, which can perform concurrent negotiations with the SPA_Z using the Additional Agent Controller AAC_Z, where $Y \in (1, k)$ and $Z \in (1, m)$. In this scenario, the average performance of the negotiation framework can be measured through the TNT and CO of the respective C_X. Hence, the objective function is to minimize the TNT and CO of the C_X negotiations involved in the proposed ADSLANF. Formally, it can be defined as shown in equation (1).

$$Min_{X \in (1,n)} TNT(C_X) \,||\, CO(C_X) \qquad (1)$$

In the many-to-many indirect negotiation model, first the TNT function can be minimized by changing the operational characteristic of the AC_Y from single-point to multi-point. The current state-of-the-art makes the AC_Y to perform sequential negotiation with all the SPAs, through the single-point characteristic, which increases the number of TNTs, as shown in equation (2), where Φ, λ and ψ represent the expected strategy delay time, expected communication time and expected service lookup time respectively.

$$\begin{aligned} TNT(C_X) &= \Phi(SCA_X) + 2 \times \lambda(SCA_X, ITBA_X) + \\ & k \times \sum_{Y=1}^{m} \{\Psi(ITBA_X) + \Phi(ITBA_X) + 2 \times \lambda(ITBA_X, SPA_Y) + \Phi(SPA_Y)\} \end{aligned} \qquad (2)$$

Then, ‘k’ denotes the number of atomic services present in C_X managed by the respective $ITBA_X$. In the existing multi-point characteristic, AC_Y concurrently negotiates with the SPAs through the single AC for reducing the number of TNTs with the expected concurrent delay time $\Phi(AC)$, as shown in equation (3).

$$\begin{aligned} TNT(C_X) &= \Phi(SCA_X) + 2 \times \lambda(SCA_X, ITBA_X) + k \times \{\Psi(ITBA_X) + \\ & \Phi(ITBA_X) + 2 \times \lambda(ITBA_X, SPA_Y) + \Phi(SPA_Y) + \Phi(AC_X)\} \end{aligned} \qquad (3)$$

In the proposed approach of the multi-point characteristic, the AC_Y negotiates concurrently with the SPAs through the AAC_Z, which reduces the number of TNTs, as shown in equation (4).

$$\begin{aligned} TNT(C_X) &= \Phi(SCA_X) + 2\times\lambda(SCA_X, ITBA_X) + \Psi(ITBA_X) \\ &+ \Phi(ITBA_X) + 2\times\lambda(ITBA_X, SPA_Y) + \Phi(SPA_Y) \end{aligned} \tag{4}$$

Next, the CO function can be minimized through two stages of optimization, using the CSM and TNGGSSM algorithms respectively. The CO represents the total number of Interactions (I) or communications that occur between the agents of sub-containers, like the Service-Consumer, Intelligent-Third-party-Broker and the Service-Provider.

The value of CO can be computed as the summation of the interaction between the SCA and ITBA, and ITBA and SPA, as shown in equation (5), where I_{CB} denotes the number of interactions that occur between the SCA_X and $ITBA_X$, and I_{BP} denotes the number of interactions that occur between the $ITBA_X$ and SPA_Y.

$$CO(C_X) = I_{CB}(SCA_X, ITBA_X) + I_{BP}(ITBA_X, SPA_Y) \tag{5}$$

I_{BP} can be computed based on the negotiation rounds as shown in equation (6), where $I_{BP}^{Initial}$, I_{BP}^{Actual} and I_{BP}^{Final} represent the number of interactions that occur during the initial, actual and final negotiation rounds respectively.

$$I_{BP} = I_{BP}^{Initial}(ITBA_X, SPA_Y) + I_{BP}^{Actual}(ITBA_X, SPA_Y) + I_{BP}^{Final}(ITBA_X, SPA_Y) \tag{6}$$

The first level of optimization is initiated, before the starting of the negotiation process between the $ITBA_X$ and SPA_Y. The second level of optimization is initiated, after the initial round $(I_{BP}^{Initial})$ of negotiation between the $ITBA_X$ and SPA_Y.

In order to model the above scenario, first the number of atomic services involved in the negotiation process can be represented as $As = \{A_{11}, A_{12}, ..., A_{XY}\}$, where $X \in (1, n)$ and $Y \in (1, k)$. Then, the total number of atomic services (A_{tot}) involved in the negotiation process computes the sum of the atomic services present in each composite service, as shown in equation (7), where $n(As)$ denotes the number of elements in the set As.

$$A_{tot} = n(As) = \sum_{X=1}^{k}\sum_{Y=1}^{k} A_{XY} \tag{7}$$

Consider, ‘P’ as the number of non-similar atomic service groups represented by $A_{NSGroup} = \{A_{XY}^1, A_{XY}^2, ..., A_{XY}^P\}$, where $A_{XY}^P \in As$. Here, the SM Algorithm is used to identify the Q number of similar service groups in AS denoted as $A_{SGroup} = \{A_{Group}^1, A_{Group}^2, ..., A_{Group}^Q\}$

where each $A_{Group}^{i\in(1,Q)} = \{A_{XY}^{1}, A_{XY}^{2}, ..., A_{XY}^{R}\}$ contains 'R' similar atomic services. Then, the set of final negotiable services (A_{FN}) is the combination of all $A_{NSGroup}$ services, along with one service from each $A_{Group}^{i\in(1,Q)}$ as shown in equation (8), where $j \in (1, R)$ and $n(A_{FN})$ denote the number of atomic services optimized for the Final Negotiation (FN) by the CSM algorithm.

$$A_{FN} = \{A_{SGroup}, A_{XY}^{j} \in A_{Group}^{1}, A_{XY}^{j} \in A_{Group}^{2}, ..., A_{XY}^{j} \in A_{Group}^{Q}\} \tag{8}$$

This is the first stage of optimization initiated before the starting of the trial negotiation round, which minimizes the CO function to some extent. To further minimize the CO function, the second stage of optimization is initiated after the trial negotiation round, using the TNGGSSM algorithm.

The ITB negotiation strategy reduces the TNT and CO among the negotiation parties using concurrent negotiation in the AC, and exploiting the optimization approaches (CSM and TNGGSSM) in the negotiation strategy. This can be made possible by automating the negotiation process that takes place among the negotiation parties. The automation of the SCAs, ITBAs and SPAs in the respective multi-agent system can be simulated, using the Java Agent Development (JADE) framework tool [39]. Further, the multi-agent system can be integrated with the real private cloud environment, using the Eucalyptus tool [40], for the negotiation of real time cloud commerce applications. The negotiation process in the automated cloud application follows the Agent Communication Language.

4. Many-to-Many Composite Service Negotiation Model

In order to demonstrate the many-to-many composite service negotiation model, a novel ADSLANF is proposed for the multi-attribute negotiation process among the SCAs, ITBAs and SPAs. This proposed ADSLANF exploits the similarity matching and gale shapely stable matching approach for obtaining the optimal broker negotiation strategy, which reduces the TNT and CO of the model. The above framework formulation and its methodology are explained in the following subsections.

4.1 Automated Dynamic SLA Negotiation Framework

An Automated Dynamic SLA Negotiation Framework is proposed, for composite service negotiation between many-to-many negotiating parties, and its architecture is shown in **Fig. 1**. This framework includes the negotiation process involved between 'n' number of SCs and 'm' number of SPs through the ITB, using the multi-agent system. To automate the negotiation process, agents such as the SCAs, ITBAs and SPAs are used to negotiate on behalf of the SCs, ITB and SPs respectively. The SCAs, ITBAs and SPAs are simulated in the sub containers like the Service-Consumer, Intelligent-Third-party-Broker, and Service-Provider respectively. Initially, all the SPAs will publish their services with the specialized agent called the Directory Facilitator (DF), which is a yellow page service available in the JADE platform. Then, the SCAs negotiate with the ITBAs, which in turn, look up the appropriate SPAs in the DF, and start the negotiation process with the concerned SPAs, through the AAC (controlled by the AC). During this time, the negotiation strategy is used by the respective negotiating parties, to support the automated negotiation with varying parameters in several rounds of a Proposal

(ρ_P) and Counter Proposal (ρ_{CP}). The strategy will exploit certain protocols for making negotiation among the parties, and follow a set of rules in its operational behavior for better trade-off and concession. This strategy, in turn, requires the time interval of negotiation, and the threshold limit of negotiation parameters to take a decision in the operational behavior.

