Association for Information Systems
AIS Electronic Library (AISeL)

ECIS 2010 Proceedings

European Conference on Information Systems (ECIS)

2010

A Multi-Tier Negotiation Protocol for Logistics Service Chains

Paul Karaenke
University of Hohenheim, karaenke@uni-hohenheim.de

Stefan Kirn
University of Hohenheim, kirn@uni-hohenheim.de

Follow this and additional works at: http://aisel.aisnet.org/ecis2010

Recommended Citation

Karaenke, Paul and Kirn, Stefan, "A Multi-Tier Negotiation Protocol for Logistics Service Chains" (2010). *ECIS 2010 Proceedings*. 83.
http://aisel.aisnet.org/ecis2010/83

This material is brought to you by the European Conference on Information Systems (ECIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in ECIS 2010 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

A MULTI-TIER NEGOTIATION PROTOCOL FOR LOGISTICS SERVICE CHAINS

Journal:	*18th European Conference on Information Systems*
Manuscript ID:	ECIS2010-0068.R1
Submission Type:	Research Paper
Keyword:	Intelligent agents, Electronic value chains, Electronic markets, Artificial intelligence

A MULTI-TIER NEGOTIATION PROTOCOL FOR LOGISTICS SERVICE CHAINS

Paul Karaenke, University of Hohenheim, Information Systems 2, Schwerzstr. 35, 70599 Stuttgart, Germany, karaenke@uni-hohenheim.de

Stefan Kirn, University of Hohenheim, Information Systems 2, Schwerzstr. 35, 70599 Stuttgart, Germany, kirn@uni-hohenheim.de

Abstract

Logistics service chains are characterized by multiple service providers contributing to the provision of a composite logistics service to a customer. In particular, various contractual dependencies exist across service chain levels. The object of our research is resource allocation which has to consider these dependencies to avoid overcommitment and overpurchasing. We propose a multi-tier negotiation protocol for solving this problem. The proposed artifact is developed from an interaction protocol engineering perspective in accordance with the design science paradigm. First evaluation experiments show that the protocol prevents overcommitments and overpurchasing, leading to higher expected profits for logistics service providers.

Keywords: logistics, resource allocation, negotiation, multiagent systems

1 INTRODUCTION

Logistics service chains are characterized by multiple service providers contributing to the provision of a composite logistics service to a customer. An explicit formal statement of the obligations and guarantees regarding services in a business relationship is referred to as a service level agreement (SLA) (Verma 1999, p.1-5). Thus, a SLA provides the operational definition of a service as part of a contract between a service provider and a service consumer. In real-world service chains, contractual agreements exist along the flow of goods and services. This is also true for individualized services for which customer-specific agreements have to be negotiated. These agreements depend either directly or indirectly on other agreements along the service chain (e.g., for procurement, outsourcing, etc.).

The *object of our research* is service level agreement-based resource allocation in logistics service chains. We address the *problem* of considering contractual dependencies across service chain levels. If these dependencies are not considered, overcommitment or overpurchasing can happen. We propose a multi-tier negotiation protocol for solving this problem. Therefore, we study SLA negotiation with regards to dependencies between SLAs on different service chain levels and formally analyze these dependencies. The proposed artifact is developed from an *interaction protocol engineering perspective* (Huget and Koning 2003) in accordance with the *design science paradigm* (Hevner, March, Park, and Ram 2004). First evaluation experiments show that the protocol avoids overcommitment and overpurchasing, leading to higher expected profits for logistics service providers.

The *contribution* of this paper is a dependency-aware, multi-tier negotiation protocol for composite service provision over multiple service chain levels. The remainder of this paper is as follows: section 2 investigates the problem based on formal models for service chains and SLAs, and provides a specification of the requirements. In section 3, we propose the negotiation protocol specification. In section 4, we provide an evaluation of this artifact in an application scenario of airport logistics. Section 5 discusses related work. Section 6 summarizes the result and gives an outlook to future work.

2 PROBLEM ANALYSIS

2.1 Formal Model

A service chain can be defined by service flows between actors. Then, a service chain is a directed graph $V = (A, F)$ consisting of the set of actors A and the set of service flows F. An actor is an abstraction that is delimitable from other actors which contribute to the creation of services. This contribution is carried out by production; i.e., the combination of production (input) factors and the transformation of these into services (output factors). F is a relation over the actors so that $F \subseteq A \times A$. A service flow $f \in F$ connects two actors $a_1 \in A$ and $a_2 \in A$ with $f = (a_1, a_2)$. Service flows are directed and primarily carried out from upstream actors down to the customer actor, which does not show primary service flows to other actors itself. Figure 1 shows a generic model for service chain (SC).

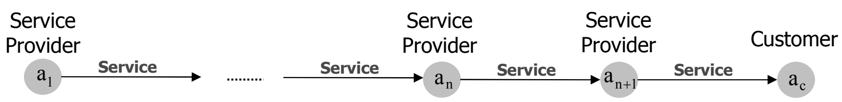

Figure 1. Generic service chain model.

