Towards Improving Supply Chain Coordination through Agent-Based Simulation

Areti Manataki, Yun-Heh Chen-Burger, and Michael Rovatsos

Abstract. One of the most significant paradigm shifts of modern business management is that individual businesses no longer compete as autonomous entities but rather as supply chains. However, the majority of companies, especially small and medium enterprises, fail to design and manage their supply chains in a profitable way, as it is difficult to understand the complex dynamics of Supply Chain Management (SCM). In this paper we argue that agent technologies can provide an intelligent solution to the improvement of SCM. We present a multiagent-based framework for simulating supply chain (SC) operation and re-configuration, with the vision of helping to improve overall SC performance and coordination. The suggested key innovation lies in the better explanation of simulation results and its attractiveness to SCM practitioners. Its theoretical conceptualisation, a logic-based formalisation and the system's architecture that combines agent technologies with business rules and business process modelling are presented.

1 Introduction

Since the 1990s firms realised the importance of Supply Chain Management for gaining and sustaining a competitive advantage. It is only recently, however, that the business world has moved from the traditional "competing firms" model to a "competing supply networks" perspective [12, 8]. The current vision of SCM involves the alignment and integration of SCs, hence a systemic and holistic view of SCM is becoming prominent.

However, SC reality is far from this vision, as most SCs are underperforming and no integration is achieved. This is due to the high complexity of SCM and the lack of understanding of SCM dynamics, especially when it comes to overall SC coordination [13].

Existing approaches towards this problem span over two extremes: they are either theoretical, thus not dealing with SCM dynamics; or they are computational, but with limited theoretical grounding. Furthermore, the few simulation tools that

Areti Manataki · Yun-Heh Chen-Burger · Michael Rovatsos
School of Informatics, The University of Edinburgh, Edinburgh EH8 9AB, UK
e-mail: `A.Manataki@sms.ed.ac.uk`,
`{jessicac,mrovatso}@inf.ed.ac.uk`

tackle the SC coordination problem are not useful in the new era of SC-based competition as they do not adopt a holistic view of SCM, and they fail to explain the business rationale behind simulation results. On the other hand, agent technologies have been successfully applied for improving SCM [11], but limited work has been done on SC coordination. Moreover, despite their success, multi-agent systems are not widely adopted in industry.

Considering the deficiencies of existing approaches, we have identified the following requirements for a solution to the SC coordination problem: An intelligent solution is needed [10] that focuses on inter-organisational coordination [13]. The ability to explain simulation results and attract SC managers are also important. The latter means that the solution should be easy to use and use the same language as SCM practitioners.

In this paper it is argued that agent technologies can provide an intelligent solution for SCM improvement that satisfies all the above-mentioned requirements. We thus present a three-tiered multiagent-based framework for simulating SC operation and re-configuration, with the vision to help improve overall SC performance and enhance SC coordination, thus assisting SCM decision-making. The intelligence provided by agent technologies and the focus on global aspects of SCM (i.e. overall SC performance and SC-wide coordination) tackle the first two requirements. Furthermore, a logic-based approach is suggested in order to enable better explainability of simulation results. Finally, we argue that a theoretical grounding, the adoption of SCM standards and the incorporation of technologies that are popular in industry, will make the suggested solution appealing to SCM practitioners.

The remainder of the paper is as follows: Section 2 provides background information on SCM, explains the motivations for the use of agent technologies for SCM, and describes relevant work. Section 3 presents the suggested three-phased multiagent-based framework for SC simulation, and the system architecture and design decisions are explained. Section 4 discusses lessons learnt and future work.

2   Background Information

A *Supply Chain* is defined as "all parties involved, directly or indirectly, in fulfilling a customer request" [3]. Along this network of parties (e.g. manufacturers, suppliers, distributors, retailers, customers, etc.) there are three types of flows: downstream flow of products, upstream flow of funds and a bidirectional flow of information. Managing these flows with the objective of maximising total SC profitability is defined as Supply Chain Management. Managing a SC is a difficult task due to its dynamic nature and the SC members' conflicting objectives. We identify three main SCM problems: SC planning and demand forecasting provides estimations on future demand, SC configuration specifies the system's structure, policies and processes in a static way, while SC operation/coordination refers to the SC members' actions and interactions, leading to the above-mentioned flows.