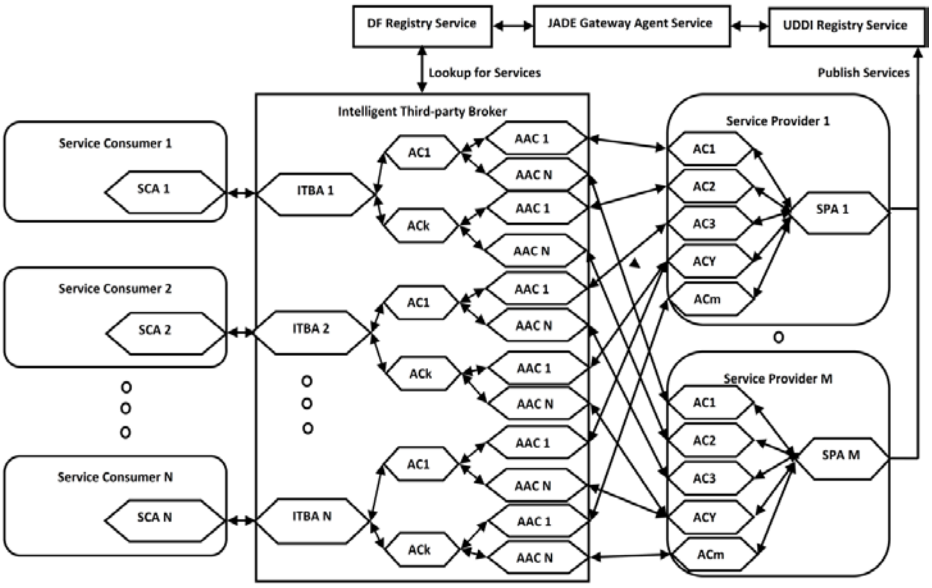


Fig. 1. Architecture of Automated Dynamic SLA Negotiation Framework

In the proposed ADSLANF, the FIPA-CNP is extended with the feature of the Alternate Offer Protocol for realizing the negotiation among the parties. Initially, the SCAs will send the proposal request to the ITBAs, which in turn, will negotiate with all the SPAs on behalf of the SCAS and forward the uncommitted or committed agreements to the SCAs as a reply. In the above process, the actual negotiation will take place through the generation of (ρ_P) and (ρ_{CP}), by the negotiation strategy of the ITBAs and SPAs respectively, during the time interval $T_{t\in(1,K)}$. This negotiation strategy generates various rounds of ρ_P and ρ_{CP} as shown in equation (9), based on the threshold limit $(\tau_{\min}, \tau_{\max})$ of the negotiable parameters.

$$\rho_{PorCP}^{T} = \{\rho_P^{T_1}, \rho_{CP}^{T_2}, \rho_P^{T_3}, \rho_{CP}^{T_4},..., \rho_P^{T_k}\} \qquad (9)$$

Let $\rho_P^{T_t} or \rho_{CP}^{T_t} = \{X_1^{\tau}, X_2^{\tau},..., X_n^{\tau}\}$ be the set of negotiable attributes of the ρ_P or ρ_{CP}, with respect to the threshold limit τ. The threshold limits suggested by the SCs and SPs to the SCAs and SPAs are represented as $(\tau_{\min}^{SCA}, \tau_{\max}^{SCA})$ and $(\tau_{\min}^{SPA}, \tau_{\max}^{SPA})$ respectively.

The novel idea of this research work includes the optimization of the TNT and CO, with respect to composite service negotiation among the parties in a specified T. To do this, the composite services involved in the negotiation process are represented as $C = \{C_1, C_2,..., C_n\}$, and their corresponding Negotiation Rounds (NR) involved in any C_X are classified into initial $NR_{Initial}$, actual NR_{Actual} and final NR_{Final}, where $X \in (1,n)$. First, the TNT function can be optimized in C_X through the concurrent negotiation of ‘k’ multiple atomic services, using multiple AC_Y, which in turn, concurrently negotiate with multiple SPA_Z through the co-ordination of multiple AAC_Z, where $Y \in (1,k)$ and $Z \in (1,m)$. This will minimize the TNT function, using equation (4) as shown in section 2, by optimizing the single-point negotiation used in AC_Y, to the multipoint negotiation characteristic. Next, the CO function can be optimized, using the equation (5), as shown in section 2, and by two levels of optimization in the negotiation strategy, by using the CSM and TNGGSSM approach. The first level of optimization is applied before the $NR_{Initial}$, using the CSM algorithm, shown in Section 3.2. This will minimize the CO to a considerably less number of interactions by avoiding the atomic service redundancy among the composite services available for the negotiation process.

After the $NR_{Initial}$, a set of PLs is generated for the opponent negotiating parties, from the perspective of the SCAs and SPAs, as shown in equations (10) and (11).

$$PL(SCA_X) = \{PL(SPA_1), PL(SPA_2), ..., PL(SPA_m)\} \tag{10}$$

$$PL(SPA_Z) = \{PL(SCA_1), PL(SCA_2), ..., PL(SCA_n)\} \tag{11}$$

The $PL(SCA_{X\in(1,n)})$ contains the list of negotiating parties with a higher rank preference to the SCA_1 and lower rank preference to the SPA_m, which have the maximum and minimum total utility functions respectively. In a similar manner, the $PL(SPA_{Z\in(1,m)})$ ranks the SCA_1 to SCA_m. This total utility function of the multi-attribute ρ_P and ρ_{CP} can give the different levels on PL (high or low) from the perspective of the Negotiating Party (NP), such as the SCA and SPA. During the negotiation process, the NPs will give different preferences over the utility value $\bigcup_{NP}(\rho_i)$ and weigh the $W(\rho_i)$ of each attribute. The negotiated attributes $(X_1, X_2, ..., X_n)$ of one’s ρ_P or ρ_{CP} may give a high utility value to one party and a low utility value to another, and vice versa. Sometimes, they may give equal utility value to both the parties. Using the above information, the total utility function $\underset{\bullet}{\bigcup}{}^{t}_{NP}(\rho)$ of the ρ_P or ρ_{CP} is computed by using equation (12).

$$\underset{\bullet}{\bigcup}{}^{t}_{NP}(\rho) = \sum_{i=1}^{n} W(X_i) \times \bigcup_{NP}(X_i) \tag{12}$$

Here, the total weight of the attribute should be equal to one; i.e., $[W(X_1) + W(X_2) + ..., W(X_n) = 1]$ and the utility value of the NP should range from zero to one; i.e., $\bigcup_{NP}(X_i) \in (0,1)$. Here, the utility value ‘1’ denotes full satisfaction and ‘0’ denotes dissatisfaction with the NPs. The Second level of optimization can be applied after the

$NR_{Initial}$ or before the NR_{Actual}, using the TNGGSSM algorithm, as shown in Section 4.3. This algorithm will optimize the multiple one-to-many negotiation strategy involved in the process, to a multiple one-to-one negotiation strategy, based on the PL generated by the SCAs and SPAs. Hence, the novel idea of the optimal ITB strategy in the proposed ADSLANF, helps in achieving less TNT and communication overhead among the NPs.

4.2 Classified Similarity Matching Decision Approach

The CSM approach focuses on the scenario where the broker negotiation strategy is needed to be optimized, for the composite service negotiation among the NPs. This optimization will overcome the problem of redundancy in composite services negotiation addressed in the above subsection. Here, the objective is to avoid the multiple redundant or similar negotiations among the NPs, by grouping various similar services into clusters. Here, each cluster is treated as a single service negotiation, instead of a multiple similar service negotiation. To do this, a Taversky's ratio model can be used to identify only the similarity between two objects, which constitute the whole attribute (feature) of objects. This can be applicable for deterministic matching, where there is no need of any preference and classification of the attributes of objects. It may not be applicable where the NPs are not aware of the entire negotiation attribute in advance, and their ρ_P or ρ_{CP} needs to have some classification and preference over the attributes for meeting the expectations of their opponents. In addition, the negotiation attribute present among the NPs may have complete (certain), partial (risk) and incomplete (uncertain) information. So, the CSM approach is proposed to extend Taversky's ratio model, with the probabilistic approach of similarity matching. In this approach, the set of service attributes can be classified as the Service Name Attribute (SNA), Service Requirement Attribute (SRA), and Service Negotiable Parameter Attribute (SNPA). In this connection, the similarity among the composite services with complete information can be computed by summing the similarity of the classified service attributes, with its equal probability weight as shown in equation (13). In the case of incomplete and uncertain information, the probability of the classified attribute has to be changed based on the preference of the NPs.