Abstracted from implementation details, a SLA (e.g., WS-Agreement (OGF 2007)) contains information about (1) the involved parties A, (2) the service definitions S, and (3) guarantees G assured in an agreement. According to the service chain perspective, the parties involved in agreements are equivalent to the set of actors A. As the set of SLAs C can also be mapped to the set of value flows F, *C* can be mapped to the according service chains, $\forall C' \subseteq C \exists f : Pot(C') \rightarrow Pot(V)$. A set of observable service properties (SLA parameters) P_S is part of the service definitions *S* ($P_S \subset S$). These properties are mapped to assurances on service quality as part of the set of guarantees *G*. *G* includes qualifying conditions on external factors *COND* (e.g., time of the day), quality characteristics $Q = \{q \mid q : P_S \rightarrow \mathbb{R}\}$, and business values (economic values) *BV*; i.e., $G = COND \times Q \times BV$. The set of SLAs *C* can then be defined as $C = (A \times A, Pot(S), Pot(G))$.

For a given request $r_{a_c, a_n, o} \in R$ from customer a_c to service provider (SP) a_n, the SC constitutes the solution space. The concrete actors for service delivery have to be determined and contracts (SLAs) between all respective actors have to be negotiated in a coordinated manner. Service parameters and their values are subject to changes along the SC (*SLA parameter aggregation*), since the service provided to the customer is a *composite service*. For SLA parameter aggregation, Jaeger, Rojec-Goldmann, and Mühl (2004) have identified seven relevant *abstract composition patterns* (CPs). For each combination of CP and SLA parameter type (quality of service dimension) one aggregation definition has to exist. For numerical SLA parameters, aggregation functions can be defined. Non-numerical SLA parameters can not be mathematically aggregated. Other aggregation definitions (e.g., rule-based) might be required (Jaeger et al. 2004). We limit the scope to parameters which can be mathematically aggregated. The set of aggregation functions can be defined as $X = \{x_{cp,p} \mid cp \in CP \wedge p \in P_S, x_{cp,p} : \mathbb{R}^n \rightarrow \mathbb{R}\}$ where CP is the set of composition patterns. Therewith, the set of customer requests can then be defined as $R := \overline{C} \times Pot(X)$ where $\overline{C} = (A \times A, Pot(S), Pot(G))$ is the set of SLA templates sent in requests; i.e., the expression of the customers' preferences for the SLAs to be established.

2.2 Requirements Analysis

Negotiation between actors takes place by a negotiation protocol to which the actors commit themselves. Therefore, considering contractual dependencies across service chain levels can be addressed by extending current negotiation protocols. The protocol's overall goal is to support contracting in logistics service chains. Figure 2 shows a UML2 use case diagram for the application of the protocol in logistics contracting.

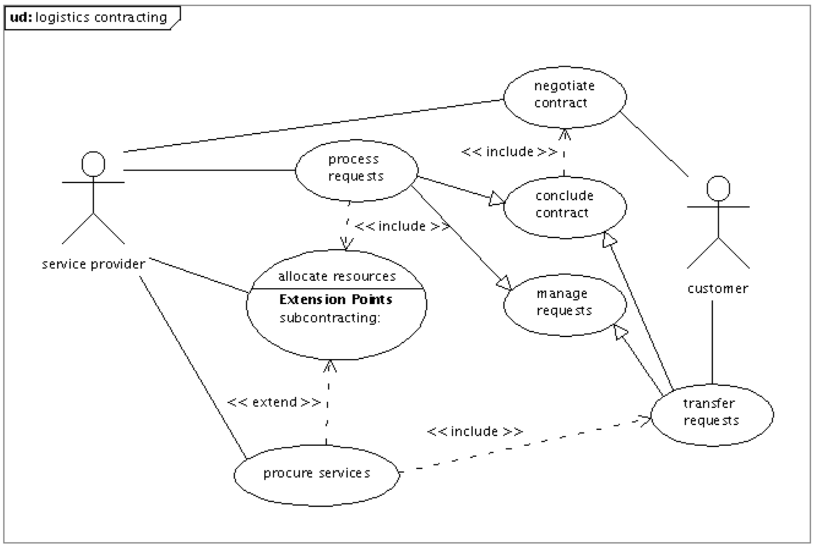

Figure 2. Logistics contracting use case diagram.

The problem in the production of the individualized service consists of both the non-determinateness of the individual customer requirements until the point in time of the demand, as well as the individuality of the requirements themselves. For the service creation it is possible to (1) utilize resources from the own inventory and (2) buy services from a third party (subcontracting). The latter is done if the own capacity is not sufficient or if the utilization of the own capacity is economically not favorable because of the cost function (e.g., step costs). Here, the provider has to consider economically relevant values to determine the concrete actors for the service creation. For individualized services, this implies that the individual requirements of the customer determine the requirements to SLAs that have to be established on upstream service chain levels. Due to the non-determinateness of the requirements on needed resources and SLAs to be established on upstream service chain levels until the moment of service demand, the value flow has to be adapted. The capability of adaptation (i.e., the adaptation potential) is denoted as adaptivity. For software support of service chain adaptivity, the structure of the service chain has to be mapped to the level of information systems. This is done by technical SLAs specifications, which originate from Web service technology and describe the contractual agreements of actors involved in service creation.

We develop the protocol in accordance with the interaction protocol engineering approach from the multiagent technology area. In the analysis phase, we follow a goal-oriented requirements engineering approach towards a formal definition of design goals. On the basis of the use-case and service chain model, a structured informal document with all features of the protocol can be provided (Huget et al. 2003, p.2-4). It is shown in Table 1.