Multiagent Systems (MAS) have been widely applied on the SCM context, as they match the SC nature. As Moyaux et al. [11] argue, "supply chains are made up of heterogeneous production subsystems gathered in vast dynamic and virtual

coalitions; intelligent distributed systems, e.g. MAS, enable increased autonomy of each member in the SC". SC members also have the same characteristics as agents [11]: They are autonomous (i.e. the business operation of a SC member does not involve direct intervention of others, while it has control over its actions and internal state), social (i.e. there is high interaction between SC members), proactive (i.e. they perceive their environment, especially their market and competition, and respond to it), and reactive (i.e. they take initiatives in order to maximise their profits). Further motivations for using MAS for SCM are provided in [10], including distributed problem solving and the facilitation of SC integration. It is also worth mentioning that agent technologies are suitable for tackling all three main SCM problems, either by automating, simulating or recommending solutions. Agent-based negotiation and optimisation can be utilised for SC configuration, agent-based coordination is linked to SC operation, while the agents' learning capabilities can facilitate SC planning.

Representative work in agent-based simulation for SC coordination and configuration includes [5] and [15]. The former adopts a systemic view of SCM and introduces conversation plans that capture coordination knowledge; however no supportive business theories are presented and SC operations are not explicitly addressed. The latter utilises a library of structural and control elements and a classification of messages to simulating all SC flows, but it does not support knowledge discovery or explain simulation results. SC configuration is automated in [4] through the combination of MAS with machine learning; however, no underlying business theory is presented.

3 A Multiagent-Based Framework for Overall SC Simulation

With the aim of providing an intelligent solution for SCM improvement, that is realistic and easy to use by industry, we propose a three-phased agent-based SC modelling and simulation framework. This framework tackles the SC operation and configuration problems, and complies with business theories and SCM standards. As shown in Fig. 1, the suggested framework consists of three phases: conceptualisation, formalisation and simulation of SCM. The *conceptualisation* phase involves abstracting the domain of SCM through suitable constructs. During the *formalisation* phase the selected constructs of Phase 1 are defined and libraries of SCM concepts are developed. The *simulation* phase utilises the libraries of Phase 2 and simulates SC operation and configuration (i.e. SC partner selection), thus explaining dynamic aspects of SCM and allowing for experimentation. Note that throughout our framework a logic-based approach is adopted in order to allow for knowledge-enriched analysis of the domain.

3.1 Conceptualising SCM

We conceptualise SCM through three basic components: SC roles, SC services and SCM processes. Each SC role provides a specific SC service through the execution of the corresponding enabling SCM processes. Hence, we regard the SC as
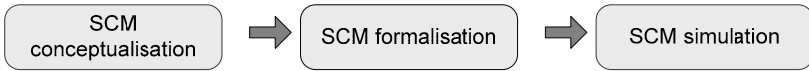
Fig. 1 Three-phased framework for SC simulation

a virtual organisation (VO) with the goal of satisfying the requirements of the final customer and maximising total SC performance. A high-level plan towards achieving this goal consists of basic functions or tasks, which we call SC services. SC members may bid for and be delegated a specific SC service, thus being assigned the corresponding SC role. Last, a SC role prescribes the execution of SCM processes that realise the delegated SC service; hence, the SCM process model of a SC role constitutes its internal plan towards delivering the delegated SC service.

A *SC service* is defined as an archetypical function that supports the flow of products requested by the final customer. Illustrative examples include the following: manufacture final product, sell product to final customer, transport product, etc. A *SC role* is defined as the combination of the archetypical service of a SC member towards the SC (and hence the corresponding position in the SC network) and its business model, i.e. how it makes money from what it does [17]. Adopting the business model typology by [17], we recognise SC roles such as supplier-creator, manufacturer and retailer. Note that a SC member may adopt multiple SC roles within a certain SC. Last, a *SCM process* is defined as a SC role's business process supporting the flow of the requested product to the final customer. Receive product, send invoice and consolidate order are examples of SCM processes.