$$\begin{aligned} Sim(A,B) = {} & W_{SNA} \times Sim(A^{SNA}, B^{SNA}) + W_{SRA} \times Sim(A^{SRA}, B^{SRA}) + \\ & W_{SNPA} \times Sim(A^{SNPA}, B^{SNPA}) \end{aligned} \tag{13}$$

The value of $W_{SNA} = W_{SRA} = W_{SNPA} = 1/3$, and the similarity ratio of the corresponding service attributes can be computed, as shown in equations (14), (15) and (16), which may range from 0 to 1. Here the value '0' denotes no (low) similarity, '1' denotes the correct (high) similarity, and the remaining values that range between 0 and 1 represent partial similarity.

$$Sim(A^{SNA}, B^{SNA}) = \frac{f(A^{SNA} \cap B^{SNA})}{\begin{array}{l} \theta \times f(A^{SNA} \cap B^{SNA}) + \alpha \times f(A^{SNA} \cap B^{SNA}) + \\ \beta \times f(A^{SNA} \cap B^{SNA}) \end{array}} \tag{14}$$

$$Sim(A^{SRA},B^{SRA})=\frac{f(A^{SRA}\bigcap B^{SRA})}{\theta\times f(A^{SRA}\bigcap B^{SRA})+\alpha\times f(A^{SRA}\bigcap B^{SRA})+\beta\times f(A^{SRA}\bigcap B^{SRA})} \tag{15}$$

$$Sim(A^{SNPA},B^{SNPA})=\frac{f(A^{SNPA}\bigcap B^{SNPA})}{\theta\times f(A^{SNPA}\bigcap B^{SNPA})+\alpha\times f(A^{SNPA}\bigcap B^{SNPA})+\beta\times f(A^{SNPA}\bigcap B^{SNPA})} \tag{16}$$

The values of [$\theta=1$ and $\alpha=\beta=0$] determine the similarity between the objects based on common features, and [$\theta=0$ and $\alpha=\beta=1$] determines the similarity between the objects based on distinctive features. The similarity decision can be classified into three types, such as decisions taken under certainty, uncertainty and risk, based on their respective high, low and partial similarity ratios. The above sequence of operations is generalized into the CSM approach, as shown in Algorithm 1.

Algorithm 1 Classified Similarity Matching

```
Begin
Initialize $X\leftarrow(1,n)$ and $Y\leftarrow(1,k)$
$C\leftarrow\{C_1,C_2,...,C_n\}$ be the set of composite services
$C_X\leftarrow\{A_1,A_2,...,A_k\}$ be the set of atomic services in $C_X$
$A_S\leftarrow\{A_{11},A_{12},...,A_{XY}\}$ be the set of atomic services in C and $C_X\in As,\forall X\in(1,n)$
for all $C_X\in C$ do
Iteration 1:
  Assume $i\leftarrow 1$, $C_i\in C_1$ and $C_j\in\{C_2,C_3,...,C_n\}$
  Assume $SGroup1[g\leftarrow 1]\leftarrow\{\}$ and $NSGroup1[h\leftarrow 1]\leftarrow\{\}$
G1:Assume $C_i\leftarrow\{A_1,A_2,...,A_k\}$ and $C_j\leftarrow\{A_1,A_2,...,A_k\}$
  Assign $A_u\leftarrow\{X_1,X_2,...,X_r,...,X_p\},\forall u\in(1,k)$ and $A_u\in C_i$
  Assign $A_u^{SNA}\leftarrow\{X_1\}$, $A_u^{SRA}\leftarrow\{X_2,...,X_r\}$ and $A_u^{SNPA}\leftarrow\{X_{r+1},...,X_p\}$
  Assign $A_v\leftarrow\{X_1,X_2,...,X_r,...,X_p\},\forall v\in(1,k)$ and $A_v\in C_j$
  Assign $A_v^{SNA}\leftarrow\{X_1\}$, $A_v^{SRA}\leftarrow\{X_2,...,X_r\}$ and $A_v^{SNPA}\leftarrow\{X_{r+1},...,X_p\}$
  for $u\leftarrow 1;u\le k;u++$ do
   for $j\leftarrow 1;j\le n-1;j++$ do
    for $v\leftarrow 1;v\le k;v++$ do

$Sim(A^{SNA},B^{SNA})=\frac{f(A^{SNA}\bigcap B^{SNA})}{\theta\times f(A^{SNA}\bigcap B^{SNA})+\alpha\times f(A^{SNA}\bigcap B^{SNA})+\beta\times f(A^{SNA}\bigcap B^{SNA}}$
```

$$Sim(A^{SRA},B^{SRA})=\frac{f(A^{SRA}\cap B^{SRA})}{\theta\times f(A^{SRA}\cap B^{SRA})+\alpha\times f(A^{SRA}\cap B^{SRA})+\beta\times f(A^{SRA}\cap B^{SRA}}$$

$$Sim(A^{SNPA},B^{SNPA})=\frac{f(A^{SNPA}\cap B^{SNPA})}{\theta\times f(A^{SNPA}\cap B^{SNPA})+\alpha\times f(A^{SNPA}\cap B^{SNPA})+\beta\times f(A^{SNPA}}$$

$$Sim(A,B)=W_{SNA}\times Sim(A^{SNA},B^{SNA})+W_{SRA}\times Sim(A^{SRA},B^{SRA})+W_{SNPA}\times Sim(A^{SNP}$$

```
        $SR \leftarrow Sim(A_u \in C_i, A_v \in C_j)$
        if $SR \leftarrow 1$ then
          $Group[g] \leftarrow Group[g] \cup \{A_u, A_v\}$
        else
          $NS[h] \leftarrow \{A_u\}$
          $h \leftarrow h+1$
        end if
      end for
    end for
    $g \leftarrow g+1$
  end for
  return $(Group1[g] \leftarrow Group[g], NSGroup1[h] \leftarrow NS[h])$
-------------
Iteration N:
  Assume $i \leftarrow 1$, $C_i \in C_n$ and $C_j \in \{C_1, C_3, ..., C_{n-1}\}$
  Assume $SGroupN[g \leftarrow 1] \leftarrow \{\}$ and $SGroupN[h \leftarrow 1] \leftarrow \{\}$
  GOTO G1
  return $(GroupN[g] \leftarrow Group[g], NSGroupN[h] \leftarrow NS[h])$
end for
$IGroup[N] = \left\{ \begin{array}{l} \exists A_i \in Group1[g], ..., \exists A_i \in GroupN[g], \\ \forall A_i \in NSGroup1[g], ..., \forall A_i \in NSGroupN[g] \end{array} \right\}$
return (SGroup[N], NSGroup[N], IGroup[n])
End Algorithm
```

The proposed CSM algorithm identifies the various groups of atomic service similarity present among the available Cs, and returns the array of Similar atomic service Groups $(SGroup[N])$, Non Similar atomic service Groups $(NSGroup[N])$ and Individual atomic service Groups $(ISGroup[N])$. The $SGroup[N]$ can be generated from each array similarity, which contains the list of similarity groups, such as $SGroup[x]=\{GroupX[1], GroupX[2], ..., GroupX[M]\}$. Then, the total similarity group available among the Cs can be represented as the combination of all the $SGroup$ sets as shown in equation (17).

$$SGroup = \begin{Bmatrix} Group1[1],...,Group1[M],...,Group2[1],..., \\ Group2[M],...,GroupN[1],...,GroupN[M] \end{Bmatrix} \tag{17}$$

Next, the total Individual atomic service group $(IGroup)$ is additionally generated by the proposed CSM approach, as shown in equation (18). It is the combination of one atomic service from each $SGroup$ set, and all atomic services present in the $NSGroup$.

$$IGroup[N] = \begin{Bmatrix} \exists A_i \in Group1[g],...,\exists A_i \in GroupN[g], \\ \forall A_i \in NSGroup1[g],...,\forall A_i \in NSGroupN[g] \end{Bmatrix} \tag{18}$$

This CSM approach is applied before the ITB starts the decision of the actual negotiation round. As a result of the similarity matching, the ITBAs start the negotiation of composite services by assigning all the dissimilar atomic services to the individual ACs, and the similar atomic services of several groups to the corresponding shared AC. So, this approach impacts the proposed ADSLANF by avoiding the redundant service negotiation among the composite services available in the ITB.