Name	Multi-tier negotiation protocol for logistics service chains
Keywords	nested negotiation, multi-tier negotiation
Agents' role	customer, service provider (SP)
Initiator	customer
Prerequisite	the agents must know how to send messages to other agents
Function	A customer requests to buy a logistics service which has to be provided at a certain service level (quality of service). SPs offer bids on the request. The requested service may be realized as a composite service. Therefore, aggregation definitions on the service parameters have to be provided. The SPs may procure (sub-)services from other SPs (subcontracting, outsourcing) in chained order processing if the SPs' capacities are insufficient or if the utilization of the SPs' capacities is not economically favorable. The protocol enables to negotiate all contracts along the service chain in a coordinated manner.
Behavior	The protocol can be decomposed into three phases: 1. In the collect proposal phase, the customer's request is transferred to potential SPs on service chain tier *s*. The SPs process the request and may decide to outsource the request to tier *s+1* of the service chain; i.e., they transfer own requests to other SPs on tier *s+1*. SPs' subcontracting activities can be recursively extended to an arbitrary number of tiers. Before a SP on tier *s* answers the final customer's request with a binding offer, it evaluates the offers received from SPs on tier *s+1*. Thus, the SP can make an offer to the final customer based on the results of all nested negotiation offers. 2. In the acceptance notification phase, the SPs are informed about the acceptance of the offers they have provided. The SPs on tier *s* will subsequently inform the SPs on tier *s+1* about the acceptance of dependent offers. The recursion is executed to the number of tiers on which offers were made. 3. The execution phase comprises the actual service provision. The result of the execution (e.g., service level provided) is reported to the customer. SPs on tier *s* report the state of the provision of the services to customers on tier *s-1*. The recursion is executed to the number of tiers on which agreements have been established.
Constraints	All agents must be authenticated.
Termination	(1) All SPs have provided the services as contracted, (2) the provision of one or more services has failed, or (3) no agreement has been established.

Table 1. Informal description of the protocol.

2.3 Design Goal Specification

The consideration of contractual dependencies, in terms of service parameter dependencies over multiple service chain levels, requires a multi-tier negotiation protocol that is capable of making the parameter aggregation definitions explicit. Further, the protocol must allow for subcontracting activities by service providers (SPs) and for negotiating all (sub-)contracts for a concrete customer request in a coordinated manner. Table 2 shows the definitions of identified design goals of the protocol in terms of goal-oriented requirements engineering.

Goal	Informal definition	Formal definition
g_1	The protocol must support negotiation of contracts (SLAs) between customers and SPs on the basis of the orders placed by customers.	$\exists r_{a_c,a_n,o} \in R \mid \exists f_N : C \times R \rightarrow C' \mid$ $\left[(a_c,a_n) \in c_o \in C'\right] \wedge \left[c_o \notin C\right]$
g_2	The protocol must support the explicit definition of parameter aggregation functions in order transfers to enable the provision of composite services.	$\left[\forall \bar{c} \in \overline{C}, \bar{c} \in r_{a_c,a_n,o} \in R\right] \wedge \left[\forall cp \in CP\right]$ $\exists x_{cp,p} \in X \mid p \in \bar{c}$
g_3	The protocol must support SPs in service procurement from other SPs (subcontracting, outsourcing) to enable chained order processing if the SPs' capacities are insufficient or if the utilization of the SPs' capacities is not economically favorable.	$\forall r_{a_c,a_n,o} \in R \exists f_R : R \rightarrow R \mid$ $\left[f_R(r_{a_c,a_n,o}) = r_{a_n,a_m,l}\right] \wedge$ $\left[\exists p \in \overline{c_o} \in r_{a_c,a_n,o} \mid p \in \overline{c_l} \in r_{a_n,a_m,l}\right]$
g_4	The protocol must prevent contracts to be established between customer and SP if no agreement is reached during the negotiation of subcontracts between SPs (overcommitment).	$\forall r_{a_c,a_n,o}, r_{a_n,a_m,l} \in R, f_R(r_{a_c,a_n,o}) = r_{a_n,a_m,l} \mid$ $f_N(c_l, r_{a_n,a_m,l}) \notin C \Rightarrow f_N(c_o, r_{a_c,a_n,o}) \notin C$
g_5	The protocol must prevent subcontracts to be established between SPs if no agreement is reached during the negotiation between customer and SP (overpurchasing).	$\forall r_{a_c,a_n,o}, r_{a_n,a_m,l} \in R, f_R(r_{a_c,a_n,o}) = r_{a_n,a_m,l} \mid$ $f_N(c_o, r_{a_c,a_n,o}) \notin C \Rightarrow f_N(c_l, r_{a_n,a_m,l}) \notin C$

Table 2. Design goal definitions.

3 COMBINATORIAL CONTRACT NET PROTOCOL

In this section, we propose a dependency-aware, multi-tier negotiation protocol for composite service provision over multiple service chain levels. The basis for the proposal is the FIPA Contract Net Interaction Protocol (CNP) (FIPA 2002a), the FIPA interpretation of the original contract net protocol proposed by Smith (1980). The protocol should allow subcontracting activities by the participants; i.e., a participant can evaluate if subcontracting is possible and feasible in advance to making binding proposals. We denote the proposed interaction protocol as Combinatorial Contract Net Protocol (CCNP), since the protocol enables the *combination of tiers* for coordinating interactions on different service chain levels. That means that the protocol allows considering dependencies over multiple tiers for subcontracting.

