Expected Utility and Risk Management in Complex Projects

Florina Livia Covaci *florina.covaci@ubbcluj.ro*
Business Informatics Research Center, "Babes-Bolyai" University,
Cluj-Napoca, Romania

Cristian Bologa *cristian.bologa@econ.ubbcluj.ro*
Business Informatics Research Center, "Babes-Bolyai" University,
Cluj-Napoca, Romania

Gheorghe Cosmin Silaghi *gheorghe.silaghi@econ.ubbcluj.ro*
Business Informatics Research Center, "Babes-Bolyai" University,
Cluj-Napoca, Romania

Abstract

Much research work has been conducted to the study of methods for managing ISD projects. This resulted in a large amount of literature on a variety of often normative ISD methods. The increasing projects' complexity provide new challenges regarding management and development. In complex projects scenarios where the outcome is composed of several deployed components, guaranteeing specific contract requirements for the prime contractor of the project is a real challenge. The focus of this paper is to find appropriate methods to facilitate end-to-end contract parameters in complex projects environments by automated supply chain formation and to establish and enforce contract parameters between each pair of component consumer/provider. Communication between agents along the supply chain is done by message exchange and provides propagation of constraints between subcontractors in the supply chain. Our findings reveal that the proposed method is able to address the emerging issues arising in complex projects.

Keywords: Information systems development, Complex projects, Supply Chain Formation, Utility functions, Project management

1. Introduction

Nowadays information systems (ISs) are becoming increasingly complex and create new challenges regarding management and development.

Research in ISD methods can be traced back to the early 1960s, when software engineering gradually moved away from being primarily used and developed by expert engineers and scientists to becoming an essential element of the business world as it is now [1]. ISD methods include various different techniques and processes that aim at controlling complexity and thereby reducing project risks, depending on their nature. These techniques include among others: planning exercises [2], continuous prototyping [3], and stage-based acceptance tests [4]; however they do not address issues that arise from the increased complexity of ISs nowadays.

Schneberger and McLean [5] defined complexity as dependent on a system's number of different types of components, its number of types of links, and its speed of change. Turner and Cochrane [6] propose uncertainty as a dimension of project complexity, which is the extent to which the project goals and means are well defined and are subject to future changes. Benbya and McKelvey [7] introduced the principle of requisite complexity, stating that ISD projects have an intrinsic characteristic which dynamically aligns their complexity with the complexity of the organizational environment they are embedded in and its needs.

Our society is becoming more complex through the development and use of more complex technologies and organizational forms that are both essential to ongoing globalization processes. There is need to understand this complexity and to develop new methodologies that enables the control of this complexity.

The current paper proposes to address this complexity and the emerging issues arising from it. We consider the scenario of a complex IS development project, the IS being a "collection as a whole" of high level components of many technological kinds like software components, data base management systems components, security components, communication components, etc. and connections between them. These components are provided by subcontractors as the IS as a whole cannot be provided by a prime contractor. The subcontractors act in a globalization context and span over geographical borders. The research challenge is providing support for automated negotiation in the Supply Chain (SC) and for linking prime contractor requirements to underlying suppliers to conjointly guarantee end-to-end agreed contract parameters.

The SC is described in terms of a directed acyclic graph where the nodes are represented by the agents. The participants in the supply chain are modeled by self-interested agents that own utility functions and communicate directly with the other participant agents representing their potential supplier/consumer and take actions in order to maximize their utility functions.

The current paper is structured as follows: section 2 describes the emerging issues arising from projects complexity, section 3 provides the background and related work for supply chain formation, section 4 describes our proposed mechanism for negotiation and messages exchange that propagate constraints across the whole project subcontractors, section 4 provides an experimental evaluation of the proposed approach and finally section 5 provides conclusions and future work.

2. Project Complexity and Emerging Issues

ISs complexity has grown as the number of components and their integration has increased and have led to an integration of platforms and applications across organizational and geographical borders.

In complex projects scenarios where the project outcome is the result of several deployed components, guaranteeing specific contract requirements to the prime contractor is a real challenge. The contract parameters of the final ISs as a whole is strongly affected by those components employed to compose it. If just one of the composing services violates the contract requirements, the global contract parameters delivered to the prime contractor might get definitively compromised.