4.3 Truncated Negotiation Group Gale Shapely Stable Matching approach

The existing Gale Shapely approach can find the optimal stable matching pairs for an equal number of parties, without any redundancy in their PL. So this approach cannot be directly applied to the proposed ADSLANF, which consists of an unequal number of NPs, with redundancy in the PL of the SCAs and SPAs. To overcome the restriction in the above problem, a TNGGSSM approach is proposed to form a Negotiation Group (NG) from the $IGroup$, and to extend the Gale Shapely approach, with the truncation operation for optimizing the CO among the NPs as shown in Algorithm 2. First, the NG is generated as the set of multiple Concurrent Negotiation Groups (CNG), as shown in equation (19).

$$NG = \begin{Bmatrix} CNG_1[A_1^{SCA_1}, A_2^{SCA_2},..., A_n^{SCA_n}], CNG_2[A_1^{SCA_1}, A_2^{SCA_2},..., A_{n-i}^{SCA_{n-i}}],..., \\ CNG_N[A_1^{SCA_1}, A_2^{SCA_2},..., A_{n-j}^{SCA_{n-j}}] \end{Bmatrix} \tag{19}$$

Here, assume that the size and number of atomic services in each CNG does not exceed the size of the NPs, and assume that no two services from the same SCAs form the CNG. Then each CNG is assigned to the respective ITBA, which computes the utility value of ρ_P or ρ_CP generated for each atomic service, using equation (12). Next, the Gale Shapely approach is extended with the truncation operation, to provide the same number of SPAs for the number of atomic services present in the CNG. So, the proposed TNGGSSM approach generates separate preference lists of SCAs and SPAs with respect to CNGs, and then truncates its SPAs which have less utility values.

Algorithm 2 Truncated Negotiation Group Gale Shapely Stable Matching

```
Begin
Initialize $IGroup[N]=\left\{\begin{matrix} \exists A_i \in Group1[g],...,\exists A_i \in GroupN[g], \\ \forall A_i \in NSGroup1[g],...,\forall A_i \in NSGroupN[g] \end{matrix}\right\}$
Generate $G \leftarrow \{CNG_1[A_1^{SCA_1}, A_2^{SCA_2},..., A_n^{SCA_n}],..., CNG_N[A_1^{SCA_1}, A_2^{SCA_2},..., A_{n-j}^{SCA_{n-j}}]\}$ from $IGroup$
Assume $n(CNG_X) \le n(SCA_X)$ and $n(G) \le n(SCA_X)$
While ( $n(G) \ne null \,\&\, CNG_{X\in(1,n)}$ negotiation-engaged-pair) do
  Assume the constrain $n(AC_X^{ITBA_X}) = n(AC_Y^{SPA_Y})$
  Assume $\forall AC_{X\in(1,n)}^{ITBA_X}$ do not propose NR>1to same $AC_{Y\in(1,m)}^{SPA_Y}$
  Assume $\forall AC_{Y\in(1,m)}^{SPA_Y}$ do not propose NR>1 to same $AC_{X\in(1,n)}^{ITBA_X}$
  Initialize all $AC_{Y\in(1,m)}^{SPA_Y}$ and $AC_{X\in(1,n)}^{ITBA_X}$ to be free without negotiation-engaged-pair
  Compute the $\underset{\bullet}{\bigcup}{}_{NP}^{t}(\rho \in AC^{NP})$
G1:Generate $PL(AC_X^{ITBA_X}) = \{AC_{Y\in(1,m)}^{SPA_1}, AC_{Y\in(1,m)}^{SPA_2},..., AC_{Y\in(1,m)}^{SPA_m}\}$
  Generate $PL(AC_Y^{SPA_Y}) = \{AC_{X\in(1,n)}^{ITBA_1}, AC_{X\in(1,n)}^{ITBA_2},..., AC_{X\in(1,n)}^{ITBA_n}\}$
  if $n(AC_X^{ITBA_X}) = n(AC_Y^{SPA_Y})$ then
   While $AC_{i\in X}^{ITBA_X}$ is free in $AC_X^{ITBA_X}$ do
    Choose the free $AC_{i\in X}^{ITBA_X}$
    Check the highest ranked $AC_Y^{SPA_Y}$ from $PL(AC_{i\in X}^{ITBA_X})$ where not proposed
    Let $AC_{i\in Y}^{SPA_Y}$ be the highest ranked $AC_Y^{SPA_Y}$
          if $AC_{i\in Y}^{SPA_Y}$ is free then
           Set $(AC_{i\in X}^{ITBA_X}, AC_{i\in Y}^{SPA_Y})$ as negotiation-engaged-pair
           return (negotiation-engaged-pair)
          else $AC_{i\in Y}^{SPA_Y}$ is engaged to $AC_{j\in X}^{ITBA_X}$ then
     Check the PL of $AC_{j\in X}^{ITBA_X}$
           if $AC_{i\in Y}^{SPA_Y}$ prefer $AC_{i\in X}^{ITBA_X}$ over other $AC_{j\in X}^{ITBA_X}$ in PL then
            Set $AC_{j\in X}^{SCA}$ to be free
           else if $AC_{i\in Y}^{SPA_Y}$ prefer $AC_{j\in X}^{ITBA_X}$ over other $AC_{i\in X}^{ITBA_X}$ in the PL then
            Set $AC_{i\in X}^{ITBA_X}$ to be free
            Set $(AC_{j\in X}^{ITBA_X}, AC_{i\in Y}^{SPA_Y})$ negotiation-engaged-pair
            return (negotiation-engaged-pair)
           end if
   end while
```

```
    else if $n(AC_X^{ITBA_X}) > n(AC_Y^{SPA_Y})$ then
     while $n(AC_Y^{SPA_Y}) \neq n(AC_X^{ITBA_X})$ do
      Truncate any $AC_{i\in X}^{ITBA_X} \leftarrow Min_{i\in(1,n)}\left\{\bigcup_{\bullet} {}^{t}_{ITBA_X}(\rho \leftarrow AC_{i\in X}^{ITBA_X})\right\}$
     end while
     GOTO G1
    else $n(AC_X^{ITBA_X}) < n(AC_Y^{SPA_Y})$ then
     while $n(AC_Y^{SPA_Y}) \neq n(AC_X^{ITBA_X})$ do
      Truncate any $AC_{j\in Y}^{SPA_Y} \leftarrow Min_{j\in(1,m)}\left\{\bigcup_{\bullet} {}^{t}_{SPA_Y}(\rho \leftarrow AC_{j\in Y}^{SPA_Y})\right\}$
     end while
     GOTO G1
    end if
        return (negotiation-engaged-pair)
    end while
End Algorithm
```

After the truncation, all the CNGs have an equal number of NPs with respect to their atomic services, and each CNG follows the Gale Shapely stable matching individually. In the $CNG_{X\in(1,N)}$, each atomic service is assigned to the ACs of $ITBA_{X\in(1,n)}$ for concurrent negotiation with the ACs of $SPA_{Y\in(1,m)}$ respectively. Then, the $ITBA_X$ strategy takes the decision to reduce the multi-point negotiation in its ACs (supports through AAC) to a single-point, by generating the preference over the ACs of SPA_Y. This decision, first generates the $PL(AC_X^{ITBA_X})$ as shown in equation (20), and denotes the rank list of $AC_Y^{SPA_Y}$, which gives a high rank preference to the opponents having high total utility values.

$$PL(AC_X^{ITBA_X}) = \left\{AC_{Y\in(1,m)}^{SPA_1}, AC_{Y\in(1,m)}^{SPA_2}, \ldots, AC_{Y\in(1,m)}^{SPA_m}\right\} \tag{20}$$

$$PL(AC_Y^{SPA_Y}) = \left\{AC_{X\in(1,n)}^{ITBA_1}, AC_{X\in(1,n)}^{ITBA_2}, \ldots, AC_{X\in(1,n)}^{ITBA_n}\right\} \tag{21}$$

Next, it generates the $PL(AC_Y^{SPA_Y})$ in the same way, as shown in equation (21), denoting the rank list of the $AC_X^{ITBA_X}$, which gives a high rank preference to the opponents having high total utility values. Finally, both $PL(AC_X^{ITBA_X})$ and $PL(AC_Y^{SPA_Y})$ undergo the stable matching process, to produce the negotiation-engaged-pairs for one-to-one negotiation. Similarly, the remaining $CNG_{X\in(1,N)}$ undergoes stable matching, and produces the negotiation-engaged-pairs for making one-to-one negotiation in all its ACs. Obtaining the negotiation-engaged-pairs in all the $CNG_{X\in(1,N)}$ present in the ITB, leads to the optimization (minimization) of the CO, in the proposed ADSLANF model.