When adopting the existing CNP, its current set of communicative acts (performatives) (FIPA 2002b) has to be extended as follows:

- The *cfp* (call for proposals) message has to include explicit service parameter aggregation definitions (cf. g_2); i.e., the execution of the task (1) explicitly requires multiple services or (2) can be realized with a composite service, potentially composed at run time.
- The *inform-result* message type in the CNP does not relate to separate FIPA communicative acts but to the general inform act. Once an agent has completed one or more tasks, it sends a message to the initiator in the form of an *inform-done*; more information about the execution can be provided

by an *inform-result*. In case of an *inform-result* message, the agent has to aggregate the results of the single services which have been utilized to execute the task. This can be realized using the parameter aggregation definitions in the *cfp* message.

3.1 Protocol Specification

The sequence diagram of the protocol's interactions is shown in Figure 3a. It consists of three encapsulated interaction sequences. The CCNP-collect-proposals interaction sequence includes the interactions for collecting the proposals (offers) by the initiator (customer) from the participants, the FIPA notation for the agents executing the tasks (SPs). Collecting the proposals can also include subcontracting interactions. If and only if the participant has made a proposal, he is notified in the CCNP-acceptance-notification interaction sequence about the allocation result. If and only if the participant's proposal is accepted, the CCNP-execution interaction sequence is executed. This sequence includes the provision of the success state for all allocated actions by the participants which can also involve multiple tiers as for the allocation. Details on the referred interactions are outlined in the following.

The CCNP-collect-proposals interaction sequence is shown in Figure 3b. The initiator on tier s sends a *cfp* message to the participants on the same tier. Optionally, if (i) participants exist on tier $s+1$ which are potentially capable of execution actions or subactions described in the *cfp* on tier s and (ii) (one or more of) the tier s participants prefer to subcontract actions or subactions on tier $s+1$ (acting as tier $s+1$ initiators), the CCNP-collect-proposals interactions are recursively executed on tier $s+1$. The recursion can be extended to an arbitrary number of tiers (up to tier $s+x$).

Similarly to the CNP (FIPA 2002a), participants receiving the *cfp* generate n responses. The tier s participants may decide that they refuse to propose, resulting in $i=n-j$ *refuse* act responses. Alternatively, j participants propose to perform the task, specified as *propose* acts.

The initiator evaluates the j proposals received and selects participants to perform the tasks. The CCNP-acceptance-notification interaction sequence, shown in Figure 3c, covers informing the participants of the allocation result. The l participants of the selected proposal(s) will be notified with an *accept-proposal* message. The remaining k participants will receive a *reject-proposal* message. The recursion is executed to the number of tiers on which proposals were made (propose messages were sent).

The CCNP-execution interaction sequence is shown in Figure 3d. Once the sub-contracted participants (if any) on tier $s+1$ have completed the tasks, they send completion messages to the tier $s+1$ initiator in the form of an *inform-done* or a more explanatory version in the form of an *inform-result*. If a participant fails to complete one or more tasks, a failure message is sent. Similarly, the tier s participants report the state of the execution of tasks to the tier s initiator. (FIPA 2002a)

3.2 Protocol Implementation

The CCNP has been implemented as a reusable capability (plan library) for the Jadex BDI agent framework (Jadex 2009). The existing CNP protocol capability implementation (Pokahr and Braubach 2007, p.117f) has been extended to support nested negotiations. The goals *ccnp_collect_proposals* and *ccnp_acceptance_notification* have been added to be able to dispatch the respective activities and process their results from the outer negotiations.

The implementation can thus be used for resource allocation in multiagent systems (MAS), in which the service chain actors are represented by software agents. The belief-desire-intention (BDI) agent paradigm is a model for describing rational software agents – agents that reason, based on beliefs, which action to perform to reach given goals. That is, the BDI paradigm facilitates goal-driven system behavior. The underlying software architecture of our system has been presented in previous work (Karaenke, Micsik, and Kirn 2009).