An emergent issue in the scenario stated above is composite decision making. Project managers of complex IS development projects are facing composite decisions making situations like: Should they use the low-price subcontractor? Should they adopt a state-of-the-art technology? Should they use a subcontractor that promises to deliver the component in a shorter time? While making many decisions is difficult, the particular difficulty of making these decisions is that the results of choosing from among the alternatives available may be variable and ambiguous.

In order to address the composite decision making situation that arise in complex project scenarios, we propose using utility functions as a means for incorporating in agreed contracts besides cost, parameters like: quality, delivery constraints etc.

However each subcontractor has a different utility function. For example for one might be more important to have the component delivered in a longer period but at a lower cost while another subcontractor might prefer a higher cost with fast delivering. In order to model this preferences we use weighted utility functions in which each parameter of the contract has a certain utility for a certain subcontractor. The prime contractor and subcontractors at different levels in the supply chain are not aware of others utility functions. They negotiate on specific values for contract parameters and are not aware about the utility that the opposite partner gets.

We denote by $U(v)$ the utility that a participant gets, by obtaining the actual value $v=(v_{i1},v_{i2},...,v_{ik})$ of the component that he gets. When a supplier (seller) negotiates with a consumer (buyer), both parties are interested in obtaining those contract values $v=(v_{i1},v_{i2},...,v_{ik})$

that maximize their utility functions $U(v)$. This means that during the negotiation, each agent sends messages to his neighbors regarding the states of his variables that maximize his own utility function.

In the current paper the utility value obtained by an agent $U(v)$ will be calculated by means of weighted sum as follows:

$$U(v) = \sum_{i=1}^{k} w_i * v_{k,} \qquad with \; \sum_{i=1}^{k} w_i = 1 \tag{1}$$

where $0 <= w_i <= 1$ represent the weights measuring the importance of a given contract parameter i for a certain agent in the chain. Note that the form of the utility functions is not a central piece in the proposed model; the proposed approach can use any form of utility function.

A second issue that arises from the increased complexity of ISs is the increased risks. For example for the prime contractor there is a penalty in his contract with the main client for every day he delivers late. He needs to decide which sub-contractor to use for a critical activity. It is often difficult to argue for using the higher-priced sub-contractor, even if that one is known to be reliable. The lower-bidding sub-contractor also promises a successful delivery, although we suspect that he cannot do so reliably. We need to know if there is any benefit to using the higher-cost sub-contractor, and we suspect it may lie in the greater reliability of performance we expect.

Thus instead of using traditional risk analysis and modelling techniques like Expected Monetary Value (EMV) our approach uses Expected Utility in order to incorporate risk in decision making. Figure 1 describes the influence on the expected utility μ of variable risk and an action (the choice he makes over the possible partners) taken by a subcontractor based on his utility function.

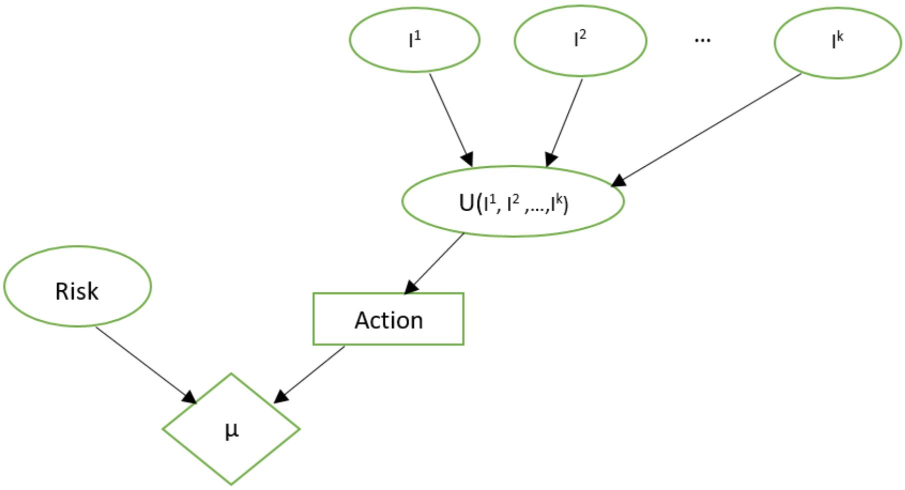

Fig. 1- Influence diagram on expected utility μ

Each agent shares states variables with possible partners in the supply chain $(I^1, I^2, \ldots, I^k)$. The action variable which is different from the states variables provides means for agents to choose among partners which maximize their utility. The utility values are calculated according to each own utility function.