5. Empirical and Experimental Evaluation

5.1 Experimental Setup

In this research work, an agent based test-bed is created for cloud commerce application using the Eucalyptus and JADE tools. The open source Eucalyptus tool provides the real private cloud setup, in which virtual machines are created for providing a negotiation environment to SC, ITB and SP. Virtual machines containing the operating system images are first bundled with the JADE tool, for the notion of integrating the cloud with the automated agent based service communication. Second, the apache tomcat web server with an axis environment is bundled for cloud service provisioning in the SP. Finally, the web service integration gateway add-on is bundled for bidirectional discovery and invocation, between the JADE and cloud service. A cloud service provides the interface for cloud resource provisioning, like Infrastructure as a Service (IaaS), Platform as a Service (PaaS) and Software as a Service (SaaS). The Web Service Description Language (WSDL) is used to publish and discover the cloud service from the Universal Description Discovery and Integration (UDDI) registry. All the SPs will publish their services in the UDDI, using the Simple Object Access Protocol (SOAP) message.

Once the JADE tool is started, it will provide default agent services such as Remote Management Agent (RMA), Agent Management System (AMS), DF, and JADE gateway agent, running under the control of the main container. The proposed ADSLANF implements the NPs as a set of SCAs, ITBAs and SPAs in the respective Service-Consumer, Third-party-Broker and Service-Provider sub container of the JADE platform. The ITBA contains multiple ACs and AACs for managing the composite service and its concurrent negotiation of atomic services with SPAs. The RMA service monitors and controls the life-cycle of the remote agent in the platform, and the AMS service has supervisory control over the agent platform for accessing the agent. Then, the DF is a yellow page service registry used for publishing and discovering the agent services in the platform, using the Agent Communication Language (ACL) message.

The JADE gateway agent monitors the WSDL service modifications that occur in the UDDI registry, and transparently update the corresponding service in the DF registry. This gateway agent uses the SOAP and ACL messages to communicate with the UDDI and DF registries respectively. First, the SPAs publish their service in the UDDI (updated in DF), and then, the ITBAs look up the available SPAs from the DF service, and follow the negotiation strategy through the AC and AAC. This experimental setup follows the extended FIPA-CNP with the feature of the Alternate Offer Protocol, for interacting with the agents in the JADE platform.

5.2 Composite Service Data Set

In this test-bed, composite service datasets are self generated for cloud commerce application, due to lack of benchmark datasets. Moreover, the proposed research focuses on identifying the similarity in the composite services; it is not possible to collect the relevant dataset from the literature. In cloud commerce applications, services like Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS) are available for negotiation. The precise information of the generated datasets is described as SNA, SRA and SNPA, as shown in **Table 3**.

Table 3. Composite Service Dataset for 4×4 negotiation parties

Composite Service	**Atomic Service**	**SNA**	**SRA**			**SNPA**	
			CPU (GHZ)	**RAM (GB)**	**Storage (GB)**	**Price (Dollar)**	**Time-Slot (Duration)**
C1	A11	IaaS	2	3	100	10-15	10am-11am
	A12	IaaS	3	3	200	15-30	10am-12am
	A13	IaaS	1	1	100	09-11	12am-01pm
	A14	IaaS	3	3	100	20-35	12am-01pm
C2	A21	IaaS	3	3	400	30-40	11am-04pm
	A22	IaaS	2	3	100	10-15	10am-11am
	A23	IaaS	1	2	100	15-20	12am-01pm
	A24	IaaS	3	4	200	25-40	10am-01pm
C3	A31	IaaS	3	3	200	15-30	10am-12am
	A32	IaaS	3	3	400	30-40	11am-04pm
	A33	IaaS	2	2	200	20-15	12am-01pm
	A34	IaaS	3	8	300	40-50	11am-02pm
C4	A41	IaaS	3	3	200	15-30	10am-12am
	A42	IaaS	3	3	400	30-40	11am-04pm
	A43	IaaS	2	3	100	10-15	10am-11am
	A44	IaaS	2	4	500	50-65	12am-04pm

In the 4×4 test-bed, the proposed ADSLANF is experimented with four composite services; each has a combination of four atomic services. Here, the composite service (such as an IaaS request) is used for accessing the infrastructural facility associated with varied time and price over the above dataset. This variation in the specification time and price disintegrates the composite service into four atomic services, with a specific price at a certain time.

5.3 Experimental Observation and Evaluation

The above dataset is used by the SCAs for making negotiation requests to the ITB. First, the ITB identifies the similarity groups among the services, using the CSM algorithm. This algorithm identifies 7 non similar atomic services and 3 similarity groups (each with 3 atomic services) such as SGroup1{A11, A22, A43}, SGroup2{A12, A31, A41}, and SGroup3{A21, A32, A42}, and 10 individual atomic service groups from the above dataset. Instead of 16 atomic services stated in the dataset, only 10 individual atomic services were chosen for further negotiation, using the CSM approach, which reduces the CO to a considerable extent. Next, the TNGGSSM approach takes the input of the above 10 individual atomic services, and generates the NG with 4 CNG such as $CNG_1 = \{A_{11}^{SCA_1}, A_{21}^{SCA_2}, A_{33}^{SCA_3}, A_{44}^{SCA_4}\}$, $CNG_2 = \{A_{12}^{SCA_1}, A_{23}^{SCA_2}, A_{34}^{SCA_3}\}$, $CNG_3 = \{A_{13}^{SCA_1}, A_{24}^{SCA_2}\}$ and $CNG_4 = \{A_{14}^{SCA_1}\}$ respectively. To initiate further negotiation rounds, the atomic services of CNG_1, CNG_2, CNG_3 and CNG_4 are assigned to the $ITBA_1$, $ITBA_2$, $ITBA_3$ and $ITBA_4$ respectively. Each ITBA, needs to undergo the Gale Shapely matching, for identifying the negotiation-engaged-pairs for one-to-one negotiation. For the sake of simplicity, consider the preference list generated by the NPs of CNG_1, as shown in **Table 4**.

Table 4. Preference List of $AC_X^{ITBA_1}$ (SCAs behalf) and $AC_Y^{SPA_Y}$ (SPAs behalf)

	$AC_1^{SPA_1}$	$AC_2^{SPA_2}$	$AC_3^{SPA_3}$	$AC_4^{SPA_4}$
$AC_1^{ITBA_1}(SCA_1)$	1, 1	2, 3	3, 3	4, 2
$AC_2^{ITBA_1}(SCA_2)$	1, 2	2, 1	4, 2	3, 3
$AC_3^{ITBA_1}(SCA_3)$	2, 4	3, 2	1, 4	4, 1
$AC_4^{ITBA_1}(SCA_4)$	2, 3	1, 4	3, 1	4, 4

Here, the preference list of $AC_1^{ITBA_1}$ can be generated as $PL(AC_1^{ITBA_1}) = \{AC_1^{SPA_1}, AC_2^{SPA_2}, AC_3^{SPA_3}, AC_4^{SPA_4}\}$ which denotes the rank preference of the $AC_Y^{SPA_Y}$. In a similar manner, the preference list of $AC_2^{ITBA_1}$, $AC_3^{ITBA_1}$ and $AC_4^{ITBA_1}$ is generated. Next, the preference list of $AC_1^{SPA_1}$ can be generated as $PL(AC_1^{SPA_1}) = \{AC_1^{ITBA_1}, AC_2^{ITBA_1}, AC_3^{ITBA_1}, AC_4^{ITBA_1}\}$, which denotes the rank preference of the $AC_X^{ITBA_1}$. In the similar way, the preference lists of $AC_2^{SPA_2}$, $AC_3^{SPA_3}$ and $AC_4^{SPA_4}$ are generated. After the trial round, the TGSSM algorithm undergoes 6 computational iterations and produces 4 optimal negotiation-engaged-pairs such as $(AC_2^{ITBA_1}, AC_4^{SPA_4})$, $(AC_4^{ITBA_1}, AC_3^{SPA_3})$, $(AC_1^{ITBA_1}, AC_1^{SPA_1})$ and $(AC_3^{ITBA_1}, AC_2^{SPA_2})$ for further negotiation with the pairs. Similarly, the preference lists are generated for CNG_2, CNG_3 and CNG_4 for obtaining the optimal negotiation-engaged-pairs among the CNGs present in the respective ITBAs. Finally, a sniffer agent is created, as shown in **Fig. 2**, for visualizing the negotiation sequence among the $ITBA_X$ and SPAs.