In order to make our solution easier to understand and more realistic and attractive to SC managers, we have based our conceptualisation on widely accepted business theory. Hence, adopting a holistic approach to SCM, we view the SC as a VO, where the theory of organization design can be applied. Galbraith [6] has suggested a star model of the five major components of organization design, among which processes and structure are the prominent ones. The main dimensions of organizational structure are power and authority, reporting relationships, and organizational roles. But since SCs are known to have a chain VO structure with respect to power, authority and reporting relationships, it is reasonable to minimise SC structure to the SC role dimension. Furthermore, because SC roles are based on archetypical SC services, SC services are a construct of our conceptual model. Lastly, by recognising SCM processes as a main component of SCM, we comply with [1], according to which process improvement orientation is the most popular SCM construct.

3.2 Formalising SCM

Adopting a logic-based approach, we use first-order predicate logic to define the SCM constructs of Phase 1, as shown at the schemata provided below. A SC service is defined through its id, name and position in the SC (i.e. upstream or downstream). A SC role is defined through its id, name, the provided SC service, the corresponding business model and its SC position. The schema for SCM process

complies with the Fundamental Business Process Modelling Language (FBPML), a logic-based business process modelling language that guarantees both rich visual modelling methods and formal semantics [2]. Following the FBPML activity specification, a SCM process is defined through its hierarchical position (id), name, triggering events, preconditions and actions.

```
SCservice(ID, Name, Position)
SCrole(ID, Name, Service, BusinessModel, NetworkPosition)
SCMprocess(Position, Name, TriggeringEvents, Preconditions, Actions)
```

The formalisation phase also involves populating libraries for the SCM constructs, thus providing instances that can later serve as simulation building blocks. In order to make these libraries SCM practitioner-friendly, they should be based on widely accepted SCM theory. Therefore, we suggest the use of the Supply Chain Operations Reference (SCOR) model [14] for the SCM process library, as it is the most widely used standard within the SCM community.

3.3   Simulating SCM

This phase simulates SCM behaviours with respect to the SC operation and configuration problems, and hence two interrelated modules are recognised. The SC operation module simulates the actions and interactions of SC members for a given SC configuration and measures overall SC performance. The SC configuration module simulates the selection and contracting process with SC partners for any SC member. Hence, the integration of the two modules allows simulation of the SC operation where the SC network may be reconfigured at run-time. This way, complicated questions can be answered, such as: How flexible is the SC in the case where supplier X fails to deliver a promised order? In this paper we focus on the SC operation module, while a brief introduction is given on the SC configuration module.

An agent-based *conceptual design* of the simulation environment is proposed. We regard each SC member as an intelligent agent that can decide on SCM issues, execute processes and communicate with other SC members. Simply put, in our framework a SC member can think, act and interact, and is consequently conceptualised as an agent consisting of three layers:

- Reasoning layer: corresponds to the beliefs, desires and intentions of the agent, and drives its decision-making towards actions
- Process layer: corresponds to the agent's ability to execute processes, thus acting upon the environment
- Communication layer: corresponds to the agent's ability to receive and send messages to other agents.

We suggest the use of business rules (BR) to represent a SC agent's reasoning layer, while its decision-making process can be driven through a reasoning engine. The SC agent's process layer can be represented by a business process model (BPM), the execution of which will be facilitated with the use of a workflow engine. Finally, the SC agent's communication layer can be represented by communication predicates, which will operate upon a communication environment.