In the proposed decision making mechanism the risk variable has the form of a probability distribution ($P(y|a)$ in equation(2)), that incorporates agent previous experience in achieving similar tasks, the agent reputation etc.

This give raise to the expected utility as a factor between the risk and the action variable.

$$EU[D[a]] = \sum_y P(y|a)U(a) \tag{2}$$

Hence, the action variable provides each agent with a decision situation D. Let $A=\{a^1,\ldots,a^m\}$ be a set of possible actions and a set of states for every negotiated issue I={$i^1,\ldots,i^k$}. The possible action in the SCF context represents which is the chosen partner among the all the possible partners, according to the decision rule *D[a]* of maximizing the expected utility.

$$a^{*} = argmax_{a} EU\big[D[a]\big] \tag{3}$$

3. Background and related work for SCF

The first approaches for SCF [8, 9, 10] addressed the problem by means of combinatorial auctions that compute the optimal SC allocation in a centralized manner. In [11] the authors proposed a mediated decentralized market protocol with bidding restrictions referred to as simultaneous ascending (M+1)st price with simple bidding (SAMP-SB), which uses a series of simultaneous ascending double auctions. SAMP-SB was shown to be capable of producing highly-valued allocations solutions which maximize the difference between the costs of participating producers and the values obtained by participating consumers over several network structures, although it frequently struggled on networks where competitive equilibrium did not exist. The authors also proposed a similar protocol, SAMP-SB-D, with the provision for de-commitment in order to remedy the inefficiencies caused by solutions in which one or more producers acquire an incomplete set of complementary input goods and are unable to produce their output good, leading to negative utility.

Recent papers that consider the SCF problem are using a message passing mechanism in graphical models in order to solve the SCF problem. In [12, 13, 14], a decentralized and distributed approximate inference scheme, named Loopy Belief Propagation (LBP) was applied to the SCF problem, noting that the passing of messages is comparable to the placing of bids in standard auction-based approaches. The authors show that the SCF problem can be cast as an optimization problem that can be efficiently approximated using max-sum algorithm (1). Thus, the authors offer the means of converting a SCF problem into a local term graph, on which max-sum can operate.

In LBP, the SCF problem is represented by a model in which each of the participants' decisions is encoded in single variable. The states of each variable encode the individual decisions that the participant needs to make regarding her exchange relationships plus an inactive state. Moreover, the activation cost for a participant p is encoded by means of a simple term f_p, also called activation term. Each of these activation terms has the participant's variable as its scope and takes value zero for the inactive state and the activation cost for any of the active states.

In order to ensure that decisions are consistent among participants, in LBP, there is a compatibility term for each pair of variables representing potential partners. A compatibility term f_{p1p2} encodes the compatibility between the decisions of the two participants p_1 and p_2. Two participants are in incompatible states whenever one of them is willing to trade with the other, but the other one does not. If two states are compatible, the value of the compatibility term is zero, otherwise is negative infinity.

Hence LBP maps the SCF problem into a set of participant variables $X = \{x_1,...,x_n\}$, a set of activation terms $F_A = \{f_1,...,f_n\}$, one per variable, and a set F of compatibility terms. Then, solving the SCF problem amounts to finding a state assignment for the participant variables in X that maximizes the following reward function:

$$\mathbb{R}_{LBP}(X) = \sum_{x_i \epsilon X} f_i\,(x_i) + \sum_{f_{kl} \epsilon F} f_{kl}\,(x_k, x_l) \tag{4}$$

As LBP suffers of scalability issues in (2) the authors introduce the Reduced Binarized Loopy Belief Propagation algorithm (RB-LBP). RB-LBP is based on the max-sum algorithm and introduces binary variables in order to encode decoupled buy and sell decisions and a selection term and an equality term in order to assure coherent decisions between participants.

4. Complex Projects Development and Supply Chain Formation

In a complex project environment, an entity can be both a supplier and a consumer in different negotiation contexts at the same time. In particular, a participant entity has to confirm that its own suppliers can support the issues it negotiates with its consumer, before it commits to his contract with its consumer. Furthermore, it is assumed that a consumer and its suppliers have conflicting interests on the contract parameters they negotiate on; otherwise, both parties can simply reach an agreement by choosing their optimum in their negotiation space.