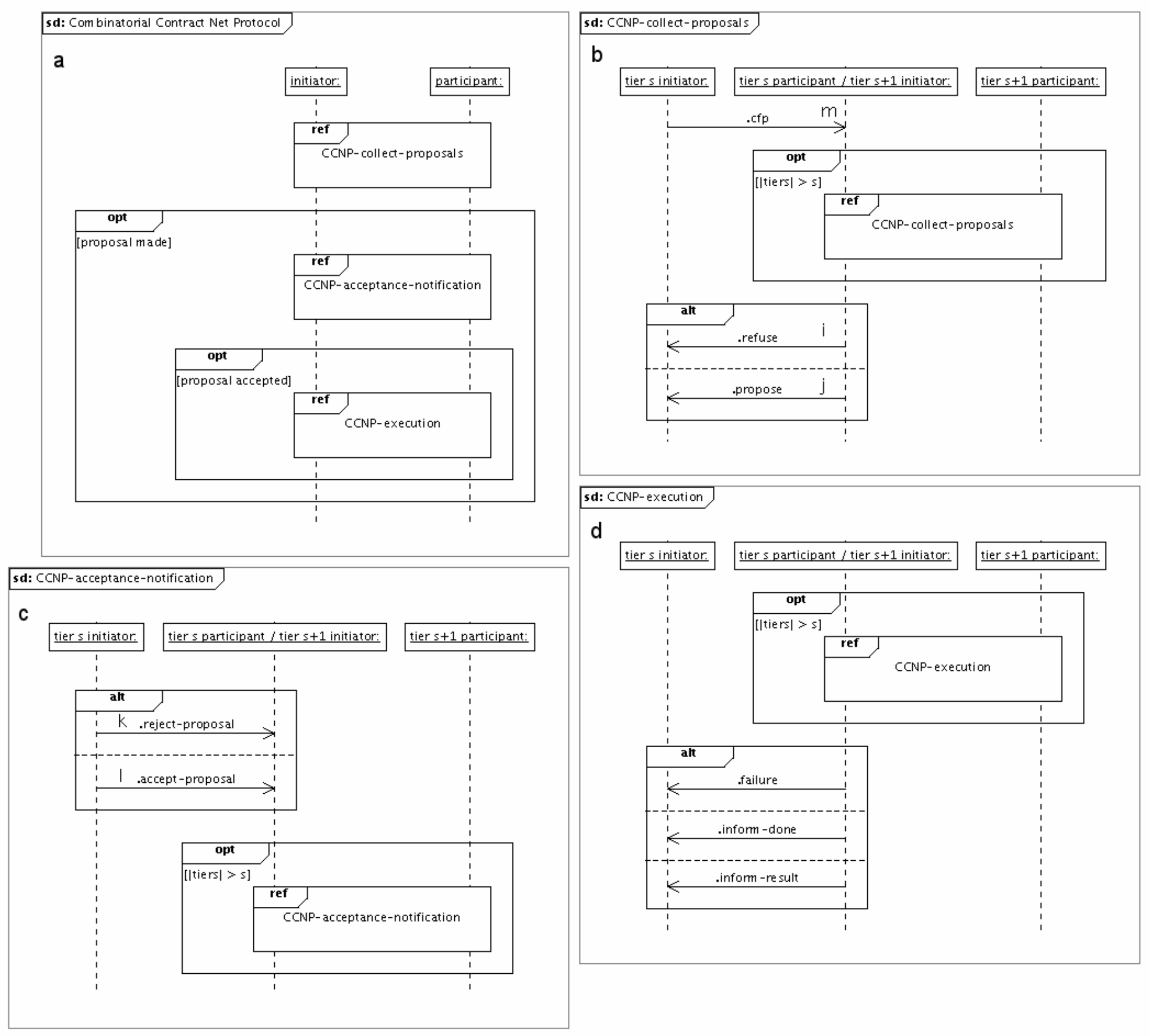

Figure 3. Combinatorial contract net protocol sequence diagrams.

4 EVALUATION

In this section, we present a simulation-based validation of the proposed artifact. Evaluation by means of simulation is an experimental evaluation method. The artifact is executed with artificial data (Hevner et al 2004, p.12). We consider the airport logistics domain: the airport management service chain produces *ground handling services* for the dispatching of aircrafts at airports' aprons. The scenario and its underlying business objectives, service types and dependencies is part of the European IST project BREIN (Jones 2008).

4.1 Scenario

The logistics services of ground handling at airports are represented in an agent-based system. The ground handling logistics service chain includes airlines as customers as well as actors from luggage, passenger and aircraft related service providers. Contracts between such actors are described in SLAs. Our investigations are exemplarily limited to passenger transports on the ground from and to aircrafts (bus transports). The individual requirements of the customer (airline) include the number of passengers, the parking position of the aircraft, planned and actual arrival and departure time, etc. The temporal dependencies of aircraft dispatching services are mapped to dependencies between SLAs. These are considered during determination and contracting of the concrete actors for service creation. Figure 4 shows the actors involved in the service creation process.

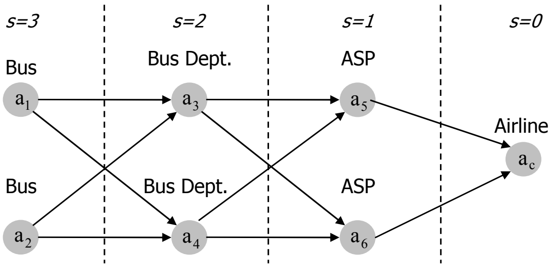

Figure 4. Ground handling service chain model

Airlines (tier 0) determine the individual requirements for the dispatching of aircrafts. The airline service provider (ASP) actors on tier 1 coordinate the service "passenger transportation on the ground" within the airline service provider (ASP) organization. The actors on tier 2 represent the bus departments, while the busses which provide "bus trip" services act on tier 3. All actors can buy the services from two SPs on the next tier to enable competition. Parts of the service definitions (s_{PT}, p_{CT}, p_{PC}) are identical for all tiers as only inbound flights (arriving aircrafts) are considered. s_{PT} denotes the general definition of the passenger transportation service. p_{CT} denotes the cycle time service property, and p_{PC} the passenger capacity. The aggregation functions for the properties for sequential as well as parallel execution (AND-split followed by an AND-join) are shown in Table 3.

$$x_{SEQUENCE,CT}(x_1,\dots,x_n) = \sum_{i=1}^{n} x_i$$
$$x_{SEQUENCE,PC}(x_1,\dots,x_n) = \sum_{i=1}^{n} x_i$$
$$x_{AND-AND,CT}(x_1,\dots,x_n) = \max(x_1,\dots,x_n)$$
$$x_{AND-AND,PC}(x_1,\dots,x_n) = \sum_{i=1}^{n} x_i$$

Table 3. Service parameter aggregation functions.