3.3.1   Functionality and Architecture of the SC Operation Module

The SC operation module reads a SC configuration and simulates the actions and interactions between SC members. Specifically, its inputs include information on SC members, market demand and the final product. Its outputs include real-time SC operation information (e.g. supplier X has completed the execution of process p34-receive order), overall SC performance (e.g. overall cost is 325), and business process analysis results (e.g. possible bottleneck p36 for supplier s12).
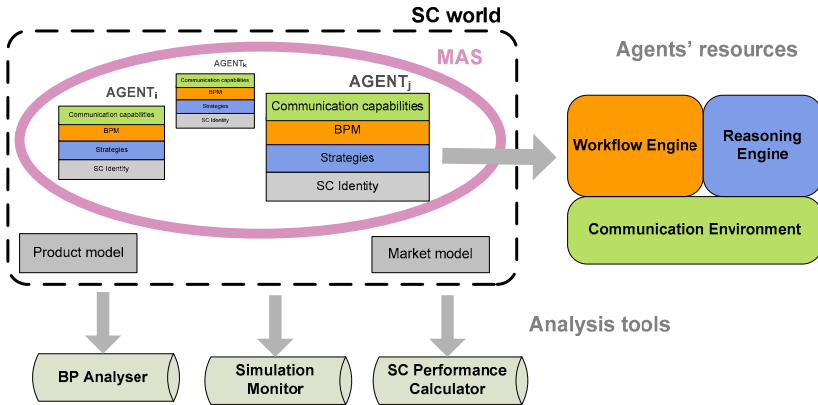

Fig. 2 Architecture of the SC operation simulation module

The suggested architecture for the SC operation simulation module is presented in Fig. 2 and it is based on the functionality and conceptual design explained previously. Three main components can be seen in Fig. 2: SC world, agents' resources and analysis tools. The *SC world* consists of a MAS of SC agents, the SC product and the market demand for that product. The product and market models are static models describing the SC's product (e.g. bill of materials, attributes, etc.) and the final market demand (e.g. average order amount and order frequency). A SC agent consists of four subcomponents: SC identity, strategies, BPM and communication capabilities, which define the SC agent in a static way. In order to exhibit dynamic behaviour, a SC agent uses resources that drive SC simulation. The *resources* that are available to SC agents are: a workflow engine, a reasoning engine and a communication environment. As implied by the colours in Fig.2, these resources are linked to the SC agent's components: The workflow engine executes processes of an agent's BPM, and thus updates its workflow state. Similarly, the reasoning engine reads the SC agent's strategies (defined as BR) and turns them into decisions towards actions for each state. The communication environment allows the exchange of messages within the SC through an appropriate infrastructure (e.g. message exchange channels or a blackboard) and a protocol. Lastly, the analysis of the simulation is enabled by three *analysis tools*: First, the simulation monitor is understood as a simulation controller, making sure that the SC world is correctly modelled and that no rules are broken during simulation.

Second, the SC performance calculator reads the overall SC simulation results and computes its performance in terms of time and cost. Third, the BP analyser studies the overall SC workflow in order to detect bottlenecks, unreachable points, etc.

Adopting a knowledge-based approach, we suggest a *logic-based implementation* of the SC simulation environment. This is work in progress; however a first version of a declarative workflow engine has been developed and tested on a real-world case. The declarative workflow engine DeWE [9] has been designed with respect to FBPML specification to simulate BPM execution and measure time and cost. DeWE has been successfully used for analysing Dell's SCM, and hence it is a good starting point for driving the process-oriented behaviour of SC agents.

3.3.2 Design Decisions Regarding the SC Configuration Module

The SC configuration module is used when a SC member needs a new partner upstream or downstream, and it simulates the selection and contracting process. Facilitating SC configuration is important, as the reconfiguration process takes place at run-time, thus resulting in possible delays and affecting overall SC performance. Furthermore, a new SC configuration defines new rules of actions and interactions between SC members, thus affecting SC coordination. As far as the module's functionality is concerned, its inputs include the selection criteria of the contracting SC member, the attributes of all candidates and their trading strategies. Its output is the new SC configuration, as defined by the agreed contract.

We propose the use of three agent-based mechanisms for simulating the SC configuration process. Firstly, an auction takes place, which allows the interested SC member to attract possible SC partners and receive offers. Secondly, the SC member selects one of the offers/candidates based on internal criteria (i.e. a constraint satisfaction problem). Thirdly, the SC member may negotiate with the selected candidate to set a contract. In order to make the simulation of this problem as realistic as possible, we suggest the use of appropriate business theories for the interaction protocol and the selection process, such as [7, 16].