Our approach for addressing the emergent issues that arise in complex projects is based on max-sum mechanism for message passing and provides means for optimization of the maximum expected utility by using messages that encode the agents' option for the multiple contract issues and by incorporating risk in agent decision making process

Below we provide a formal description of the supply chain formation problem in complex project environments in terms of a directed, acyclic graph (X,E) where $X =\{X_1, X_2,...,X_n\}$ denote set of participants in the supply chain represented by agents and a set of edges E connecting agents that might buy or sell from another.

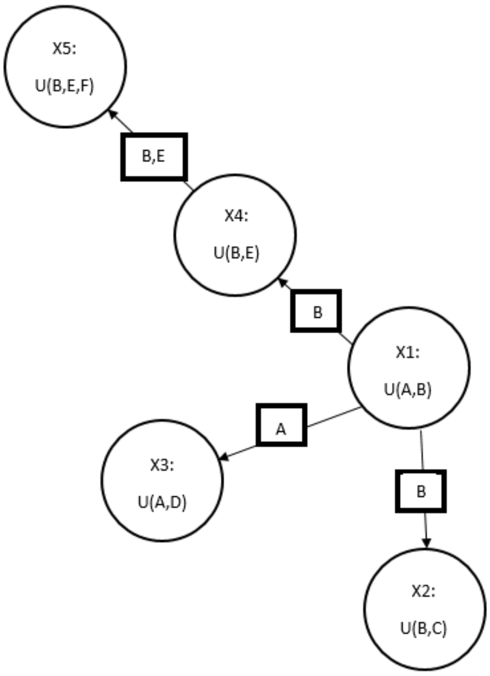

Fig. 2 Example of supply chain with agents sharing state variables

The agents negotiate on multiple contract parameters and negotiation finishes with a contract that is composed of the actual values of the issues that they have agreed on. Communication between agents along the supply chain is done by message exchange and provides propagation of agreed contract parameters between subcontractors in the supply chain.

Below we describe the process of message exchange between two possible partners in order to agree on values of the shared contract issues and assess the utility values according to each own utility functions.

In Figure 2 we have a graph in which each node represents a participant in the supply chain. Agent X_1, the root node will send messages to all his descendants until it reaches the leaves then the messages are send back from leaves to the root node. Each two nodes share at least one issue that they must agree on, and the value that they have agreed on, will be propagated to the root node. Figure 3 shows how the message is send between node X_1 and X_2 and back, when the have to agree on the value for a contract issue B.

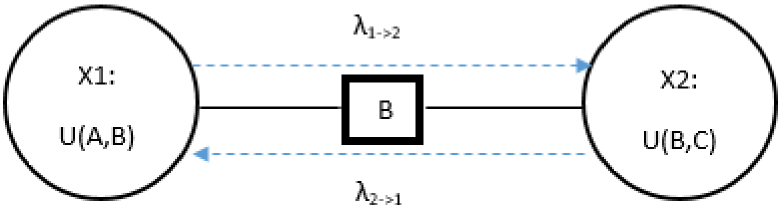

Fig. 3 Messages sent from X_1 to X_2 and back when X_1 and X_2 share issue B

By sending the message in equation (5), X_1 says to X_2 which is his preferred value from the set of values for issue B. Agent at node X_1 evaluates using his utility function, the utility that he gets for each combination of values from the set of values for issues A, B.

$$\lambda_{1\to 2}(B) = max_B\left(U(b_{j,}c_k) + max_A\left(U\left(a_i, b_j\right)\right)\right) \tag{5}$$

X_1 sends the max-marginalization of B over A $max_A\left(U\left(a_i, b_j\right)\right)$and then adds the computed utility of agent X_2 and then computes the max-marginalization of B over the above terms. Agent at node X_2 evaluates using his utility function, the utility that he gets for each combination of values from the set of values for issues B, C. X_2 sends to X_1 the message in equation (6), which is his preferred value from the set of values for issue B.

$$\lambda_{2\to 1}(B) = max_B\left(U\left(a_i, b_j\right) + max_c\left(U(b_c, c_k)\right)\right) \tag{6}$$

X_2 sends the max-marginalization of B over C ($max_c\left(U(b_c, c_k)\right)$) and then adds the computed value for utility of agent X_1 and then computes the max marginalization of B over the above terms.