4.2 Experiment

For the experiment, two sets of customer requests $R_{a_c,a_n,o}$ have been generated using pseudorandom numbers and the property constraints $p_{CT} \in [8,18] \subset \mathbb{N}$ and $p_{PC} \in [1,200] \subset \mathbb{N}$. The price function is identical for all SPs, $p_{BV}(p_{PC}, p_{CT}) = \left(\omega_1 + \omega_2 \cdot p_{PC} + \omega_3 \cdot p_{CT}^{-1} + \omega_4 \cdot \varepsilon_1\right) \cdot \left(1 + 0.1 \cdot \varepsilon_2\right)^{3-s}$, where $\varepsilon_1, \varepsilon_2 \in [0,1] \subset \mathbb{R}$ are uniformly distributed random numbers. The price function contains a fixed costs part ω_1. As costs are assumed to increase with the number of passengers, the function includes a part proportional to p_{PC} with proportionality factor ω_2. Further, the function contains a part inversely proportional to p_{CT} with proportionality factor ω_3, as a lower cycle time is assumed to imply higher costs. In addition, the function contains a random part proportional to ε_1 with proportionality factor ω_4 to represent additional internal cost factors of the SP. For the experiment, we have used the weights $\omega_1 = 15, \omega_2 = 0.01, \omega_3 = 25, \omega_4 = 2$ which lead to constraints for the prices $16.66 < p_{BV}(p_{PC}, p_{CT}) \le 24$. In case of nested negotiations, the SPs know the prices on tier *s+1* in advance. Thus, they calculate the prices based on the proposal from the SPs on tier *s+1*; i.e., $p_{BV,s+1}(p_{BV,s}) = p_{BV,s} \cdot (1 + 0.1 \cdot \varepsilon_2)$.

The SPs on all tiers are assumed to refuse to propose with probability φ. Reasons for refusals in real-world scenarios can include insufficient resources, incompatible service properties, etc. However, details about these reasons are beyond the scope of this research. If a SP fails to provide a service with the properties contracted, it will have to pay a contractual penalty of $pen(p_{BV}) = 0.1 \cdot p_{BV}(p_{PC}, p_{CT})$. The SPs on tier 3 (busses) are assumed to provide any service for which a SLA has been established. From all received bids meeting the service property constraints p_{CT} and p_{PC}, the bid with the lowest price is selected. The experiment has been executed with both the conventional, non-nested CNP and the nested CCNP for the same sets of customer requests and for two different probabilities φ each.

Table 4 shows the simulation results. The subsets $C_{n,m} \subseteq C$ contain the SLAs established between tier n and m. *rev(a_i)* dentotes the revenues, *cost(a_i)* the costs, and *prof(a_i)* the profit of actor a_i. For i=[3,6] this comprises intermediate actors only.

$\lvert R_{a_c,a_n,o}\rvert$	proto-col	φ	pro-pos-als	ref-us-als	$\lvert C_{0,1}\rvert$	$\lvert C_{1,2}\rvert$	$\lvert C_{2,3}\rvert$	$\lvert C\rvert$	$\sum_{i=3}^{6} rev(a_i)$	$\sum_{i=3}^{6} cost(a_i)$	$\sum_{i=3}^{6} prof(a_i)$	$\sum_{i=3}^{6} pen(a_i)$
20	CCNP	0.25	141	49	17	17	17	51	663.88	637.12	26.76	0.00
20	CNP	0.25	77	29	17	16	14	47	602.04	576.89	25.15	5.64
20	CCNP	0.5	52	58	9	9	9	27	350.96	338.27	12.69	0.00
20	CNP	0.5	31	45	11	7	5	23	228.44	234.27	-5.83	12.19
40	CCNP	0.25	303	91	39	39	39	117	1517.60	1467.96	49.64	0.00
40	CNP	0.25	166	56	36	35	32	103	1316.59	1272.39	44.20	7.90
40	CCNP	0.5	100	120	19	19	19	57	778.73	739.12	39.61	0.00
40	CNP	0.5	87	95	32	19	12	63	635.98	638.10	-2.12	40.09

Table 4. Experiment results.

Figure 5 shows the cummulated profit of the intermediate SPs, $\sum_{i=3}^{6} prof(a_i)$. One can see that the non-nested CNP negotiations for *φ=0.5* lead to penalty payments and that profit turns into loss. For *φ=0.25*, the CNP may temporarily lead to higher profit than the CCNP due to the non-determinateness of the prices, though the CCNP negotiations lead to higher profits of the intermediate SPs for an increased number of customer requests.

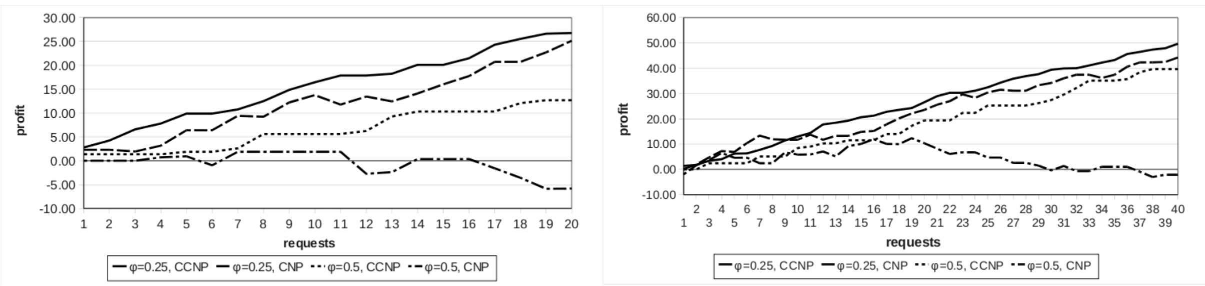

Figure 5: Profit of intermediate service providers during the simulation.