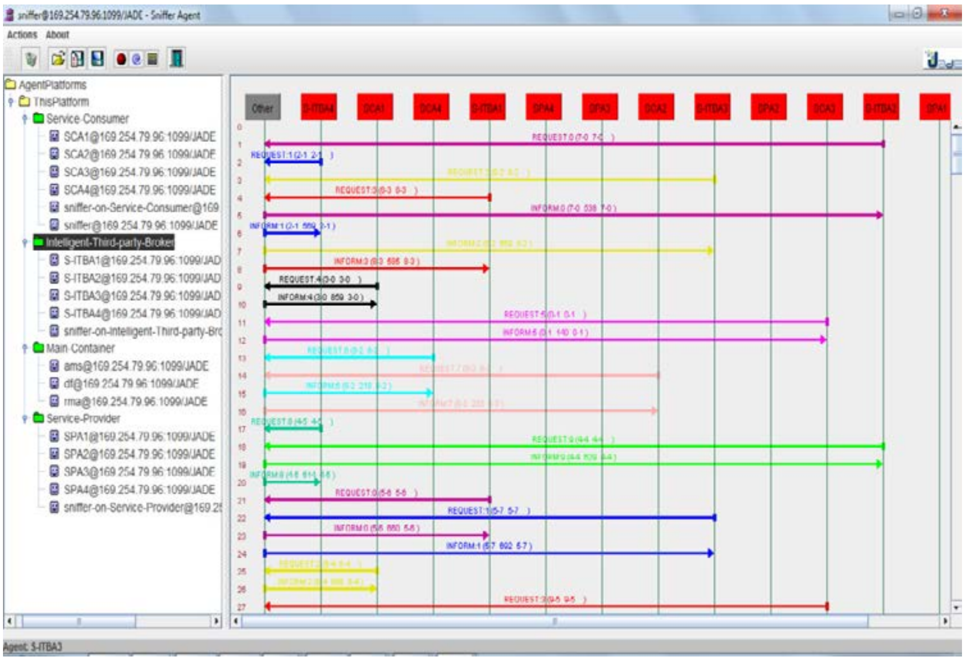


Fig. 2. Visualization of negotiation process using sniffer Agent

From the above observation, 7 non similar atomic services and 3 similarity groups (each with 3 atomic services) lead to 10 concurrent negotiations of atomic services in the ITBAs. Similarly, the experiment is conducted for the 10×10 and 15×15 test-beds, using 10 and 15 composite service datasets respectively. For the sake of simplicity, the abstract information of both the generated datasets is given in descriptive forms. First, the 10×10 test-bed is experimented with 10 composite services, each with 4 atomic services. Here, the observation of 25 non similar atomic services and 5 similar groups (each with 3 atomic services) leads to 30 concurrent negotiations of atomic services in the ITBAs. Next, 15×15 test-bed is experimented with 15 composite services, each with 4 atomic services. Here, the observation of 39 non similar atomic services and 7 similar groups (each with 3 atomic services) leads to 46 concurrent negotiations of atomic services in the ITBAs.

The negotiation results of the 4×4, 10×10 and 15×15 cloud commerce test-beds are quantitatively measured, to identify the distance between the obtained negotiated levels and the optimal ones. The optimality achieved by the proposed ADSLANF, with and without the optimization approach is quantitatively measured in terms of CO, as shown in **Table 5**.

Table 5. Optimality achieved in proposed ADSLANF

Negotiation Framework	CO		
	4×4 **NPs**	10×10 **NPs**	15×15 **NPs**
ADSLANF without Optimality	4104	25620	57630
ADSLANF with CSM approach	2568	19220	44190
ADSLANF with TGSSM approach	1128	3300	5550
ADSLANF with CSM and TGSSM approach	708	2480	4262

In this connection, the performance graph is generated, as shown in **Fig. 3**, to quantitatively evaluate the distance of optimality achievable by the proposed approach. It is clear from the 4×4 test-bed, that the proposed ADSLANF with both optimizations takes the optimal difference from the ADSLANF without optimization, the ADSLANF with the CSM approach, and the ADSLANF with the TNGGSSM approach, in the CO distance of 3396, 1860, and 420 respectively. Next, the 10×10 produces the optimal difference in the CO distance of 23140, 16740, and 820 respectively. Finally, the 15×15 produces the optimal difference in the CO distance of 53368, 39928, and 1288 respectively.

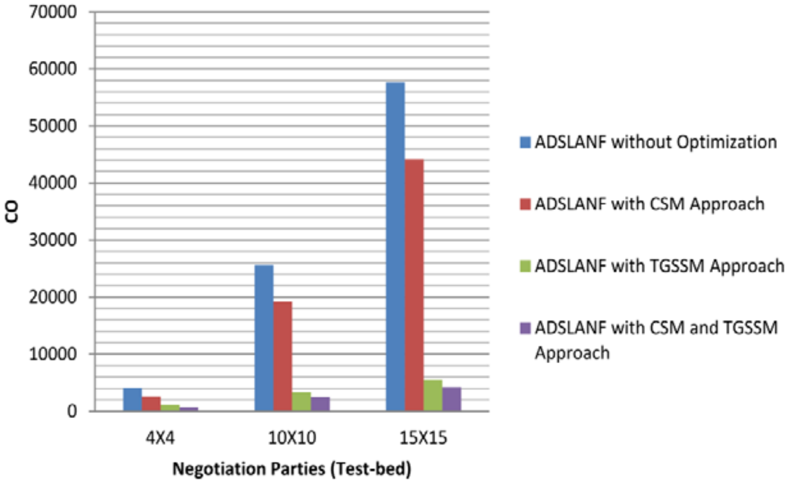


Fig. 3. Optimality achieved in ADSLANF with respect to CO

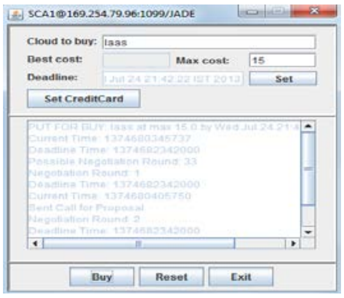
The results of the 4×4, 10×10 and 15×15 test-beds for the NFA, CNM and ADSLANF are simulated for the 32 negotiation rounds with X=4, Y=4, $\lambda(SCAs, ITBAs) = 5$, $\lambda(ITBAs, SPAs) = 5$, $\Phi(SCAs) = 10$, $\Phi(ITBAs) = 20$, $\Phi(SPAs) = 10$, $\Phi(ACs) = 5$ and $\Psi(ITBAs) = 10$, and the observations are shown in **Table 6**. Since, the multiple one-to-many negotiation in the ITB gives the same TNT for all the $ITBA_X$, it is enough to take the single value of X for computation. Similarly, the experiment is simulated for the 10×10 and 15×15 NPs, and the corresponding observations are shown in **Table 7**. The SCAs and SPAs used in the cloud commerce test-bed generate the negotiation request and response, as shown in **Fig. 4** and **Fig. 5** respectively.

Table 6. In-direct negotiation framework results for 4×4 negotiation parties

Negotiation Framework	**Round**	**λ(SCAs, ITBAs) (sec)**	**λ(ITBAs, SPAs) (sec)**	**Φ(SCA) (sec)**	**Φ(ACs) (sec)**	**Φ(ITBAs) (sec)**	**Φ(SPA) (sec)**	**Ψ(ITBAs) (sec)**	**TNT (sec)**	**CO (SCAs, IBAs)**	**CO (IBAs and SPAs)**
Existing NFA	Trial	5	5	10	-	20	10	10	25680	8	128
	Actual				-	600	300	300			3840
	Final				-	20	10	10			128
Existing CNM	Trial	5	5	10	5	20	10	10	7060	8	128
	Actual				150	20	10	10			3840
	Final				5	20	10	10			128
Proposed ADSLANF	Trial	5	5	10	-	20	10	10	1620	8	80
	Actual				-	20	10	10			600
	Final				-	20	10	10			20

Table 7. Results of TNT and CO for 4×4, 10×10 and 15×15 negotiating parties

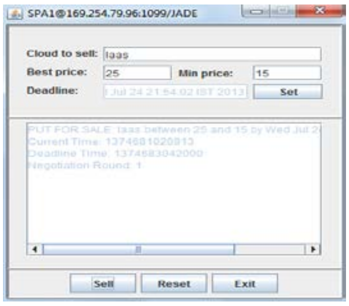
Negotiation Framework	**TNT (Sec)**			**CO**		
	4×4 **NPs**	10×10 **NPs**	15×15 **NPs**	4×4 **NPs**	10×10 **NPs**	15×15 **NPs**
Existing NFA	25680	64200	96300	4104	25620	57630
Existing CNM	7060	7060	7060	4104	25620	57630
Proposed ADSLANF	1620	1620	1620	708	2480	4262