4   Conclusions and Future Work

Recognising the recent shift towards SC-based competition, and the need for concepts and tools to assess the performance of an entire SC [12], we have suggested a three-phased multiagent-based framework for conceptualising, formalising and simulating SC operation and configuration. We believe that a theoretically well grounded conceptualisation, a formalisation adopting the SCOR model, and the coupling of agent technologies with business process modelling and business rules for the simulation, will make the system more appealing to SC managers and will allow them to directly incorporate their business rationale. By adopting a logic-based approach, we also wish to make this business rationale transparent and support the user's understanding of the SC simulation results. Furthermore, the combination of agent technologies with business operations, the incorporation of the effect of delays due to real-time communication or SC configuration on overall SC performance, and the provided business process analysis throughout the SC are aspects of the added value of the proposed system.

With the vision of an even more realistic agent-based simulation environment, our future work is to design a SCM communication protocol based on appropriate business theories and complete the implementation of the suggested framework.

References

1. Burgess, K., Singh, P.J., Koroglu, R.: Supply chain management: a structured literature review and implications for future research. International Journal of Operations & Production Management 26, 703–729 (2006)
2. Chen-Burger, Y.H., Tate, A., Robertson, D.: Enterprise Modelling: A Declarative Approach for FBPML. In: European Conference of Artificial Intelligence, ECAI, Lyon, France (2002)
3. Chopra, S., Meindl, P.: Supply Chain Management: Strategy, Planning and Operation. Prentice Hall, Englewood Cliffs (2003)
4. Emerson, D., Piramuthu, S.: Agent-Based Framework for Dynamic Supply Chain Configuration. In: Proceedings of the 37th Annual Hawaii International Conference on System Sciences, January 5-8. IEEE Computer Society, Washington (2004)
5. Fox, M.S., Barbuceanu, M., Teigen, R.: Agent-Oriented Supply Chain Management. International Journal of Flexible Manufacturing Systems 12(2-3), 165–188 (2000)
6. Galbraith, J.: Designing Organizations: An Executive Briefing on Strategy, Structure, and Process. Jossey-Bass, San Francisco (1995)
7. Goldkuhl, G.: The six phases of business processes – business communication and the exchange of value. In: 12th biennial ITS conference Beyond convergence (ITS 1998), Stockholm (1998)
8. Lambert, D.M., Cooper, M.C.: Issues in Supply Chain Management. Industrial Marketing Management 29(1), 65–83 (2000)
9. Manataki, A., Chen-Burger, Y.H.: Analysing Supply Chain Strategies Using Knowledge-Based Techniques. In: Proceedings of the 3rd KES Symposium on Agents and Multi-Agent Systems – Technologies and Applications, Invited Session on Digital Economy, Uppsala, Sweden, June 2009. Springer, Berlin (2009)
10. Mitkas, P.A., Nikolaidou, P.: Agents and Multi-Agent Systems in Supply Chain Management: An Overview. In: Protogeros, N. (ed.) Agent and Web Service Technologies in Virtual Enterprises, ch. 13. Information Science Reference, New York (2008)
11. Moyaux, T., Chaib-draa, B., D'Amours, S.: Supply Chain Management and Multi-agent Systems: An Overview. In: Chaib-draa, B., Müller, J.P. (eds.) Multiagent based Supply Chain Management, ch.1. Springer, Berlin (2006)
12. Srai, J.G., Gregory, M.: A supply network configuration perspective on international supply chain development. International Journal of Operations Management 28(5), 386–411 (2008)
13. Stadtler, H., Kilger, C.: Supply chain management and advanced planning: concepts, models, software and case studies. Springer, Heidelberg (2005)
14. Supply Chain Council, Supply-Chain Operations Reference-model, Version 9.0 (2008), http://www.supply-chain.org
15. Swaminathan, J.M., Smith, S.F., Sadeh, M.: Modeling Supply Chain Dynamics: A Multiagent Approach. Decision Sciences 29(3), 607–632 (1998)
16. Weber, C.A., Current, J.R., Benton, W.C.: Vendor selection criteria and methods. European Journal of Operational Research 50, 2–18 (1991)
17. Weill, P., Malone, T.W., D'Urso, V.T., Herman, G., Woerner, S.: Do Some Business Models Perform Better than Others? A Study of the 1000 Largest US Firms, MIT Center for Coordination Science Working Paper No. 226, MIT Sloan School Working Paper 4615-06 (2004)