4.3 Discussion

Goal g_1 is obviously fulfilled, since $\lvert C\rvert > 0$. g_2 is addressed by the data format for *cfp* messages in the CCNP implementation. g_3 is addressed by the fact that the SPs can either (i) forward the *cfp* message received or (ii) disaggregate the *cfp* according to the given aggregation functions. However, these requirements are also fulfilled by the CNP implementation without nested negotiations. For the nested negotiations, goals g_4 and g_5 are essential. It can be seen from Table 4 that the CCNP leads to the same number of established contracts on all tiers; i.e., $|C_{0,1}|=|C_{1,2}|=|C_{2,3}|$. That is, $f_N(c_l, r_{a_n,a_m,l}) \notin C \Rightarrow f_N(c_o, r_{a_c,a_n,o}) \notin C$ and $f_N(c_o, r_{a_c,a_n,o}) \notin C \Rightarrow f_N(c_l, r_{a_n,a_m,l}) \notin C$ hold $\forall r_{a_c,a_n,o}, r_{a_n,a_m,l} \in R, f_R(r_{a_c,a_n,o}) = r_{a_n,a_m,l}$ in the experiment. In contrast, the non-nested CNP negotiations lead to inequalities between the number of contracts on the different service chain tiers; i.e., the CNP negotiations do not fulfill goals g_4 and g_5. This leads to contractual penalty payments which are the reason for negative profits (losses). The probability to establish non-accomplishable SLAs (overcommitment) increases with φ. Further, the subsequent negotiations in the CNP negotiations lead to a lower probability to establish contracts, as the solution space is limited to one actor for each tier before the negotiations on the next tier start. The nested negotiations avoid this limitation by deferring the proposals until proposals of the next tier have been collected, which explains the significantly higher number of proposals in the CCNP negotiations.

5 RELATED WORK

The CCNP interactions constitute reverse (procurement) auctions. Approaches for resource allocation with *combinatorial auctions* (e.g., Walsh, Wellman, and Ygge 2000) consider *centralized* winner determination along all tiers of the service chain. Thus, a central entity that is collecting all offers and demands on all tiers of the service chain at a single point is required. The assumption that such a central entity exists contradicts the distributed nature of service chains. In addition, it is not possible to consider the dependencies between contracts along the service chain if offer and demand bids can not be combined. Then, the fulfillment of contracts may be unaccomplishable due to (1) missing contracts to other actors which are required for the fulfillment (overcommitment) or (2) missing contracts to customers (overpurchasing).

Zhang, Lesser, and Abdallah (2005) investigate the problem of *multi-linked negotiations*, interconnected negotiations which influence each other. The relationships of related negotiations are classified into two categories. Two negotiations are *directly linked* if the failure of one negotiation implied the infeasibility or unnecessity of the second. *Indirectly linked* negotiations compete for use of common resources. The approach proposes a temporal ordering of multi-linked negotiations to be carried out either sequentially or in parallel. However, it does not avoid the establishment of unaccomplishable contracts and therewith penalty payments, as negotiations are considered as atomic blocks and thus the interleaving of directly linked negotiations is not considered. Si, Edmond, ter Hofstede, Dumas, and Chong (2005) propose an approach for composing interrelated negotiations allowing compositions of alternative (one-or-the-other) and complementary (all-or-nothing) trading activities. Anthony and Jennings (2003) investigate the problem of bidding across multiple auctions to procure the best deal for the desired good. The coordination of procurement activities in multiple bilateral negotiations is investigated by Nguyen and Jennings (2004). Schillo, Kray, and Fischer (2002) analyze resource allocation with the CNP and propose three strategies for the eager bidder problem, which results from indirectly linked negotiations. Linking supply and demand side in interleaved negotiations is out of the scope of all mentioned approaches. Preist, Bartolini, and Byde (2003) present an algorithm for decision making agents which buy component services on one market and sell composite services on another market in auctions. The probabilistic approach evaluates the expected profit or loss of participating in any set of auctions. The problem of potential overcommitment is considered and the probability for it is minimized, though it is not avoided.

The concept of *nested negotiations* is applied to collaborative problem solving in (Kirn and Schlageter 1992). Karageorgos, Mehandjiev, Hämmerle, and Weichhart (2003) propose a protocol for nested negotiations with conceptual similarities to the CCNP. It is applied to support the integration of manufacturing and logistics service planning. Negotiations are considered for exactly three tiers for a concrete use case; interactions with the customer are out of scope of the approach. A pre-defined set of physical products (i.e., no services) for which the combination definitions are known to all relevant actors is considered. Thus, details about the disaggregation of requests, respectively aggregation of results, are not given.

The discussion shows that a research gap exists in the area of nested negotiation protocols for service chains that avoid both overpurchasing and overcommitment. In our approach, we propose a multi-tier negotiation protocol for solving this problem, which also addresses the problem of service composition.

6 CONCLUSION

The *contribution* of this paper is a dependency-aware, multi-tier negotiation protocol for composite service provision over multiple service chain levels. The research has been conducted in accordance with the design science paradigm. For the construction of the artifact, the interaction protocol engineering approach has been applied. The evaluation of the artifact has been conducted with multi-agent simulation. Formal models have been utilized for problem analysis and requirement specification while honoring the logistics basic functions relevant for inter-organizational resource allocation. The specification of the protocol has been provided in UML2 sequence diagrams. The implementation in the Jadex BDI framework and therewith the adoption of the BDI multi-agent paradigm fosters the application of the protocol in business goal-driven systems.