Cloud to buy: Iaas
Best cost:
Max cost: 15
Deadline: Jul 24 21:42:22 IST 2013 Set

```
PUT FOR BUY: Iaas at max 15.0 by Wed Jul 24 21:4
Current Time: 1374680345737
Deadline Time: 1374682342000
Possible Negotiation Round: 33
Negotiation Round: 1
Deadline Time: 1374682342000
Current Time: 1374680405750
Sent Call for Proposal
Negotiation Round: 2
Deadline Time: 1374682342000
```

Fig. 4. Negotiation process in SCAs



Cloud to sell: Iaas
Best price: 25
Min price: 15
Deadline: l Jul 24 21:54:02 IST 2013

```
PUT FOR SALE: Iaas between 25 and 15 by Wed Jul 2
Current Time: 1374681020813
Deadline Time: 1374683042000
Negotiation Round: 1
```

Fig. 5. Negotiation process in SPAs

5.4 Experimental Discussion

The performance of the proposed ADSLANF with the existing NFA and CNM, is compared for the 4×4, 10×10 and 15×15 negotiating parties, with respect to TNT and CO. In **Fig. 6**, the performance of the ADSLANF is better than that of the existing NFA, with respect to TNT, due to the multi-point characteristic feature used in the proposed framework. Next, the ADSLANF outperforms the CNM, since the proposed one follows the multi-point characteristics using the multiple AAC, whereas the existing one follows the multi-point characteristics using the single AC.

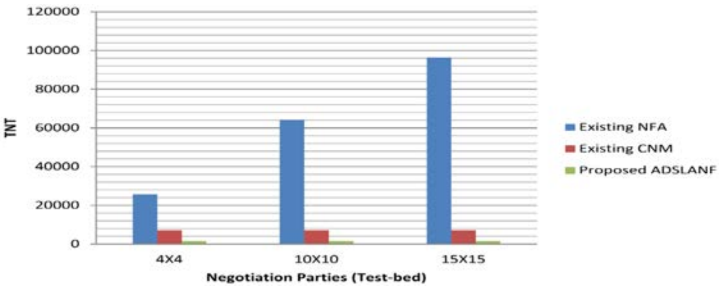


Fig. 6. Comparison of frameworks with respect to TNT

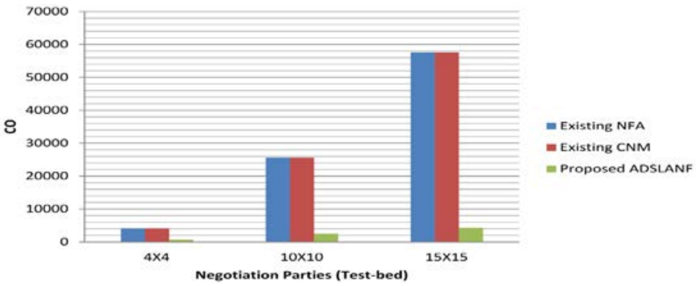


Fig. 7. Comparison of Frameworks with respect to CO

While comparing the NFA and CNM, the proposed ADSLANF outperforms both the existing frameworks with respect to the CO, as shown in **Fig. 7**. Here, the drastic improvement occurs, due to the optimal negotiation strategy obtained, using the CSM and TNGGSSM algorithms in the ITB of proposed framework. First, in the 4×4 test-bed, the proposed ADSLANF improved the performance of the TNT and CO over the existing NFA, in the ratio (round off to the nearest integer value) of 1:16 and 1:6, and the CNM in the ratio of 1:4 and 1:6

respectively. Next, in the 10×10 test-bed, the proposed ADSLANF improved the performance of the TNT and CO over the existing NFA, in the ratio of 1:40 and 1:10, and the CNM in the ratio of 1:4 and 1:10 respectively. Finally, in the and 15×15 test-bed, the proposed ADSLANF improved the performance of the TNT and CO over the existing NFA, in the ratio of 1:59 and 1:14, and the CNM in the ratio of 1:4 and 1:14 respectively. The performance ratio obtained for the improvement, represents the decrease in the TNT and CO among NPs.

The major advantages of the proposed ADSLANF are: that it is (1) flexible to integrate with the cloud management due to the compatibility in the integration of the web service used in Eucalyptus with the agent based service in a JADE environment, 2) adaptable to increase in the AAC due to the elastic nature of the cloud, which can either increase or decrease the number of agents, without any restriction to resource capacity, and (3) it works well in the public cloud environment. Since this research test-bed is made with the private cloud, the elasticity (increasing AACs in the trial negotiation round) is restricted to the bounded capacity of the private cloud. This is the limitation of the proposed framework in areas like the private cloud and grid environment. However, the concept of cloud bursting will overcome the limitation in the private cloud, by dynamically connecting to the public cloud for increasing the resource capacity. The cloud bursting is possible in this research test-bed, due to the compatibility of the Eucalyptus tool with the Amazon EC2 provider.

5. Conclusion and Future Work

The novelty of this research is to introduce the optimal ITB negotiation strategy, in the proposed ADSLANF negotiation framework of the SLAOCMS. The contribution of this research includes the incorporation of the AAC in the ITB, with an optimized negotiation strategy using the CSM and TNGGSSM approaches. These optimizations present the advantages of the significant difference in the TNT and CO, which is demonstrated and validated for the application of the cloud commerce, using the JADE and Eucalyptus test-bed. The evaluated result of this simulation is compared with that of the existing NFA and CNM frameworks, which shows a significant improvement in the proposed ADSLANF, by reducing the TNT and communication overhead among the NPs. In future, researchers can aim to address the behavior learning issues, in the dimensions of uncertainty and risk. Researchers can improve the negotiation strategy in future, by applying the fuzzy preference relation in the negotiable attribute, and using deterministic and stochastic modeling. The trust and risk management of the ITB is left for future work, and it remains an open research problem of the broker based negotiation model.

References

[1] V. C. Emeakaroha, I. Brandic, M. Maurer, S. Dustdar, "Low Level Metrics to High Level SLAs - LoM2HiS Framework: Bridging the Gap Between Monitored Metrics and SLA Parameters in Cloud Environments," in *Proc. of International Conference on High Performance Computing & Simulation*, pp. 48-54, 2010. Article (CrossRef Link)

[2] B. P. Rimal, E. Choi, “A service-oriented taxonomical spectrum, cloudy challenges and opportunities of cloud computing,” *International Journal of Communication Systems*, vol. 25, pp. 796–819, 2012. Article (CrossRef Link)

[3] J. B. Kima, A. Segev, “A Web Services-enabled marketplace architecture for negotiation process management,” *Decision Support Systems*, vol. 40, pp. 71– 87, 2005. Article (CrossRef Link)

[4] R. Rajavel, T. Mala, “SLAOCMS: A Layered Architecture of SLA Oriented Cloud Management System for Achieving Agreement During Resource Failure,” in *Proc. of 2nd International Conference on Soft Computing for Problem Solving, Advances in Intelligent Systems and Computing*, vol. 236, pp. 801-809, 2014. Article (CrossRef Link)

[5] R. Buyya, S. K. Garg, R. N. Calheiros, “SLA-Oriented Resource Provisioning for Cloud Computing: Challenges, Architecture, and Solutions,” in *Proc. of IEEE International Conference on Cloud and Service Computing*, pp. 1-10, 2011. Article (CrossRef Link)

[6] R. Buyya, C. S. Yeo, S. Venugopal, J. Broberg, and I. Brandic, “Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility,” *Future Generation Computer Systems*, vol. 25, no. 6, pp. 599–616, Jun. 2009. Article (CrossRef Link)

[7] S. Venticinque, L. Tasquier, B. Di Martino, "Agents based Cloud Computing Interface for Resource Provisioning and Management," in *Proc. of 6th International Conference on Complex, Intelligent, and Software Intensive Systems*, pp. 249-256, 2012. Article (CrossRef Link)

[8] K. M. Sim, "Towards complex negotiation for cloud economy," in *Proc. of 5th International conference on Advances in Grid and Pervasive Computing*, pp. 395-406, 2010. Article (CrossRef Link)

[9] S. Adabi, A. Movaghar, A. M. Rahmani, H. Beigy, "Market_based grid resource allocation using new negotiation model," *Journal of Network and Computer Applications*, vol. 36, no. 1, pp. 543–565, 2013. Article (CrossRef Link)

[10] I. Brandic, D. Music, and S. Dustdar, "Service mediation and negotiation bootstrapping as first achievements towards self-adaptable grid and cloud services," in *Proc. of 6th international conference industry session on Grids meets autonomic computing*, pp. 1-8, 2009. Article (CrossRef Link)

[11] M. Comuzzi, and B. Pernici, “Negotiation Support for Web Service Selection,” in *Proc. of 5th International Workshop on Technologies for E-Services, Lecture Notes in Computer Science*, vol. 3324, pp. 29-38, 2005. Article (CrossRef Link)

[12] I. Rahwan, "Interest-based Negotiation in Multi-Agent Systems", Thesis, The University of Melbourne, 2004.