Each node that is not a leaf is receiving messages from its neighbors, composing new messages and sending them upward to its neighbors. The message exchange between a pair of nodes is a vector of real numbers, one for each possible state of the variable shared by both nodes. Each node will assess messages received from the corresponding neighbor according to his own utility function and will chose among all the possible partners the one that maximizes his utility function.

During the SCF process messages are passed between a consumer and its suppliers agents regarding multiple contract issues: cost, time of delivery, quality constraints etc. The above mechanism uses the max-sum mechanism of passing messages, regarding the values of the issues that the agents are negotiating on, the agents having an exact way to estimate the utility they get, by making use of utility functions. The agreed values of the negotiated issues are reflected in a contract which has a certain utility value for every agent. By using utility functions and incorporating a risk variable, they can assess the benefits they would gain from a given contract, and compare them with their own expectations in order to make decisions.

5. Experimental Evaluation

In order to validate the model proposed in section 4, we have selected PeerSim for implementing our proposed approach. The reasons for making this choice are the PeerSim performance regarding scalability and because it is based on components that allows prototype a new protocol, combining different pluggable building blocks that are in fact Java objects. PeerSim is a single-threaded peer-to-peer simulator that is developed in a modular and scalable way [16].

The simulation starts by reading the configuration file, given as an input parameter that defines the protocols to experiment. Then, both nodes and protocols are created and initialized. After the initialization phase, by default, every instance of the protocols running on each node is executed once per simulation cycle. PeerSim models the set of nodes as a collection through the class peersim.core.Network. The overlay network is represented by a adjacency list where every peer n of the network is connected to a set of neighbors N(n).

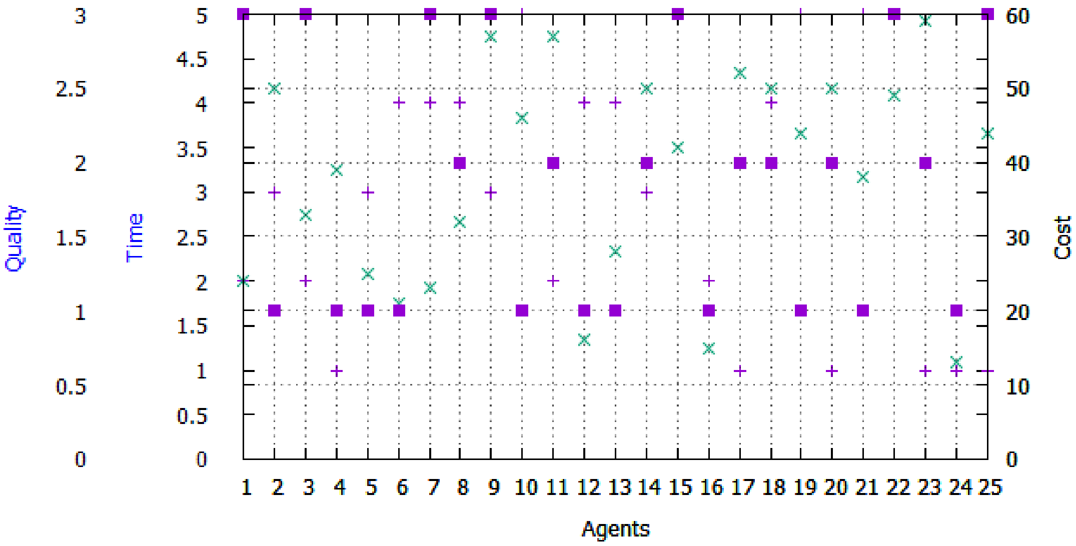

Fig. 4 - Initial random state for variables time, quality and cost

At the beginning of the simulation all the agents in the network are being initialized with random preferred values for the states variables and also for the weights and risk variables used at computing every agent expected utility.

The network used for the current simulation has 25 nodes and each node has three possible partners that he can trade with. Each node has a vector of numeric values that are the preferred states of the negotiated issues. Also each node owns a utility function which is computed as a weighted sum from each of the issues that the agents in the supply chain are discussing on. The utilities are initialized with 0 for all the agents, and when they start interacting their utilities are calculated according to each own utility function.