The experiment has been limited due to a set of assumptions: Only one type of service is considered. The scenario provides only the minimal number of actors to enable competition on three service chain tiers. In addition, the SPs on tier 3 are assumed to fulfill all established contracts at any time which is not realistic in real-world scenarios. Further, the experiment has only been rigorously executed for three tiers, though the protocol is applicable to an arbitrary number of tiers. Finally, although leading to superior results regarding the SPs' profits, the CCNP significantly increases the number of messages sent during negotiations, as the solution space is not limited to certain actors until all proposals have been collected on the next tier. In addition, only the problem of directly linked negotiations, in which the failure of one negotiation implies the infeasibility or unnecessity of the second, is addressed. Indirectly negotiations which compete for using shared resources are not considered.

We have shown that the protocol provides means for multi-tier resource allocation without centralized control. It allows the required ad hoc contracting of the required services, honoring the dependencies of SLAs over multiple service chain levels. Future research has to further underpin the utility of the artifact in simulations in advanced scenarios that relax the assumptions made.

Acknowledgements

The work presented in this paper was partly funded by the European Commission under the project BREIN (FP6-034556) and by the German Federal Ministry of Education and Research under the project InterLogGrid (BMBF 01IG0910E).

References

Anthony P.; Jennings, N. R. (2003). Developing a bidding agent for multiple heterogeneous auctions. ACM Transactions on Internet Technology, 2(3), 185–217.

FIPA (2002a). FIPA Contract Net Interaction Protocol Specification, http://www.fipa.org/specs/fipa00029/

FIPA (2002b), FIPA Communicative Act Library Specification, http://www.fipa.org/specs/fipa00037/

Hevner, A. R.; March, S. T.; Park, J.; Ram, S. (2004). Design Science in Information Systems Research. MIS Quarterly, 28(1), 75-105.

Huget, M.-P.; Koning, J.-L. (2003). Interaction Protocol Engineering. Huget, M.-P. (ed.). Communication in Multiagent Systems, Agent Communication Languages and Conversation Policies, LNCS 2650, Springer, 179–193.

Jadex (2009). Jadex BDI agent system. http://jadex.informatik.uni-hamburg.de (accessed 10.01.2010).

Jaeger, M.; Rojec-Goldmann, G.; Mühl, G. (2004). QoS Aggregation for Web Service Composition using Workflow Patterns. In Proceedings of the 8th IEEE International Enterprise Distributed Object Computing Conference (EDOC 2004), Monterey, 149-159.

Jones, I. (ed.) (2008). Detailed Validation Scenario Definition, BREIN Project Deliverable D3.1.2, http://www.eu-brein.com/index.php?option=com_docman&task=doc_download&gid=44 (accessed 10.01.2010).

Karaenke, P.; Micsik, A.; Kirn St. (2009). Adaptive SLA Management along Value Chains for Service Individualization. In Proceedings of the 1st International Symposium on Service Science (ISSS 2009), Leipzig, 217-228.

Karageorgos, A.; Mehandjiev, N.; Hämmerle, A.; Weichhart, G. (2003). Agent-based optimisation of logistics and production planning. Engineering Applications of Artificial Intelligence, 16, 335-348.

Kirn, St.; Schlageter, G. (1992). Coordinating Autonomous Problem Solvers: The 2 Dimensions of Nested Negotiations. In: Vogt, F.H. (ed.): Information Processing 92 – Proceedings of the 12th World Computer Congress / Volume III: Intelligent Systems and Personal Computers. Madrid, Spain, 7-11 September 1992. Elsevier Science Publishers B.V., North-Holland, 150-156.

Nguyen, T. D.; Jennings, N. R. (2004). Coordinating multiple concurrent negotiations. In Proceedings of the 3rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS), 1064–1071.

OGF (2007). Web Services Agreement Specification (WS-Agreement), Open Grid Forum Proposed Recommendation GFD.107.

Pokahr, A.; Braubach, L. (2007). Jadex User Guide, Release 0.96, University of Hamburg, http://jadex.informatik.uni-hamburg.de/docs/jadex-0.96x/userguide/userguide.pdf (accessed 10.01.2010).

Preist, C.; Bartolini, C.; Byde, A. (2003). Agent-based service composition through simultaneous negotiation in forward and reverse auctions. In Proceedings of the 4th ACM Conference on Electronic Commerce (EC '03). ACM, New York, NY, 55-63.

Schillo, M.; Kray, C.; Fischer, K. (2002). The eager bidder problem: a fundamental problem of DAI and selected solutions. In Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS'02). ACM, New York, NY, 599-606.

Si, Y.-W.; Edmond, D.; ter Hofstede, A. H.M.; Dumas, M.; Chong, C. U. (2005). Specification of Composite Trading Activities in Supply Chain Management. In Proceedings of the IEEE International Conference on e-Technology, e-Commerce and e-Service (EEE05), March 29 - April 01, 2005, Hong Kong, China.

Smith, R. G. (1980). The contract net protocol: High level communication and control in a distributed problem solver. IEEE Transactions on Computers, C-29(12), 1104–1113.

Verma, D. (1999). Supporting Service Level Agreements on IP Networks. Macmillan Technical Publishing.

Walsh, W. E.; Wellman, M. P.; Ygge, F. (2000). Combinatorial Auctions for Supply Chain Formation. In Proceedings of the 2nd ACM Conference on Electronic Commerce (EC00). ACM, New York, NY, 260-269.

Zhang, X.; Lesser, V.; Abdallah, S (2005). Efficient Management of Multi-Linked Negotiation Based on a Formalized Model. Autonomous Agents and Multi-Agent Systems, 10 (2), 165-205.