[13] X. Bi, J. Xiao, "Classification-based self-adaptive differential evolution and its application in multi-lateral multi-issue negotiation," *Frontiers of Computer Science*, vol. 6, no. 4, pp. 442–461, 2012. Article (CrossRef Link)

[14] M. Wang, H. Wang, D. Vogel, K. Kumar, D. K. W. Chiu, “Agent-based negotiation and decision making for dynamic supply chain formation,” *Engineering Applications of Artificial Intelligence*, vol. 22, no. 7, pp. 1046–1055, 2009. Article (CrossRef Link)

[15] R. Y. K. Lau, “Towards a web services and intelligent agents-based negotiation system for B2B eCommerce,” *Electronic Commerce Research and Applications*, vol. 6, no. 3, pp. 260–273, 2007. Article (CrossRef Link)

[16] A. Cuomo, G. D. Modica, S. Distefano, A. Puliafito, M. Rak, O. Tomarchio, S. Venticinque, U. Villano, “An SLA-based Broker for Cloud Infrastructures,” *Journal of Grid Computing*, vol. 11, no. 1, pp. 1-25, 2013. Article (CrossRef Link)

[17] N. A. Mehdi, A. Mamat, “On the Fly Negotiation for Urgent Service Level Agreement on Intercloud Environment”, Journal of Computer Science, vol. 7, no. 10, pp. 1596-1604, 2011. Article (CrossRef Link)

[18] K. M. Sim, “Complex and Concurrent Negotiations for Multiple Interrelated e-Markets”, IEEE Transactions on Cybernetics, vol. 43, pp. 230-245, 2013. Article (CrossRef Link)

[19] K. M. Sim, “Agent-based Cloud Computing”, IEEE Transactions on Service Computing, vol. 5, pp. 564-577, 2012. Article (CrossRef Link)

[20] J. Richter, M. B. Chhetri, R. Kowalczyk, Q. B. Vo, “Establishing composite SLAs through concurrent QoS negotiation with surplus redistribution,” *Concurrency and Computation: Practice and Experience*, vol. 24, no. 9, pp. 938–955, 2012. Article (CrossRef Link)

[21] A. Rorissa, “Perceived features and similarity of images: An investigation into their relationship and a test of Taversky’s contrast model,” *Thesis, Doctor of Philosophy*, University of North Texas, May, 2005.

[22] A. Tversky, “Preference Belief and Similarity: Selected Writings, A Bradford Book,” *The MIT Press*, Cambridge, Massachusetts, London, England, 2004.

[23] D. Gale, L. S. Shapely, “College Admissions and the Stability of Marriage,” *The American Mathematical Monthly*, vol. 69, no. 1, pp. 9-15, 1962. Article (CrossRef Link)

[24] F. Fowley, C. Pahl, and L. Zhang, "A Comparison Framework and Review of Service Brokerage Solutions for Cloud Architectures," in *Proc. of 1st International Workshop on Workshop on Service-Oriented Computing, Lecture Notes in Computer Science*, vol. 8377, pp. 137-149, 2014. Article (CrossRef Link)

[25] S. Hudert, H. Ludwig, G. Wirtz, "Negotiating SLAs-An Approach for a Generic Negotiation Framework for WS-Agreement," *Journal of Grid Computing*, vol. 7, no. 2, pp. 225–246, 2009. Article (CrossRef Link)

[26] M. Resinas, P. Fernandez, R. Corchuelo, "A bargaining-specific architecture for supporting automated service agreement negotiation systems," *Science of Computer Programming*, vol. 77, no. 1, pp. 4–28, 2012. Article (CrossRef Link)

[27] C. C. Huang, W. Y. Liang, Y. H. Lai, Y. C. Lin, "The agent-based negotiation process for B2C e-commerce," *Expert Systems with Applications*, vol. 37, no. 1, pp. 348–359, 2010. Article (CrossRef Link)

[28] P. Hasselmeyer, H. Mersch, B. Koller, H. N. Quyen, L. Schubert, P. Wieder, "Implementing an SLA Negotiation Framework," in *Proc. of eChallenges Conference*, pp. 154-161, 2007.

[29] F. Toni, J. Bentahar, "Computational logic-based agents," *Autonomous Agents and Multi-Agent Systems*, vol. 16, no. 3, pp. 211-213, 2008. Article (CrossRef Link)

[30] M. Wooldridge and S. Parsons, "On the use of logic in negotiation," in *Proc. of Workshop on Agent Communication Languages*, 2000.

[31] A. Cretan, C. Coutinho, B. Bratu, R. J. Goncalves, "NEGOSEIO: A framework for negotiations toward Sustainable Enterprise Interoperability," *Annual Reviews in Control*, vol. 36, no. 2, pp. 291-299, 2012. Article (CrossRef Link)

[32] T. Groleat, H. Pouyllau, "Distributed inter-domain SLA negotiation using Reinforcement Learning," *Integrated Network Management*, pp. 33-40, 2011. Article (CrossRef Link)

[33] S. Son and K. M. Sim, "A Price- and-Time-Slot-Negotiation Mechanism for Cloud Service Reservations," *IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics*, vol. 42, no. 3, pp. 713-728, Jun. 2012. Article (CrossRef Link)

[34] V. C. Emeakaroha, I. Brandic, M. Maurer, S. Dustdar, “Cloud resource provisioning and SLA enforcement via LoM2HiS framework,” *Concurrency and Computation: Practice and Experience*, vol. 25, no. 10, pp. 1462-1481, 2013. Article (CrossRef Link)

[35] T. Groleat and H. Pouyllau, "Distributed Learning Algorithms for Inter-NSP SLA Negotiation Management," *IEEE Transactions on Network and Service Management*, vol. 9, no. 4, pp. 433-445, 2012. Article (CrossRef Link)

[36] E. Gerding, D. V. Bragt and H. L. Poutre, “Multi-Issue Negotiation Processes by Evolutionary Simulation, Validation and Social Extensions,” *Computational Economics*, vol. 22, no. 1, pp. 39-63, 2003. Article (CrossRef Link)

[37] J. Octavio G. Garcia, K. M. Sim, "Agent-based Cloud service composition," *Applied Intelligence*, vol. 38, no. 3, pp. 436-464, 2013. Article (CrossRef Link)

[38] H. Pouyllau, and G. Carofiglio, "Inter-carrier SLA negotiation using Q-learning," *Telecommunication Systems*, vol. 52, no. 2, pp. 611-622, 2013.

[39] M. Wooldridge, “An Introduction to Multiagent Systems,” John Wiley & Sons Ltd, England, 2002.

[40] D. Nurmi, R. Wolski, C. Grzegorczyk, G. Obertelli, S. Soman, L. Youseff, and D. Zagorodnov, "The Eucalyptus Open-source Cloud-computing System," in *Proc. of 9th IEEE/ACM International Symposium on Cluster Computing and the Grid*, pp. 124-131, 2009. Article (CrossRef Link)

Rajkumar Rajavel is pursuing Ph.D. as Anna Centenary Research Fellow in the Department of Information Science and Technology, Anna University, India. He has obtained his M.Tech Graduate in the specialization of Information Technology during the year 2010. His research areas include Negotiation, Scheduling, Load Balancing and Resource Management in Grid and Cloud Computing. He has presented 5 research publications in the international conferences and 1 research publication in the international journal.

Mala Thangarathinam is an Associate Professor in the Department of Information Science and Technology, Anna University, India. She completed her Ph.D. on NLP at Anna University during the year 2008. Currently, she is guiding more than ten research scholars. Her research areas include Natural Language Processing, Virtualization Technologies, Grid Computing and Cloud Computing. She has more than 4 research publications in national conferences, 17 research publications in international conferences and 7 research publications in international journals.