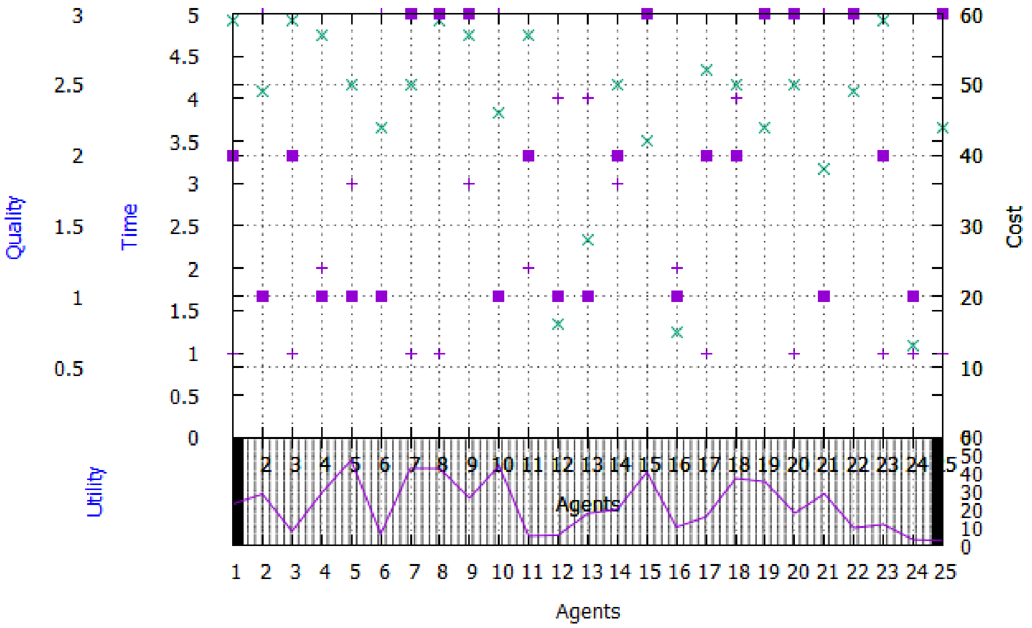

Fig. 5 - Final state for variables time, quality and cost and utility obtained by each agent

In the present simulation we have used as negotiated issues over the supply chain: delivery time, quality and cost. The delivery time is measured in months, quality parameters represent a scale between 1 and 3 (1-Quality reduction requires client approval, 2-Only very demanding modules are affected, 3-Best quality parameters) and the cost is measured in the amount of money spent. We have divided cost by 1000 for the simplicity of the illustration of the values obtained. The initial values of the state variables for the current simulation are the ones presented in Figure 4.

Invoking the simulator, and running the proposed protocol, the interacting agents change their states for variable time, quality and cost as they are propagated upward towards root from the underling suppliers network and obtain the utilities presented in Figure 5.

There are two possible allocation sub-graphs that the implemented protocol finds. One of them is formed of the following nodes $\{X_{11}, X_4, X_1\}$ and the utility that the prime contractor gets is 22.30 with the states of the variables being propagated to the root (t=2; c=57;q=2). The second solution that the implemented protocol finds is the allocation sub-graph being formed of the following nodes $\{X_{23}, X_8, X_3, X_1\}$ and the utility that the prime-contractor gets is 23.07 with the states of the variables being propagated to the root (t=1;c=59;q=2). The propagation of the states of the variables for agent X_2 has two phases: in phase 1 the agent X_3 chooses X_{11} as a partner as it finds that the utility that he gets is higher than the previous one but in phase 2 the agent chooses agent X_8 as he gets a higher utility that the one provided be agent X_{11}. The delivery time is shorter and the cost is higher than the ones at the allocation in solution 1 and according to the utility function of the agent at node X_1 it provides a higher utility. The agent gets a higher utility if the required component is provided in shorter time even it costs more.

The experimental results show that our approach provides means to address the emerging issues in managing complex projects. By using utility functions and risk variables we were able to model each subcontractor preferences and manage risk at every level in the supply chain, guaranteeing the stability of the entire supply chain. Also our approaches addresses the challenge of guaranteeing specific contract parameters for the prime contractor as the values of the state variables are propagated toward root by the message exchange mechanism.

6. Conclusions and future work

In this paper we have proposed a new method for managing emergent issues arising in complex ISs development process. Our approach incorporates multiple negotiated issues and uses utility functions and risk variables in order to compute maximum expected utility. The current approach that this paper presents uses message passing between agents during the supply chain formation.

Project management decisions, can be difficult to make because their implications are often not certain. This is a fact of the activity for most project managers, who often face situations like those explored above: the choice of alternative subcontractors. Each of these decisions poses clear alternatives but different consequences.

Because the consequences of each decision for subcontractors’ contracts parameters combinations are not known with certainty, the choice of the most beneficial decision and its value is typically calculated based on the utility provided of the possible result, taking into consideration also the risk associated with that particular subcontractor. The rational prime contractor and subcontractors decide on the option that offers the highest expected utility.

As a future work we propose to investigate the impact on the development and management of complex ISs when subcontractors make coalitions. Since the possible participants in a SCF might involve more or less self-interested agents and hence more or less utility oriented we propose to investigate how coalition formation among the subcontractors may affect the development of complex ISs.

Aknowledgement

G.C. Silaghi acknowledges support from UEFISCDI under project PN-II-PT-PCCA-2013-4-1644.

References

1. Avison, D., Fitzgerald, G.: Methodologies for Developing Information Systems : A Historical Perspective. The Past and Future of Information Systems: 1976–2006 and Beyond. pp. 1-2 (2006)
2. Schwaber, K., Beedle, M.: Agile Software Development with Scrum. Development.,Vol. 34. doi:10.1109/2.947100 (2001)
3. Naumann, J. D., Jenkins, A. M.: Prototyping : The New Paradigm for Systems Development. s.l. : MIS Quarterly, Vol. 6 (1982)
4. Bishop, C.M.: Pattern recognition and machine learning, New York: Springer (2006)
5. Schneberger, S. L., McLean, E. R.: The Complexity Cross: Implications for Practice, Vol. 9. (2003)
6. Turner, J.R., Cochrane, R.A.: Goals-and-Methods Matrix: Coping with Projects with well Defined Goals and/or Methods of Achieving Them. In: International Journal of Project Management (1993)
7. Benbya, H., McKelvey, B: Toward a complexity theory of information systems development, Vol. 19 (2006)
8. Walsh, W. E., Wellman, M. P., Ygge, F.:Combinatorial auctions for supply chain formation. In : Proceedings of the 2nd ACM conference on Electronic commerce (2000)
9. Collins, J., Ketter, W., Gini, M., Mobasher, B.: A multi-agent negotiation testbed for contracting tasks with temporal and precedence constraints. In. : International Journal of Electronic Commerce, Vol. 7 (2002)
10. Cerquides, J., Endriss, U., Giovannucci, A., Rodriguez-Aguilar, J. A.: Bidding languages and winner determination for mixed multi-unit combinatorial auctions. In : IJCAI, Morgan Kaufmann Publishers Inc. (2003)
11. Walsh, W. E., Wellman, M. P.: Decentralized supply chain formation: A market protocol and competitive equilibrium analysis. In : Journal of Artificial Intelligence Research, Vol. 19 (2003)
12. Winsper, M., Chli, M.: Decentralised supply chain formation: A belief propagation-based approach. In : Agent-Mediated Electronic Commerce (2010)
13. Winsper, M., Chli, M.: Using the max-sum algorithm for supply chain formation in dynamic multi-unit environments. In: Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems (2012)
14. Winsper, M., Chli, M.: Decentralized supply chain formation using max-sum loopy belief propagation. In. : Computational Intelligence, Vol. 29 (2013)
15. Penya-Alba, T., Vinyals, M., Cerquides, J., Rodriguez-Aguilar, J. A.: A scalable Message-Passing Algorithm for Supply Chain Formation. In. : 26th Conference on Artificial Intelligence (2012)
16. Montresor, A., Jelasity, M.: PeerSim: A scalable P2P simulator. In : Proc. of the 9th Int. Conference on Peer-to-Peer, Seattle (2009)
17. Baccarini, D.: The Concept of Project Complexity-A Review. In: 14th International Journal of Project Management (1996)
18. Boehm, B. W.: A Spiral Model of Software Development and Enhancement. In: IEEE Computer (1987)